POWERING A WIRELESS SENSOR NETWORK FOR MACHINE CONDITION MONITORING

David Nku

A thesis submitted to the University of Ottawa in partial Fulfillment of the requirements for the for the degree of Master of Applied Science in Electrical and Computer Engineering

Department of Electrical and Computer Engineering
Faculty of Engineering
University of Ottawa

Abstract

Failure of a machine can lead to production downtime and significant financial losses. Condition monitoring is implemented to avoid such downtime and devices can be used to collect data used for monitoring machine health. Vibration data is the most common type of data used for predicting machine failure. To reduce the need for hazardous cables, such devices are often battery-operated, but this can decrease monitoring device lifespans to less than 3 years, if non-rechargeable batteries are used. This thesis first proposes a design framework for implementing radio frequency energy harvesting (RFEH) at a network level. All of the necessary inputs and parameters to ensure the successful implementation of RFEH for a wireless sensor network are explored. A second design framework is then proposed for using RFEH as a source of energy to power devices for condition monitoring. This includes a power analysis of all device components, as well as the design details for an implementation of wireless power transfer using a wireless transmitter and receiver. A comparison of different types of energy sources for the device is given, followed by a case study, using commercially-available components. A simulation is used to analyze the trade-offs for different values of RFEH parameters, trading off the total cost of implementation with the system's lifetime, based on total energy consumed.

Acknowledgments

This work has been the result of a lot of effort and sacrifice. But there are many people without whom this work would not have been possible. First, I would like to give thanks to God for all the blessings he has given me and because he has always been present in my life. Without his presence this work would not have taken place. Next, I would like to thank my father who supported me financially throughout my school career, and my mother who always encouraged me. I would also like to thank my supervisors, David Knox and Patrick Dumond. They have always been present, available, and motivated to help me complete this work.

I would like to say a word about the university. The environment, but also the University of Ottawa system, has been a huge benefit. Thanks to the employment opportunities offered by the university, I was able to work on my thesis while gaining experience in the workplace.

Table of Contents

List of Figures

List of Tables

Terms and Abbreviations

AC: Alternating Current

ALG: Application Layer Gateway

CM: Condition Monitoring

COTS: Commercial Off-The-Shelf

DC: Direct Current

EIRP: Effective Isotropic Radiated Power

EMF: Electromotive Force

FCC: Federal Communications Commission

GPS: Global Positioning System

ID: Identification

MEMS: Micro-Electro-Mechanical Systems

MT: Multiple Transmitter

mW: Milliwatts

NCC: Node Coverage Count

RF: Radio Frequency

RFEH: Radio Frequency Energy Harvesting

RFID: Radio Frequency Identification

RPM: Rotation Per Minute

STMN: Single Transmitter Multiple Nodes

STSN: Single Transmitter Single Node

PRX: Power Receiver

PTSA: Power Transmission Signal Attenuation

PTX: Power Transmitter

SNR: Signal to Noise Ratio

WLAN: Wireless Local Area Networks

WPT: Wireless Power Transfer

WSN: Wireless Sensor Network

Chapter 1

Introduction

1.1 Condition Monitoring Overview

Large machines are often used to manufacture products in bulk. As such, machine failures can be very expensive, especially during extended downtime [1]. These failures can be caused by improper operation, failure to perform preventative maintenance, accidents, corrosion, misalignment, bearing failure, etc. [2]. Most equipment requires regular maintenance to optimize performance and ensure longevity. Condition-based monitoring, often abbreviated to condition monitoring (CM), is used when the operating condition of a piece of equipment is monitored using many elements, including real-time data collected from the equipment via sensors of different types [2],[3]. By detecting changes in sensed data, signs of potential failures can be identified before breakdowns occur. If faults can be detected early enough, repairs can be scheduled at a convenient moment, avoiding either unnecessary part replacement or costly, unexpected machine shutdowns, especially if the system in question is critical to production. Generally, devices that contain sensors are mounted directly on the equipment that they are monitoring and usually require power to operate [4]–[6]. In a manufacturing environment, this power can be supplied to devices via cables. However, monitored equipment points can be difficult to access [7], requiring a complete machine teardown if devices are being installed or replaced. Moreover, the dynamic nature of many machines and the hazards presented by wires often prevent their use altogether [8],[9].

To address this problem, many companies now use non-rechargeable battery-based technologies to provide remote monitoring using an internet of things (IoT) approach [10]. However, non-rechargeable battery technologies are typically limited to three-year battery lifespans and can require the replacement of the entire device when the battery fails [11]. This presents a significant problem in manufacturing plants containing thousands of individual devices, especially when they are difficult to access. Rechargeable energy storage can be used instead of non-rechargeable energy storage, if a charging method is possible. Many different charging methods exist, including many that harvest energy from distinct sources (e.g. solar, thermal, wind, kinetic, etc.). Each method has advantages and disadvantages depending on the environment. Radio frequency energy harvesting (RFEH) is growing in popularity as a power source for ultralow-power wireless devices for indoor industrial environments [12], [13]. This technology can enable battery-less IoT devices, but also enables wireless charging at a longer distance than commercially available charging pads that use inductive coils. These inductive coils require that devices to be brought into close (i.e. almost touching) proximity to an inductive charger [14].

To use RFEH with a typical wireless CM device, a reduction in the total amount of operating energy required must be considered. As illustrated in Figure 1-1, typical wireless CM devices consist of: a power management unit (PMU), a microcontroller unit (MCU), a wireless communication component, a sensor, and an energy source. In this thesis, a design framework for selecting these components is proposed, along with a study of the impact of using RFEH to provide wireless power for charging or powering CM devices directly (i.e. without primary energy storage). When considering RFEH as the power source for these devices, additional components are required. These include a power transmitter (PTX) and a power receiver (PRX). Finally, systems with either direct RFEH or RFEH combined with rechargeable (RB)/non-rechargeable batteries (NRB) are studied, with the goal of increasing wireless CM device lifetime.

1.2 RFEH in Wireless Sensor Networks

In a wireless sensor network (WSN), nodes are often powered by batteries. However, these battery-powered sensor devices face a limitation on their energy source's lifetime, requiring them to be changed or recharged, depending on their energy source type, every time they are depleted. Several researchers have shown that using energy harvested from an ambient radio frequency (RF) source or a dedicated RF source is feasible for powering single sensors [15]–[17]. However, the amount of energy harvested is small, and the maximum allowable distance between the sensor and the power source might prevent cost-effective large-scale implementations.

Various system inputs are important when using RFEH in a WSN. These include node energy consumption, node distribution, and power transmission signal attenuation (PTSA) due to non line-of-sight (NLOS) transmission conditions. RF energy can be provided to sensor nodes by using two different antenna types (omni-directional and directional), as well as two power architecture types (stationary and mobile). However, their ability to provide energy to all nodes in a WSN differs depending on the conditions at hand. The system inputs and parameters considered in this work are shown in Figure 1-1. To understanding how device components, as well as system inputs and parameters affect RFEH in a WSN, a simulation will be used to determine their effect on the implementation cost and lifetime of a WSN.

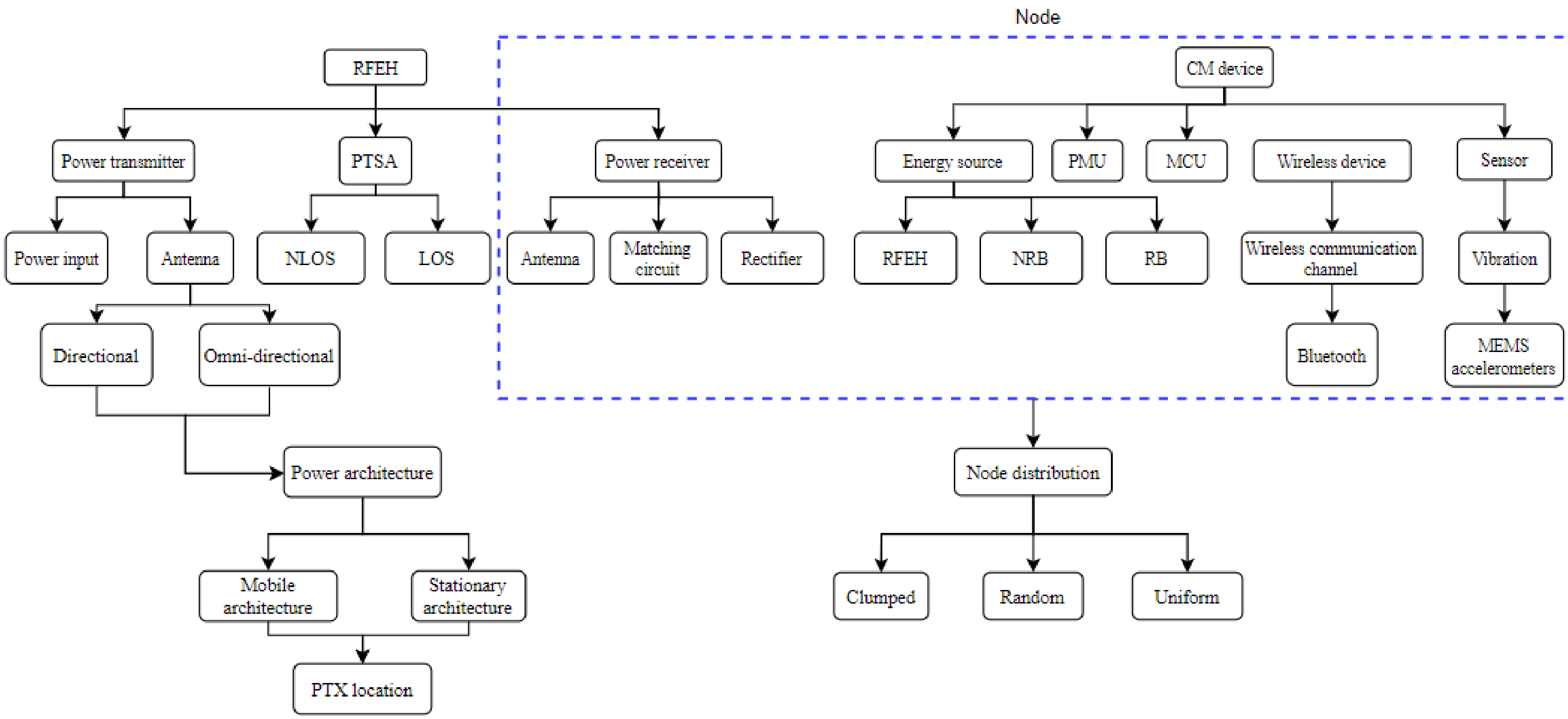


Figure 1-1. Power architecture taxonomy for wireless sensor networks

1.3 Thesis Motivation

With advancements in technology, device component energy consumption is decreasing at a fast pace, allowing energy harvesting technologies to become feasible in powering low-power devices. Despite numerous studies on RFEH in WSNs [18]–[21], RFEH for CM is still not widely used in industry as most solutions rely on battery powered devices [22]–[24]. In fact, most research is currently focused on using RFEH to power a single device rather than looking at optimizing networks based on current knowledge and technology. Moreover, there has been little work that characterizes the financial cost of using RFEH in large-scale WSNs with hundreds of sensors requiring energy. With technology rapidly improving, there is a need for a framework that can be used to observe the impact that individual CM device components and user inputs will have on the overall system and give an insight into which parameters should be improved in the short term and how.

1.4 Thesis Objectives and Contributions

Figure 1-2 shows an example of a typical WSN environment powered by RFEH. For this thesis, the following question will be considered:

Given n nodes in a WSN, what is the optimal placement of PTXs for use with RFEH to minimize the total number of PTXs required to power all nodes, minimize the total cost of the system and maximize network lifetime in the following cases:

a. Fully powered by RFEH and without using batteries
b. Combining RFEH and batteries, while minimizing the number of non-rechargeable WSN nodes (unreachable via RFEH) in the resulting hybrid network

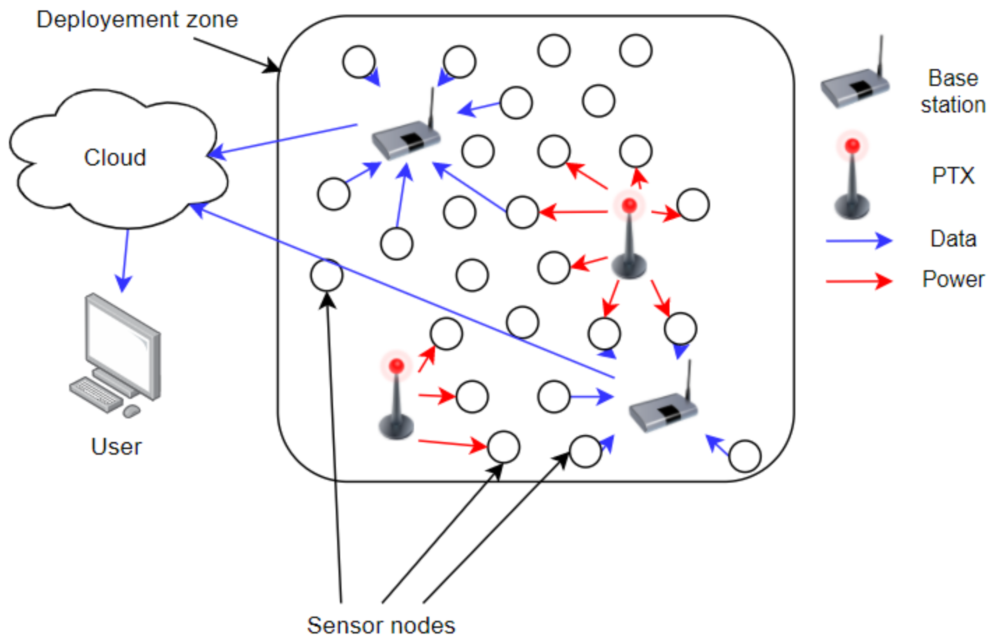


Figure 1-2. RFEH-powered WSN system overview

This question will be studied under the following fixed inputs:

a. Node energy consumption

b. Node distribution

c. Environmental PTSA

In answering this question, this thesis provides the following contributions:

- Identified and described critical parameters for utilising RFEH as the primary source of power for nodes in a WSN.
- Considered two different methods (stationary and mobile) for supplying energy to WSN nodes via RFEH.
- Developed a design framework for RFEH-powered CM devices and analyzed the effect of component selection on the CM device's power consumption.

- Proposed a ‘hybrid’ method for powering low-power condition monitoring devices using a combination of rechargeable, non-rechargeable storage units, as well as directly harvested energy.
- Developed a simulator that illustrates the effects that the most critical inputs and parameters have on a system’s lifetime and implementation cost when using RFEH to power nodes in a WSN.

Table 1-1 provides the inputs, outputs, and parameters that will be studied in this work.

Table 1-1. Inputs, outputs, and parameters considered for this work

Inputs	Node energy consumption	• Duty cycle • Node composition (MCU, PMU, energy storage, wireless unit, sensor)
	Node distribution	• Clumped distribution • Random distribution • Uniform pattern distribution
	PTSA	• LOS • NLOS
	Node density	• Number of nodes • Size of the environment
Parameters	PTX antenna	• Omni-directional • Directional • PTX energy transmitted
	Power architecture	• Stationary • Mobile
Output 1	Maximum allowable distance between the PTX and any given node	
Output 2	Final PTX locations	

1.5 Thesis Structure

The rest of this thesis is divided as follows:

- Chapter 2 provides a literature review of WSNs, CM devices, and RFEH technologies.
- Chapter 3 considers the important inputs and parameters for using RFEH to power WSNs and presents their effect on the WSN system.
- 0 proposes a design framework for a RFEH-powered condition monitoring device and determines the energy consumption of each component.

- **Erreur ! Source du renvoi introuvable.** simulates WSNs using the proposed architectures, and discusses the effects of the inputs and parameters presented in previous chapters on the simulation outcomes.
- Chapter 6 provides conclusions and recommendations for future work.

Chapter 2

Literature Review

This chapter will elaborate on the importance of this study by providing clear examples of what has been accomplished so far for powering condition monitoring WSNs with RFEH. Topics include: wireless architecture, WSNs for CM, energy harvesting technologies, wireless power transfer, RFEH design parameters, as well as commercially available CM products. When considering WSNs most studies have focused on data transmission [25],[26] and much fewer have looked at power transmission. This work will focus on methods that can be used to power entire WSNs wirelessly.

2.1 WSN for CM

A WSN, illustrated in Figure 1-2, is made up of devices (nodes) that are spatially distributed and sink nodes (also called base stations) that will collect data from the nodes and transfer it to the end user via a wireless communication channel. A node is a self-contained compact device that not only integrates sensors, but also includes other units for processing and delivering sensory data. As a result, sensor nodes generate data by sensing and capturing physical properties such as temperature, vibration, or motion, and transmit this data to a base station and then to the user. Sensors, wireless communication units, microcontrollers, power management units, and power supplies are typical components of sensor nodes [27],[28]. In a monitoring application, nodes send data to the base station(s) when they detect the occurrence of events of interest. The base station can communicate with the end user in a variety of ways (e.g. ethernet, satellite, wireless, etc.) provided the bandwidth of that communications link is sufficient. For this thesis, the base station is assumed to be at a fixed location. The number and location of wireless nodes deployed in an environment and optimal network topology is influenced by the application's requirements for coverage, capacity, connectivity, cost, and network lifetime. As a result, the performance of a WSN is heavily dependent on the type of deployment. Some typical WSN models and architectures are:

- Small [29], medium [30], large WSNs: The size of the WSN varies depending on several factors, including the characteristics of the nodes, the environment, and the user's requirements.
- Homogeneous versus heterogeneous WSNs [31]: A WSN is homogeneous if all of its devices have the same capabilities (sensing, processing, communication, etc.) which result in them having the same energy consumption. A heterogeneous WSN is made up of devices with varying capacities that can be used for a variety of purposes which means that they each consumes different amount of energy.
- Stationary, mobile, and hybrid WSNs: A stationary WSN is a network made up of nodes that cannot move once they are deployed. With advancements in mobile devices, some nodes can now move on their own. This is typically accomplished by embedding the nodes within mobile platforms. A mobile WSN only has mobile device nodes, whereas a hybrid WSN has both stationary and mobile device nodes. For this thesis, it is assumed that all device nodes are stationary.

A wireless communication channel removes the need for wires to be routed from the node to the base station and allows for more flexible placement of nodes. There are several communication channels for low power nodes. The most common include Bluetooth [32], Zigbee [33], Wi-Fi [34], [35], and LoRaWAN [36].

2.2 Energy Harvesting Sources for WSNs

When considering WSNs most studies have focused on data transmission [25],[26] and much fewer have looked at power transmission. This work will focus on methods that can be used to power entire WSNs wirelessly. Often, all sensors in a WSN are powered via batteries [37], [38]. Because batteries have a limited lifetime, alternative power sources for sensors have been studied, including many types of energy harvesting methods (Figure 2-1). These include solar energy harvesting [39], magnetic field energy harvesting [40], thermal energy harvesting [41], kinetic energy harvesting [41], vibration energy harvesting [42], and wireless energy harvesting [43]. Energy harvesting is the process by which energy is extracted from external sources, captured, and stored/used for small, wireless autonomous devices, such as those used in wearable electronics and WSNs (pervasive networks) [44]. Energy harvesting devices are primarily used to collect small amounts of energy that would be lost otherwise as heat, light, sound, vibration, or movement. Typically, an energy-harvesting system is composed of three major elements: an energy source, a harvesting architecture, and the device that will use the energy harvested [45]. Energy-harvesting technology architectures can be broadly classified into two categories: harvest–use and harvest–store–use [45]. In harvest–use architectures, sensor nodes are powered directly by the energy-harvesting system [38]. If sufficient energy cannot be harvested, the harvesting node will be disabled. On the other hand, in harvest–store–use architectures, the energy-harvesting system harvests energy whenever possible and stores the energy in a storage unit for future use [46].

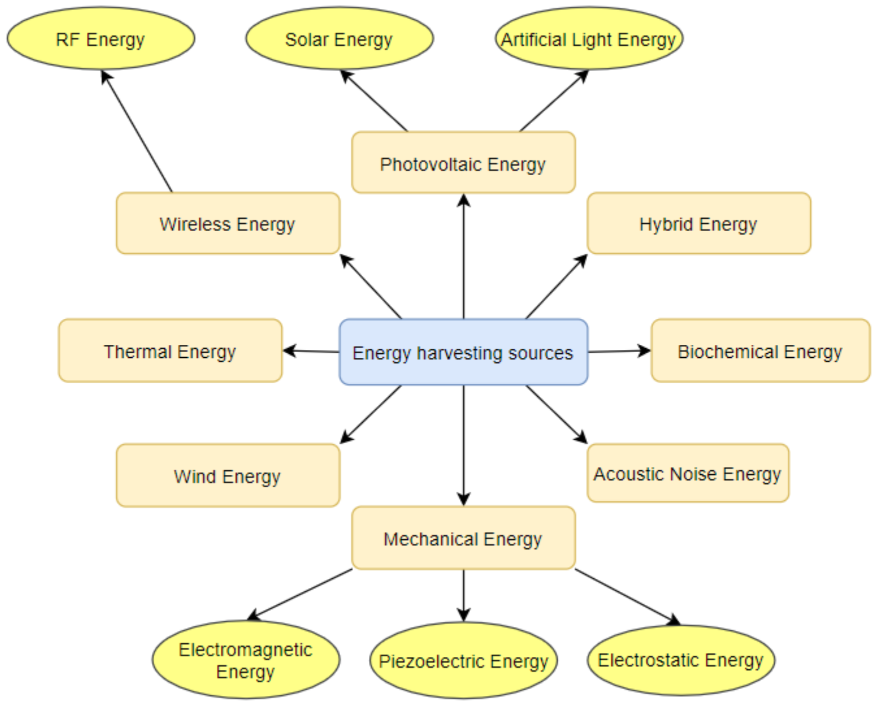


Figure 2-1. Types of energy harvesting technologies

The energy available surrounding industrial machines can be harvested separately or can be combined with other sources to form a hybrid energy harvesting system. Because of the significant amount of wasted energy in the form of heat, sound and vibration, energy harvesting has an inordinate potential for creating autonomous CM systems. Furthermore, the power densities of various types of ambient energies available within proximity to WSNs used in CM are summarized in Table 2-1.

Table 2-1. Ambient energy available in industrial locations [47]

Energy type	Specification	Power Density	Advantages	Disadvantages
Acoustic	Ambient noise	191.4 μW/cm³ @ 120 dB	Available	Unpredictable
RF	Ambient Dedicated	0.01 – 0.1 μW/cm² 1 – 10 mW/ cm²	Always available Predictable Controllable	Distance dependent
Solar	Indoor Outdoor	< 100 μW/cm² 10 – 100 mW/ cm²	Predictable High output voltage	Unavailable at night Non-controllable
Thermal	Piezoelectric	60 μW/cm³	Available	Unpredictable
Vibration	Piezoelectric Electromagnetic Electrostatic	4 – 250 μW/cm³ 300 – 800 μW/cm³ 50 – 100 μW/cm³	Controllable Low-cost design	Large size Unpredictable

Among these types of sources, RF energy harvesting has been chosen for this study because it allows sensors to be placed at locations that are difficult to access [5] and it also permits the utilization of a dedicated source which have a higher power density. Dedicated sources can be used to produce constant input power that can also be adjusted depending on the application, whereas ambient power can be unpredictable.

2.3 Wireless Power Transfer

A wireless power system consists of a transmitter device connected to a power source that converts electrical energy to a time-varying electromagnetic field, and one or more receiver devices that receives the field energy and converts it back to electrical energy that can be consumed by an electrical load. Wireless power transfer (WPT) is a broad term that encompasses a variety of different technologies for transmitting energy via physical fields. An electromagnetic field is frequently composed of two components: near field WPT and far field WPT [48]. Near field WPT is the region that is closest to an antenna. It is a solution for short-range electronic device powering and has been commercialized for a variety of wireless applications [48]. Additionally, it is often used for wireless implantable devices [36]. The near field is divided into two sections: the reactive and radiative near field regions. In the reactive near field, the electric field and magnetic field are 90 degrees out of phase with each other and are therefore reactive. The radiative near field is the region between the near and far field. Far field WPT is the region where the field tends to act normal regarding electromagnetic radiation. The field region is determined by the following:

$$\begin{cases} d \leq \dfrac{\lambda}{2\pi} & \text{near field inductive} \\ \dfrac{\lambda}{2\pi} > d \leq \lambda & \text{near field radiative} \\ d > 2\lambda & \text{far field radiative} \end{cases} \tag{2-1}$$

where λ is the wavelength in meters [49], which is related to the carrier frequency by

$$\lambda = \frac{c}{f} \tag{2-2}$$

where f is the frequency and c the speed of the light. For this thesis, only electromagnetic fields capable of transmitting power in the far field will be considered, because they allow a longer separation distance between a PTX and a PRX. The state-of-the-art for near field power transfer will be looked at as it makes up most of the charging application currently in use.

2.3.1 Magnetic Inductive Coupling

Magnetic inductive coupling uses magnetic field induction to transfer energy between two coils, from the primary coil to the secondary coil, without the use of wires. The primary energy source produces a variable magnetic field across the secondary coil. The near field process then generates a voltage and current that can be used to power devices wirelessly [19], [50]. It transfers power in the near field and over short distances (power transmitter and receiver are almost touching) (equation (2-1)), and it mostly operates at low frequencies (i.e. in kHz). Inductive coupling has been developed and used in a variety of applications, including charging electric toothbrushes, charging pads for cell phones, and charging pads for laptop computers. In [51], a WPT system was developed to charge electric vehicles overnight. Several factors can influence inductive coupling efficiency, including coil coupling tightness, which is determined by alignment and distance between the two coils. Furthermore, the quality factor, which represents the ratio of energy stored in the capacitor and inductor to energy dissipated as thermal losses, is an important parameter for optimizing efficiency by choosing the right material, shape and dimensions, and frequency operation [50].

2.3.2 Magnetic Resonant Coupling

Magnetic resonant coupling is a concept based on coils operating at the same resonant frequency and also operate in the near field. Apart from the inductive coupling technique, the primary and secondary coils in resonant coupling are adjusted to resonate at the same frequency by adding a capacitor [19], [50]. Then, high power transfer efficiency can be achieved. In comparison to inductive coupling, this technique offers significant advantages, including the ability to transfer power over slightly longer distances compared to magnetic inductive coupling, but still within the near field [19], [50]. In [52], magnetic resonant coupling has been developed for non-radiative power transfer.

2.3.3 Radio Frequency Energy Harvesting

RFEH is a technique that converts electromagnetic field energy into direct current (DC) electricity [53], [21]. There are two types of RFEH sources: dedicated RF energy sources and ambient RF sources. [18].

Ambient RF energy is typically available from public communication services such as television, GSM, wireless local area networks (WLAN), and Wi-Fi, RF energy is typically available in transmission frequencies ranging from 0.2 to 2.4 GHz [17],[54],[55],[56]. The RFEH power available from ambient RF sources varies over time, is unpredictable, and often results in a low power density. Furthermore, transmitter characteristics such as output power, antenna radiation, and polarisation are usually unknown. Therefore, the transmitter performance should be estimated first to determine the WPT system's power link budget. As a result, the receiver side should be designed to cover such unknown specifications, and its antenna should operate over a broad/multiband frequency range in order to harvest and scavenge the majority of the surrounding RF energy. [57]. Because the RFEH system must be combined with miniaturized devices, designing a harvesting receiver is difficult (requiring a multiband frequency antenna to be able to harvest energy on multiple frequencies) and trade-offs between system size and performance must be considered [58]. This might also require a bank of diodes to accommodate all energy levels harvested at different frequencies.

Dedicated RF sources, on the other hand, are on-demand supplies and can reach high power densities due to directional transmission, and can be used to power wireless nodes that require a predictable and high amount of energy [59],[60]. This technique has the advantage of knowing the RF source parameters such as the operation frequency and polarisation of the incident power when compared to ambient sources (Table 2-2). When requested, a dedicated RF energy source should be able to deliver the required power to the system. Furthermore, this power must abide by the targeted frequency band's transmitting and receiving power regulations, derived to safeguard people from harm caused by excessive levels of radiation [61].

As a result, using a dedicated RF source for the WPT system is thought to work better than ambient RF for the CM application. It enables knowledge of energy availability as well as the ability to improve the overall efficiency of the receiver. Harvesting energy from ambient sources is inherently opportunistic and unreliable. Transferring power from a dedicated energy source can be made to be consistent and to optimize and improve the amount of energy available at the receiver.

Compared to magnetic inductive coupling and magnetic resonant coupling, RFEH operates in the far field region, allowing energy to be sent over longer distances, which is the main reason for choosing RFEH for this thesis.

Table 2-2. Types of RF energy sources [58]

Section	Parameters	Dedicated RF source	Ambient RF source
PTX	Power Generator	• High-efficiency power amplifier • Optimized signal waveform	• Not controllable • Multi-tone
	DC-RF efficiency	• Can be optimized	• Cannot be optimized
	Antenna types	• Directive • Defined polarization and positions • Smart beaming is possible	• Directivity unknown • Polarization and positions unknown • Multiband source
Radio channel	RF-RF efficiency	• Known • Optimized	• Unknown • Worst case estimate
PRX	Antenna type	• Directive • Defined polarization and position • Single band	• Nondirective • Circular or dual polarization • Multiband resonator
	Rectifier	• Designed for a specific power density and RF frequencies	• Designed for a broad range of power and frequencies
	RF-DC efficiency	• Can be optimized	• Can be optimized

2.4 RFEH Design Parameters

To design a RFEH system, whether for ambient or dedicated sources, several important parameters need to be considered. These parameters include the operating frequency, the link budget, as well as PTX and PRX components.

2.4.1 Operating Frequency

In both near and far field regions, there is a wide range of frequencies that can be used to harvest energy and wirelessly transfer power. Low-frequency bands, for example, are highly valued and suitable for inductive coupling and coupling resonance operating in the near field region, such as 125-134 kHz for charging laptops and cellphones wirelessly or 13.56 MHz for powering smart/RFID cards. Far field frequencies range from a few kHz to around 30 GHz.

Because of its flexibility and low interference, the ultra high frequency band is commonly used for energy harvesting. In WPT systems, wavelength is a critical parameter that influences antenna size, attenuation level, and receiver element parameters. Low frequencies require a larger antenna, but attenuation through nearby objects is lower and can work with lower-quality circuit elements. Higher frequencies, on the other hand, have the advantage of using smaller antennas. However, attenuation is higher, and component sensitivity and signal parasite are greatly increased. WPT systems work in the ISM bands, which include 915 MHz, 2.4 GHz, and 5.8 GHz. In this work, a frequency of 915 MHz is used because of its efficiency in environments impacted by attenuation, as well as its higher range of transmission distance,s when compared to other frequencies in indoor environments.

2.4.2 Link Budget

When the PTX and PRX have a clear, unobstructed LOS between them, a free space propagation model can be used to predict the signal strength that will be received. The free space power received by a receiver antenna, separated from a radiating transmitter antenna by a distance d, is given by the Friis free space equation:

$$P_r(d) = \frac{P_t G_t G_r \lambda^2}{(4\pi)^2 d^2 L} \tag{2-3}$$

where P_t is the transmitted power, $P_r(d)$ is the received power which is a function of the distance between the PTX and the PRX, G_t is the transmitter antenna gain, G_r is the receiver antenna gain, d is the distance between the PTX and PRX in meters, L is the system loss factor, not related to propagation, and λ is the wavelength in meters [49].

2.4.3 Transmitter and Receiver

In traditional RFEH systems, RF power is extracted by receiving a RF signal via an antenna, rectified to produce the desired direct current (DC), and then used to power an external circuit or device. This rectified antenna is often called a rectenna. It is composed of an antenna, a matching circuit, and a rectifier, as illustrated in Figure 2-2.

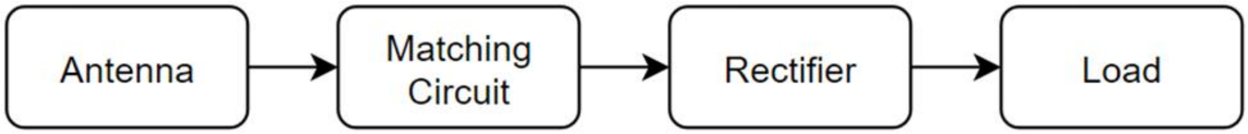


Figure 2-2. Basic rectenna components

An isotropic radiator is an ideal (and non-existent) antenna which radiates power with unit gain uniformly in all directions and is often used as a reference for antenna gains in wireless systems. The effective isotropic radiated power (EIRP) is defined as

$$EIRP = P_t G_t \tag{2-4}$$

and represents the maximum radiated power available from a transmitter in the direction of maximum antenna gain, when compared to an isotropic radiator [49]. The power density P_d at a distance d from the RF source equals the total radiated power by the transmitter divided by the surface area of a sphere scaled with the directivity of the transmitting antenna, which is the ability of an antenna to focus the radiation in a particular direction:

$$P_d = \frac{EIRP}{4\pi d^2} \tag{2-5}$$

The amount of power collected by a receiving antenna located in the far field region of the transmitter can be found by using the concept of an antenna's effective area. The antenna's effective area, A_e, is defined as the ratio of the received power, P_r, to the power density of a plane wave incident on the antenna [49]:

$$A_e = \frac{P_r}{P_d} \tag{2-6}$$

The received power is defined as the maximum power that can be extracted from an antenna and can be delivered to the load. The actual dissipated power in the antenna only equals P_r in the case of a lossless and perfectly impedance-matched antenna-electronics interface (seldom true). For RFEH, several antenna types have been studied, including dipole antennas [62], Yagi-Uda antennas, microstrip antennas [63], monopole antennas [64], and patch antennas [65].

Impedance matching networks facilitate the transfer of maximum power between the antenna and the load. It helps increase the RF signal voltage power, as well as the peak input voltage sent to the rectifier circuit [66]. This parameter is primarily affected by transmission power leakage, but also by a mismatch between the receiver antenna and the rectifier circuit [67]. Typical matching networks (also called "lossless" networks) consist entirely of reactive components, that is, components that store rather than dissipate energy [68]. This property follows naturally from the matching network's purpose, which is to maximize power transfer from source to load. If the matching network included energy-dissipating components, it would consume some of the energy being delivered to the load. As a result, matching networks employ capacitors and inductors rather than resistors [68]. The first step in the design of matching networks is to find the input impedance of the load (RF rectifier) to be matched to the output impedance of the source (antenna), typically 50 Ω [66]. Once this is accomplished, one of the matching techniques can be used to match the impedance to be matched. The L network illustrated in Figure 2-3 is one of the matching-network topologies. This term refers to eight distinct L-shaped circuits, each of which contains two capacitors, two inductors, or one capacitor and one inductor.

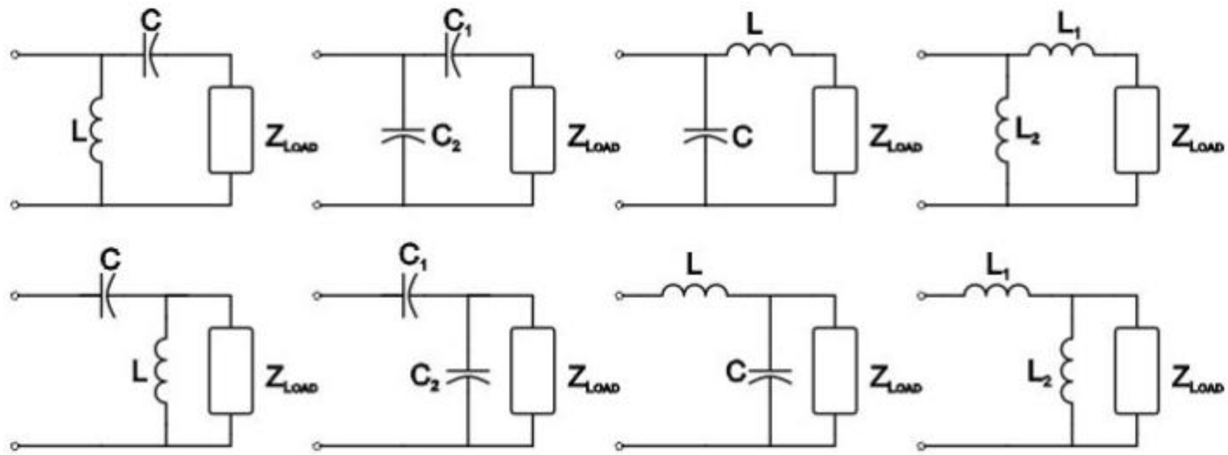
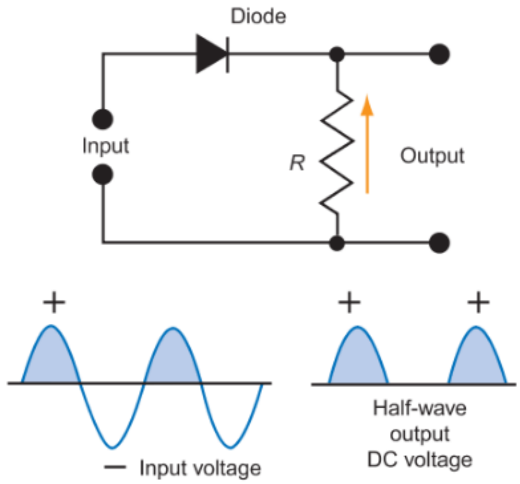
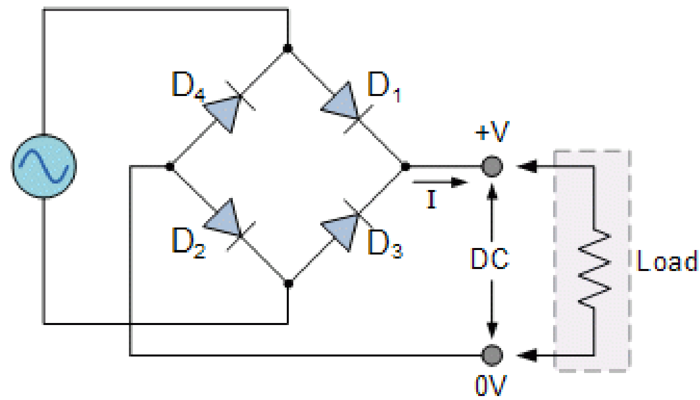


Figure 2-3. Examples of inductive matching network circuits [68]

Additional power loss occurs as a result of losses in the practical matching network, the antenna, and the rectifier implementation. A rectifier is an electrical device that uses one or more P-N junction diodes to convert alternating current (AC) to DC [69]. DC voltages are used to charge batteries and other energy storage technologies. The most frequently used rectifier circuits are diode-based, as diodes have a lower forward voltage drop than CMOS circuits. Schottky diodes are frequently used because they have a threshold voltage of approximately 0.3 V. Depending on the application, a half wave rectifier or a full wave rectifier should be selected. The half wave rectifier (Figure 2-4a**Erreur ! Source du renvoi introuvable.**) is a type of rectifier that converts half of the alternating current input signal (positive half cycle) to a pulsating direct current output signal, while blocking or losing the remaining half signal (negative half cycle) [69]. A single diode is used in the half wave rectifier circuit. A full wave rectifier (Figure 2-4b**Erreur ! Source du renvoi introuvable.**) is a type of rectifier that converts the entire alternating current input signal (positive and negative half cycles) to a pulsating direct current output signal [69]. Unlike the half wave rectifier, the full wave rectifier does not waste the input signal. Full wave rectifiers have a higher efficiency than half wave rectifiers.



(a) Half-wave rectifier



(b) Full-wave rectifer

Figure 2-4. Types of rectifier [69]

2.5 Low Power Devices for CM Powered Using RFEH

Loubet et al. proposes a prototype for monitoring temperature and humidity in harsh environments [7]. The prototype was composed of a sensing node using a LoRaWAN uplink wireless communication, a Texas Instruments bq25504 PMU, a STMicroelectronics STM32L072CZ an MCU, a Adafruit DHT22 (temperature and relative humidity) sensor and a supercapacitor of 22 mF as an energy storage unit. The energy consumption of the device was measured to be about 150 mJ. The PTX was composed of a power synthesizer (Anritsu MG3694B, operating at 868 MHz) and a patch antenna with a maximal gain of +9.2 dBi. The rectenna was composed of a dipole antenna that captures the low ambient generated electromagnetic energy field and converts it to a RF signal, as well as a single Schottky diode (Avago HSMS 2850) high-frequency rectifier that converts the guided RF signal into DC power, and a reflector to increase the antenna gain (from +2.2 dBi for a flat dipole antenna without a reflector to +6.6 dBi when a metallic reflector is added).

When processing received power of +18 dBm at 2.45 GHz, a rectenna was found to achieve a RF-to-DC conversion efficiency of 63% [56]. At a distance of 1 m and a voltage of +1.7 VDC, the rectenna was used to power a single LED. The rectenna received +7.3 dBm of energy. This equates to a 13.9 percent efficiency. The rectenna also powered a three-LED array at a distance of 75 cm and a voltage of +1.8 VDC. In this case, the rectenna's efficiency reached a maximum of up to 45.5 percent.

In [65], a proposed rectenna circuit was composed of a rectangular microstrip patch antenna, band-pass and low-pass filters, a matching circuit, and a rectifier. Because of its high-speed switching at higher frequencies and low voltage drop characteristics, a Schottky diode was used to implement the rectifier circuit. The power received by the patch antenna in the designed rectenna was -62.8 dBm, while the available power at the output port was -64.4 dBm. According to these results, the rectenna circuit was found to have an overall efficiency of 97.5 percent.

Ge et al. investigated far field battery-less wireless sensor networks powered by anticipated RF sources [70]. The system was able to detect temperature, humidity, and light with the end device powered by RF energy. At 1m, the power received by the rectenna was found to be 7.28 mW, at 2m it was 2.34 mW and at 3m, 1.18 mW. Using a different set of obstacles, the power received at 1m was 1.53mW and 1.32mW for 2-inch and 3-inch brick, respectively, and 2.88 mW for wood.

Assimonis et al. fabricated a low-cost rectenna grid and measured the RF density for a low-power RF input [71]. The design used two single series rectifier circuits that combine the currents from each diode to increase the available power at the load. For a 1 μW/cm^2 ambient power density, the rectenna-grid provided 200 μW at load, while for a 0.01 μW/cm^2 the power at load was 0.8 μW. At power inputs of −20 and −10 dBm, the RF-to-DC rectification efficiency was found to be 20.5 percent and 35.3 percent, respectively. The rectenna was connected to a DC-to-DC converter, which increased the open voltage from 298 mV to 1.4 V in 37 minutes, charging a 1 mF capacitor. The stored energy was found to be enough to supply a sensor in a wireless network with consumption of 20 mW, a rate of 1 kbps and a word-length of 20 bits.

Talla et al. developed a product that powers energy-harvesting devices while maintaining network performance [72]. This was accomplished by co-designing the harvesting hardware circuits and the Wi-Fi router transmission. It made use of a multichannel harvester to efficiently harvest energy from multiple Wi-Fi channels and generate the 1.8–2.4 V. As shown in Figure 2-5, an antenna is followed by a rectifier that converts the 2.4 GHz signal into low voltage DC power. This power is fed into a DC–DC converter that increases the voltage of the DC signal to match the voltage requirements of the sensor and microcontroller (1.8-2.4 V). The harvester's output power scales with the input power. For instance, at 2 feet the battery charging system has 100 μW available compared to 10 μW at 10 feet, as illustrated in Figure 2-6. This means that as the harvesting device moves farther from the router, it must operate at a lower duty cycle.

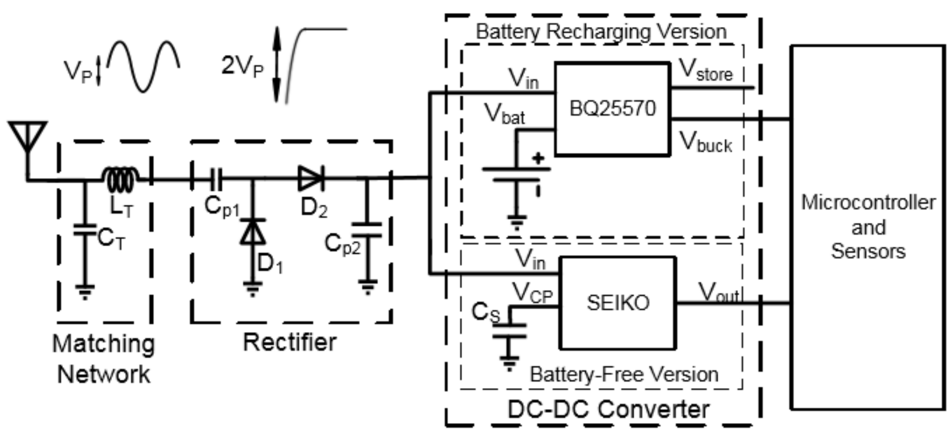
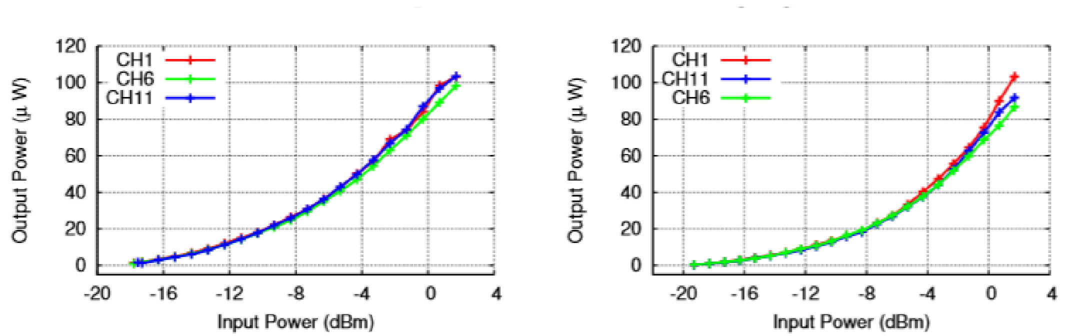


Figure 2-5. PoWiFi harvester schematic [72]



(a) Battery-free harvester *(b) Battery-charging harvester*

Figure 2-6. Available output power at the PoWiFi harvester [72]

Table 2-3 compares the parameters and result of different RFEH-based devices in the literature. At lower frequencies the power received is higher for the same PTX-PRX distance. This is the expected result from equation (**2-3**).

Table 2-3. Comparison of RFEH-based devices [58]

PTX-PRX distance	PRX antenna gain	Power received	Attenuation	Frequency	Reference
1.5 m	6.6 dBi	10 dBm (10 mW)	LOS	868 MHz	[7]
1 m	-	7.3 dBm (5.4 mW)	LOS	2.45 GHz	[56]
0.12 m	-	-62.8 dBm ($5.7e^{-7}$ mW)	LOS	2.4 GHz	[65]
1 m	6 dBi	7.28 mW	LOS	915 MHz	[70]
2 m		2.34 mW	LOS		
3 m		1.18 mW	LOS		
1 m		1.53 mW	2-inch brick		
1 m		1.32 mW	3-inch brick		
1 m		2.88 mW	wood		
-	4.4 dBi	0.2 mW	LOS	868 MHz	[71]
0.61 m	-	0.1 mW	LOS	2.4 GHz	[72]
3 m	-	0.01 mW	LOS		

2.6 Commercial Wireless CM Products

There are several products for wireless condition monitoring currently on the market. Table 2-4 compares the important specifications for each sensor. It can be seen that most commercial

CM sensors use BLE as their wireless communication technology and each device has a typical lifetime of between 2 and 4 years, depending on their utilization.

Table 2-4. Types commercial wireless CM products

Specifications	**Fluke [10]**	**ABB [11]**	**I-Alert [22]**	**Treon [23]**	**FLIR SV87 [24]**
Sensitivity range	±32 g	±2 g		±4 g	±32 g
Frequency range	10 Hz – 1 kHz	1 Hz – 1.6 kHz	3 Hz – 5 kHz	10 Hz – 6.3 kHz	10 Hz – 1 kHz
Sampling rate	25600 Hz	–	–	26700 Hz	–
Temperature range	-30°C to +80°C	-30°C to 85°C	-40°C to 84°C	-40°C to 105°C	-30°C to +80°C
Size	61.5 x 24 mm	–	–	78.5 x 28 mm	65 x 25 x 27 mm
Weight	40 g	–	–	129 g	62 g
Battery type	3.6 V, 2400 mAh lithium	3.6V LiSOCl2 cells	Lithium battery	3.6V lithium thionyl chloride	3.6 V lithium battery
Battery life	3 years typical	2 years typical	3/4 years typical	Up to 6 years	4 years typical
Wireless Communication	BLE 4.1	BLE	BLE 5	Wirepas Mesh	BLE 4.2
Frequency range	2405 MHz to 2480 MHz	2402 MHz to 2480 MHz	2.45 GHz	2.45 GHz	2.45 GHz

2.7 Mobile RFEH-based Power Architectures for WSNs

As discussed, powering a single wireless node has been shown to be feasible using batteries or even when using wireless power over reasonable distances. Nonetheless, powering multiple nodes simultaneously and efficiently in a WSN in harsh and hard-to-access environments, or when direct LOS is not available, can be challenging. To solve this challenge, it is necessary to bring the transmitter closer to the receiver or vice versa. In this thesis, the WSN is stationary, meaning that each receiver is fixed at known locations. With that criterion, researchers have proposed different ways of moving the wireless transmitter.

Xie et al. used a mobile charging vehicle that travels through the sensor network on an optimal path and wirelessly charges each sensor node's battery [73]. This mobile wireless charging vehicle (WCV) can be either human-operated or completely self-contained. The Hamiltonian cycle determines the best path for the WCV to charge the sensor nodes, as shown in Figure 2-7.

Xie et al. then used multi-node wireless energy transfer technology to recharge multiple sensor nodes in WSNs and addressed energy issues [74]. A WCV was used to replenish the energy supply of sensor nodes. The WCV travelled inside a wireless sensor network on a regular basis and wirelessly charged the sensor nodes. The charging range of the WCV was used to propose a cellular structure. This structure divides the two-dimensional plane into hexagonal cells that are adjacent to each other. The WCV comes to a halt in the centre of a cell, allowing all sensor nodes in the cell to be charged at the same time.

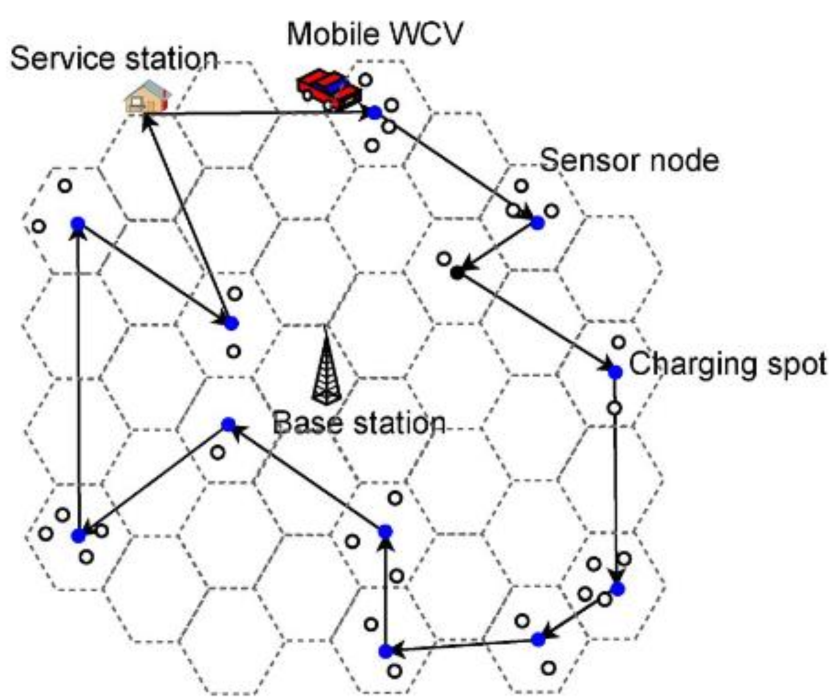


Figure 2-7. WCV trajectory based on Hamiltonian cycle [73]

He et al. proposed an on-demand mobile charging technique in which individual sensor nodes request charging from the mobile charger when their energy levels fall below a certain threshold [75]. They used a simple and efficient nearest-job-next with pre-emption (NJNP) discipline for the mobile charger to analyse the on-demand mobile charging problem. When the node's residual energy falls below a certain threshold, it will send a charging request to the mobile charger. With NJNP, the mobile charger will fulfil the requests that have been received. The mobile charger's energy capacity can be replenished at the energy tank in the centre of the area.

In each case, researchers used different approaches to charge individual device nodes. In [73] and [74], optimal charging paths were determined via the Hamiltonian cycle. This method is efficient when travel time between nodes is an important part of the system. In [75], a nearest-job-next strategy with pre-emption is used, where each node sends a signal to the PTX when in need of being recharged. This method is efficient when each node has a different energy consumption and discharge time cannot be predicted. Finding an adequate charging strategy for nodes is dependent on two main criteria: the amount of time a node takes to be charged and the travel time between nodes. These criteria help in choosing the appropriate charging strategy for mobile architectures. In this work, the charge cycle is known, (i.e. the PTX will know which node to charge next and for how long). This is because the nodes used in this thesis are homogeneous and the energy consumption, as well as the discharge time, can be determined.

Chapter 3

WSN Power Architectures Using RFEH

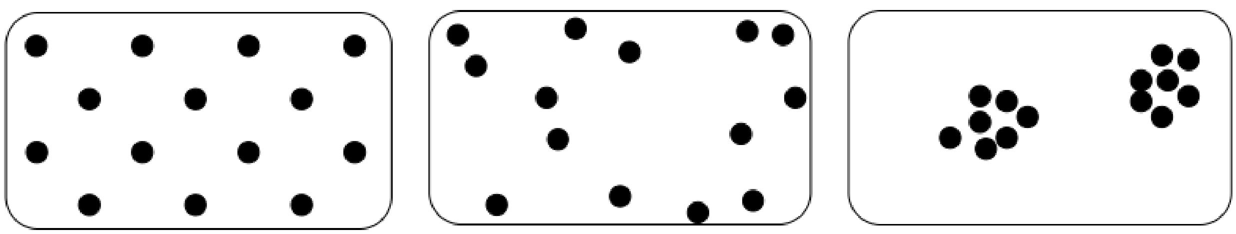
In the first chapter, the objectives of this thesis were described, and the thesis question was posed regarding RFEH for powering CM devices in a WSN. Chapter 2 provided an overview of the theory underlying RFEH, as well as all the parameters that needed to be considered in attempting to answer the thesis question. This chapter describes the behaviour of RFEH systems. Several inputs found in typical applications will be studied, including node distribution, PTSA, and node density. This chapter will also propose different power architectures for providing energy to multiple sensor nodes while observing key system characteristics. The impact of transmitter antenna choice in a power network is considered and then several ways of transmitting energy to nodes in non-line-of-sight (NLOS) environments are proposed.

3.1 Node Type

In this work, the WSN considered will be homogenous, meaning that all nodes in the network will consume the same amount of energy based on node components and duty cycle being the same for all nodes. Additionally, the nodes will be stationary meaning that they will be at a fixed location during their entire lifetime. More information about the nodes themselves will be provided in Chapter 4.

3.2 Node Distribution and Density

Node distribution represents the way that nodes are placed in an environment. There are three basic types of node distributions that will be analyzed in this study: uniform pattern, random pattern, and clumped pattern, as shown in Figure 3-1. Node density is the number of nodes within a given environment, measured in nodes/m^2. While the diagrams only show 2D node distribution patterns, real nodes can be distributed in 3 dimensions.



(a) uniform pattern (b) random pattern (c) clumped pattern

Figure 3-1. Representation of different distributions

3.3 Power Transmitter Signal Attenuation

Objects in industrial environments often have obstructed paths between them. This means that devices in a WSN could be in areas with no clear LOS for RFEH. In a wireless energy-harvesting environment, where collected energy determines network lifetime, this is important. PTSA varies with the intervening material and thickness. Table 3-1 provides the typical signal attenuation values caused by various materials (assuming common thicknesses).

Table 3-1. Attenuation caused by different types of materials [76]

Material	Attenuation (dB)
Human body	3 – 10
Glass window	2
Wood	3
Brick	3 – 7
Metal frame clear glass wall	6
Concrete wall	12 – 35

IoT devices for machine CM are frequently installed near rotating shafts or bearings located behind or inside the machines that are being monitored [4]–[6]. If the energy sent by a power transmitter is insufficient, it will not reach the receiver and, thus, the device will not be powered. The power before attenuation (P_o) can be found using the equation for PTSA (**3-1**).

$$PTSA = 10\log\left(\frac{P_o}{P_r}\right) \tag{3-1}$$

where P_r is the power received by the device.

Environmental PTSA is one of the factors that can have the largest negative impact on the system's efficiency. To avoid attenuation, researchers have applied the principle of time reversal for wireless power transfer [77]. The time reversal technique, illustrated in Figure 3-2, adjusts the beam transmission to avoid objects in its path, using a phased set of operations. This begins with the acquisition phase, in which a receiver joins the system by broadcasting or emitting a unique signal. The signal received by the transmitter will contain spatial information that is specific to the receiver's location. This signal is then reversed in time, amplified, and broadcast back into the environment. On the receiver, the amplified signal is reconstructed and converted to usable DC power by the rectifier. A small portion of the signal is used to rebroadcast a new characteristic signal into the environment, which is collected in the signal of the next receiver, and the cycle is repeated, with the result that power is focussed more optimally on the receiver [77].

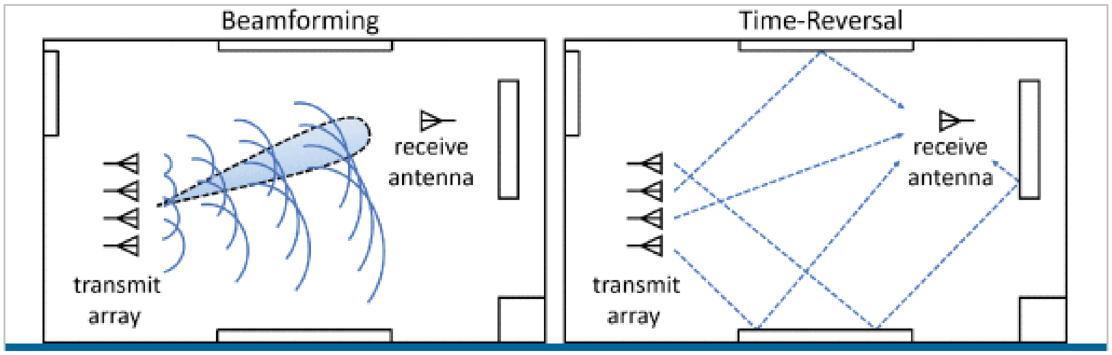


Figure 3-2. Beamforming wave propagation vs time reversal wave propagation [76]

Unfortunately, time reversal has yet to be used for wireless power transfer in practice. Nonetheless, this technique provides an interesting avenue for adapting to the constraint of material attenuation and allows nodes to receive optimized amounts of energy from the transmitter.

3.4 System Cost Efficiency

In this work, the lifetime of the system is defined as the time it will take all nodes in a WSN to die because of lack of power. Because the WSN is homogeneous, all nodes should consume the same amount of power and die at the same time. The system cost efficiency (SE) in equation (3-2) is defined as the cost of powering each node (n) for the entire lifetime (LT) of the system, or cost per node per year. This parameter indicates the economic viability of implementing a system. The system's total cost of implementation includes labor costs and capital costs. Labor costs are expressed in terms of time that a worker actively works towards the functioning of the system. This includes moving a PTX within the factory to charge nodes or replacing nodes (L_{cost}). Capital costs include the costs of the physical components required to implement a strategy (C_{cost}), the cost of a PTX (PTX_{cost}), the cost of energy to run the system (E_{cost}) and the cost of monitoring devices ($node_{cost}$).

$$SE = \frac{\sum cost}{n \times LT} \tag{3-2}$$

3.5 PTX Antenna Type

When considering RFEH, the transmitter's antenna emits energy, and the receiver's antenna collects that energy, before distributing it to the rest of the IoT device's components. Knowing the energy consumption of a node will help determine the maximum feasible distance that can be safely (according to regulations) achieved between a power transmitter and receiver for a device node to operate [55]. The radiation pattern of a transmission antenna represents the emission or reception of wave fronts, indicating the strength of the antenna in a certain direction [78]. The idealized, uniform, omni-directional pattern used by omni-directional antennas and the pencil-beam pattern used by directional antennas are two of the most common radiation patterns.

Omni-directional radiation or isotropic radiation is the radiation from a point source that radiates uniformly in all directions and has the same intensity regardless of measurement direction [78]. However, in practice, true isotropic radiation is impossible to achieve because every antenna radiates energy with some degree of directionality. Among existing antennas, dipole antennas (Figure 3-3) have the closest radiation pattern to isotropic radiation [79]. This type of radiation allows IoT devices to be placed anywhere within the region covered by the transmitter.

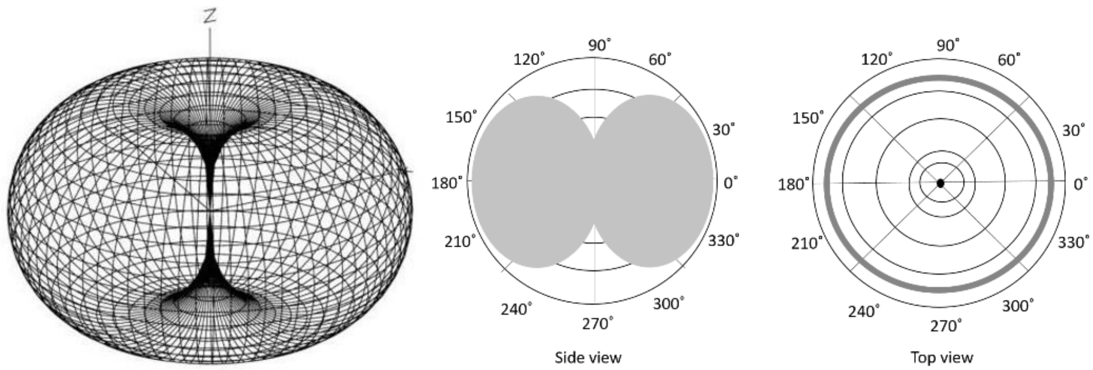


Figure 3-3. 3D and 2D representations of an omni-directional antenna radiation pattern [78]

The pencil beam pattern (Figure 3-4) is a highly directional antenna pattern, where the radiation intensity is focused in a specific direction, while transmitting or receiving [78]. The beam of a directional antenna is propagated over a greater distance. With this method, it is possible to send more energy, for a specific transmission distance, to specific points (and less energy to others). This non-uniform antenna gain allows certain IoT devices to be placed further away from the transmitter source. On the other hand, the beam width, which is the aperture angle through which most of the power is radiated, is reduced. The number of devices that can receive energy from this type of antenna decreases as the distance between the PTX and the PRX increases and the beam width decreases. The angular separation at which the magnitude of the radiation pattern decreases by 50% (or -3dB) from the peak of the main beam is known as the half power beam width ($HPBW$):

$$HPBW = \frac{70\lambda}{D} \tag{3-3}$$

where λ is the wavelength and D is the diameter of the beam. By knowing, the HPBW angle, the distance d between the transmitter and the node, and the distance c between the half power point left and the half power point right (equation (3-4)), the region of coverage (Figure 3-5) can be determined.

$$c = \frac{\sin(HPBW)}{4\sin\left(\frac{180 - HPBW}{2}\right) \cdot \cos\left(\frac{HPBW}{2}\right)} \tag{3-4}$$

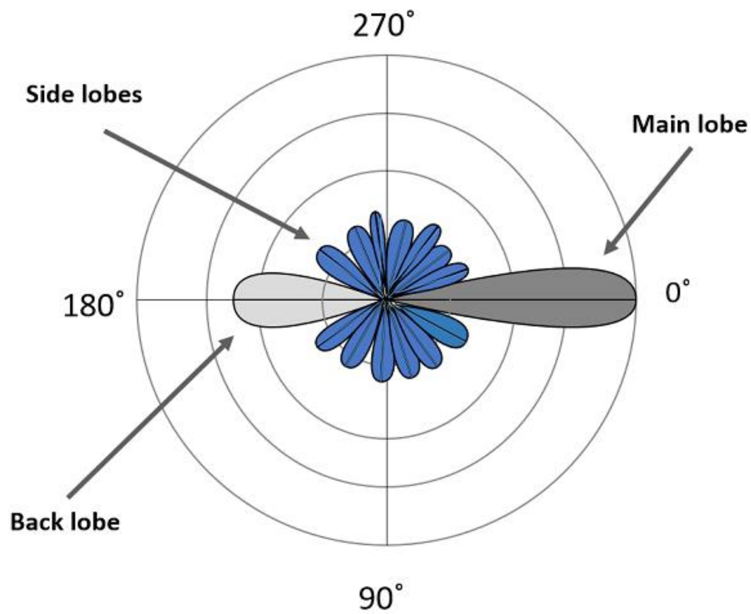


Figure 3-4. Representation of a pencil beam radiation pattern [78]

In the case of multiple IoT devices, the antenna's directionality could be adjusted so that the beam can cycle between devices and provide power to each for the required amount of time. This adjustment can be accomplished by physical movement of the transmitting antenna or by using electrical methods. Physical movement of the transmitting antenna [80] allows multiple devices to be placed in the optimal beam path, using a single mobile antenna. This requires a mechanical system to move the transmitter.

Alternatively, an electronically steerable antenna allows the beam’s transmission direction to be changed. An electronically steerable antenna [81],[82], can be implemented using phased arrays of multiple antennas, each fixed physically (Figure 3-6). All of the antennas in the array are controlled by a computer to produce a single beam of radio waves that can be directed to or focussed in different destinations, without requiring antennas to be physically moved.

A phased array antenna is a type of array antenna in which single radiators can be fed with varying phase shifts to "shape" the transmitted signal. This allows electronic steering of RF signals and is much more flexible than mechanical steering methods. This antenna's principle is based on the effect of interference, which is a phase-dependent superposition of two or (typically) several radiation sources. In-phase signals amplify each other, while counter-phase signals cancel each other out. When two radiators emit a signal with the same phase shift, a superposition occurs: the signal is amplified in the main direction and attenuated in secondary directions. Depending on the environment, antennas that are physically moved to direct their beam can also be combined with electronically steerable antennas to improve antenna directivity and further increase system performance.

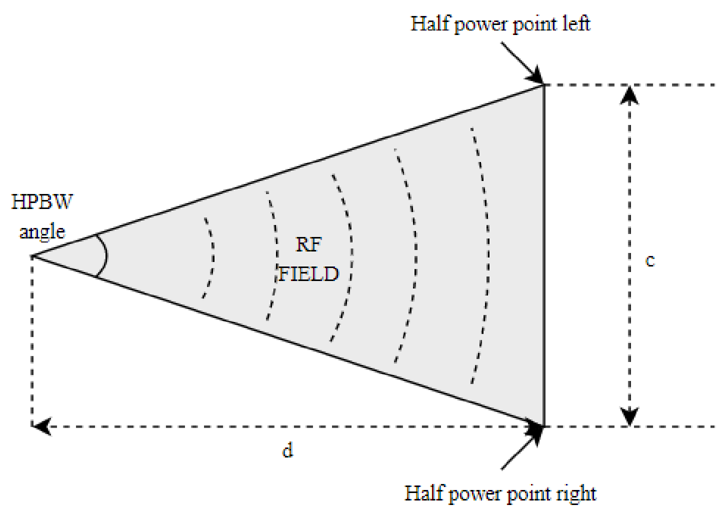


Figure 3-5. Region of coverage (in grey) for a directional antenna

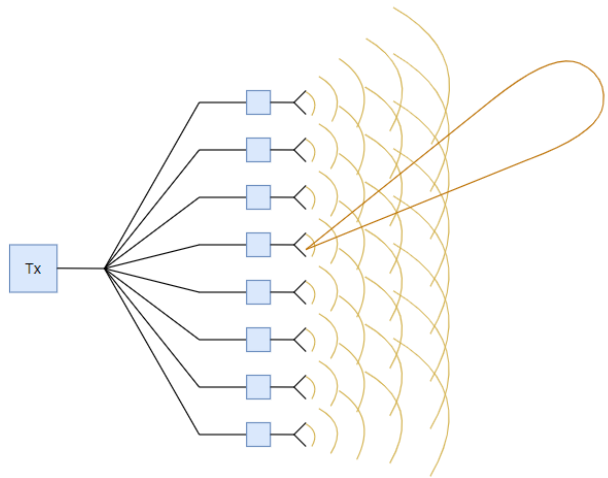


Figure 3-6. Representation of a phase array

Combining several PTXs in an indoor setting can cause interference. Waves from two or more PTXs may reinforce each other in some directions or cancel each other in other directions. In a worst-case scenario, this interference can result in spectral peaks and nulls, creating a sort of RF "minefield", unless adaptive techniques are used to even out the power levels, spatially. Phased arrays might be used to control the polarization and directivity of a bank of antennas, but this requires some kind of sensing of the field at different locations, followed by the appropriate changes to the array transmitters.

3.6 Energy Transfer Strategies for Sensor Networks

This section introduces the types of architectures (Figure 3-7) that will be used to power nodes in WSNs. Furthermore, an analysis of these concepts used with the parameters reviewed in the previous sections is conducted.

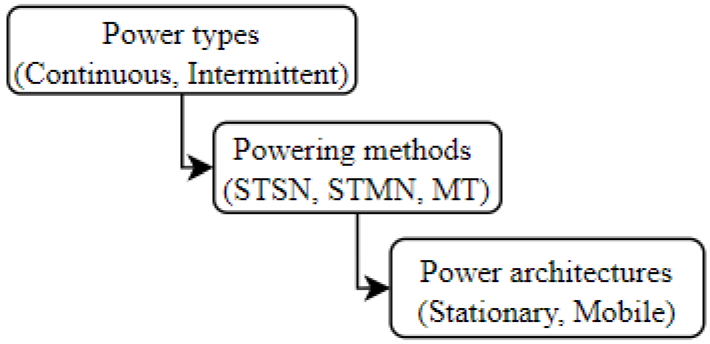


Figure 3-7. Representation of energy transfer strategies for a WSN

3.6.1 Power Types

There are two power types that will be studied in this work: continuous and intermittent. The continuous power type happens when a node constantly receives enough power from a PTX, so that no energy storage mechanisms is required (harvest-use). There is no interruption of energy distribution, and it is possible for the node to rely solely on that energy. The intermittent power type happens when a node does not constantly receive power from the transmitter, but is instead powered at specific instances. This type can occur periodically (i.e. at known and periodic intervals) or aperiodically (i.e. without following any specific pattern). To use intermittent powering, the node is required to have an energy storage unit that will store energy and power the device when energy from the PTX is not available (harvest-store-use).

In RFEH, these power types can be applied using three different strategies:

- Single transmitter single node (STSN): A node is powered/charged by one power transmitter (Figure 3-8**Erreur ! Source du renvoi introuvable.**a).
- Single transmitter Multiple nodes (STMN): Multiple nodes are powered/charged by one power transmitter (Figure 3-8**Erreur ! Source du renvoi introuvable.**b).
- Multiple transmitters (MT): A node is powered/charged by multiple transmitters (Figure 3-8**Erreur ! Source du renvoi introuvable.**c).

Finally, these powering strategies can be implemented in different types of power architectures. Two architectures are considered for powering multiple devices using RFEH: stationary transmitter architecture and mobile transmitter architecture.

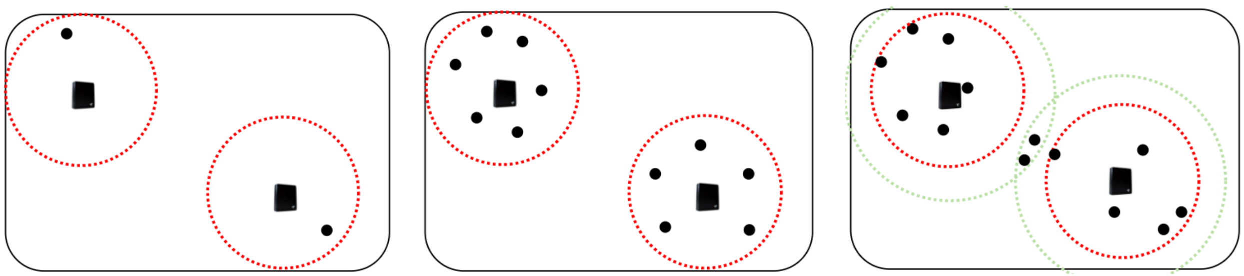
3.6.2 Stationary PTX Architecture

In this architecture, PTXs are fixed to pre-defined locations. The objective is to always maintain a power supply to all nodes, minimizing the number of PTXs that are required to achieve this objective. This architecture can use both continuous and intermittent powering techniques. When using continuous powering, nodes are constantly and directly powered by the PTX (i.e. within range). Using intermittent powering for this architecture will result in nodes being powered intermittently, forcing them to draw energy from their storage, if they have one and if they need to operate in the absence of supplied wireless power.

When using STSN in a stationary architecture, each node has one dedicated PTX to power it. The nodes must be spaced far enough apart so that each node is only covered by one PTX region. In this case, the location of each PTX will be determined by the maximum allowable distance to the nodes that it is powering. This strategy results in having an equal number of PTXs and nodes in the system.

Using STMN in a stationary architecture implies powering a group of nodes simultaneously by a single PTX. A group is made up of multiple nodes that are distributed in such a way that they can all be found within the radiation field of each PTX unit. In this case, the location of each PTX will be determined with the goal of reducing the total number of PTXs, while still powering all nodes in the network. The number of PTXs in this strategy is determined by the locations of each node and the number of groups or clusters that are formed.

Using MT in a stationary architecture implies powering one node with multiple PTXs. This strategy uses the aggregated energy transmitted by multiple PTX units to power a node that would otherwise be outside the charging range of a single PTX. Figure 3-8**Erreur ! Source du renvoi introuvable.**c shows an illustration of this strategy, where the intersection of the green circles is the aggregation of power that is sufficient to power a node. This strategy will not be simulated in this thesis.



(a) STSN (b) STMN (c) MT

Figure 3-8. Representation of different strategies using omni-directional antennas

3.6.3 Mobile PTX Architecture

Nodes can be powered intermittently or continuously using a mobile transmitter architecture. With mobile intermittent powering, a PTX charges a node for a specified period of time before travelling to another node and repeating the process. Because nodes in this architecture do not always have access to energy provided by the PTX, they must have an energy storage unit to store sufficient energy to function until the next charging cycle or be non-operable while waiting. With mobile continuous powering, nodes continue to receive power from some PTX at all times. This either restricts the mobility or requires over-provisioning of PTX units. If operation is required in intermittent power cases, an energy storage unit must be charged periodically, making charging time an important parameter. In summary, the number of PTXs required in mobile transmitter architectures depends on the trajectory of the PTX, the energy required by each node, the displacement speed of the mobile PTX, and the time required to charge each node.

When using mobile STSN, a single PTX can charge a single node before travelling to the next node for charging. A single PTX can charge several individual nodes, provided it recharges previous nodes before they discharge. As a simple example of this, a single mobile PTX could power both nodes, provided it visited them often enough. The distance between a PTX and a node depends not only that node's position, but also on the position of the next node in the PTX's trajectory. Charging time is decreased, when a PTX is closer to a node, because the received wireless power varies with the square of the distance between them (equation (**2-3**)). However, placing the PTX close to a specific node might also increase the travelling distance to another node. This trade-off between the time required to charge a node and the time required to travel to the next node affects the optimal mobile PTX trajectory.

STSN is more suitable to situations where nodes are uniformly spaced, such that powering multiple nodes with a single PTX is not possible. When focusing on a single node, directional antennas are advantageous because they radiate more energy in a concentrated direction, allowing the PTX to be placed further away from the node. When attenuation is high, STSN is more appropriate because the PTX can be placed in a location that allows a receiver to receive sufficient power, despite the attenuation. However, this can be cost prohibitive.

When mobile STMN is used, a single PTX can charge a group of nodes, before travelling to another group of nodes to charge them too. Since there is only one PTX in a STMN architecture, charging several node groups requires the PTX to return to charge previous groups often enough to prevent any particular node in a group from being fully discharged. Each node in the group will charge at a different rate, depending mostly on the distance to the PTX. Also, the discharge rate of each node might vary too, depending on the application. Therefore, the ideal PTX movement depends on the node that charges at the slowest rate (i.e. the 'farthest' node from the PTX, in integrated terms), and also on the discharge rates of each node. In the worst case, a node would discharge fast and also be the most 'distant' from a charging perspective.

STMN is more suitable when multiple nodes are clumped, as it enables a single PTX to power those nodes simultaneously. Both omni-directional and directional antennas have their advantages and disadvantages, which is determined by the node locations within a cluster. If nodes are arranged in a circular or circle-like patterns, for example, an omni-directional antenna is preferable. When nodes are aligned in more of a straight line, directional antennas would work better.

When using MT, multiple PTXs can be combined to charge nodes, prior to travelling to charge other nodes. As with the stationary architecture, the aggregated energy transmitted by multiple PTXs is used to power a node that would otherwise require a separate PTX.

A third architecture could involve combining stationary and mobile architectures to create a more efficient system. The environment could then be divided into zones, each with their own set of parameters. Suppose that an environment is divided into 2 zones, A and B. For zone A, using stationary architecture is more efficient than using mobile architecture, and for zone B, using mobile architecture is more efficient than stationary architecture. Instead of using a single architecture for the entire environment, the most efficient architecture in a zone can be used for each zone. In this case, stationary architecture will be used in zone A and mobile architecture will be used in zone B.

3.7 Methodology

To select adequate parameters, all user inputs must be defined. Subsequently, a combination of antenna type and power architecture must be determined. There are three types of inputs that can be considered. The first type of inputs are those defined by the user. This includes node distribution, number of nodes, and the desired cost efficiency range for the system. Based on the node distribution, the location of all nodes (locations where sensors need to collect data), as well as the total number of nodes that are needed, is determined. With the desired cost efficiency range for the system, the user determines their ideal cost range. Knowing this cost will help determine if the proposed solution is satisfactory to the user. The second type of inputs are those that help design other inputs/components necessary for the system to work. This includes the amount of data required to perform CM, and the desired node lifetime. This data will help determine the energy consumption of each node, and its energy supply type. The energy consumption of a node can be determined using the analysis detailed in Chapter 4. The third type of inputs are those determined by the environment. This includes the size of the plant, and the PTSA. The size of the plant can be combined with the number of nodes to form the node density. The PTSA will help determine the power attenuation level between the PTX and any given node.

Once inputs are known, system parameters can be adjusted to determine which mix of PTX antenna types and power architectures will satisfy user requirements. This is done by comparing the results of all combination (i.e. stationary architecture and omni-directional antenna (SO), stationary architecture and directional antenna (SD), mobile architecture and omni-directional antenna (MO), and mobile architecture and directional antenna (MO)). These combinations of PTX antenna types and power architectures will help determine the final PTX locations in a given environment. To find the final PTX locations, an algorithm defined in Chapter 5 will consider all antenna and architecture combinations via a series pre-defined steps. Final PTX locations will then enable a comparison of STSN, STMN and MT strategies when used for the system.

In general, to select the appropriate strategy and determine the final PTX locations in a given environment, the following steps need to be performed:

Step 1	Determine all three types of inputs (user inputs, designer inputs, and environmental inputs)
Step 2	Using the information in Chapter 4, determine node characteristics (energy consumption, energy supply type)
Step 3	Using the algorithm described in Chapter 5, determine the final PTX locations
Step 4	Compare the results in step 3 with the user's desired system cost efficiency to validate the system's parameters

An overview of the steps required to implement a WSN using RFEH is illustrated in Figure 3-9.

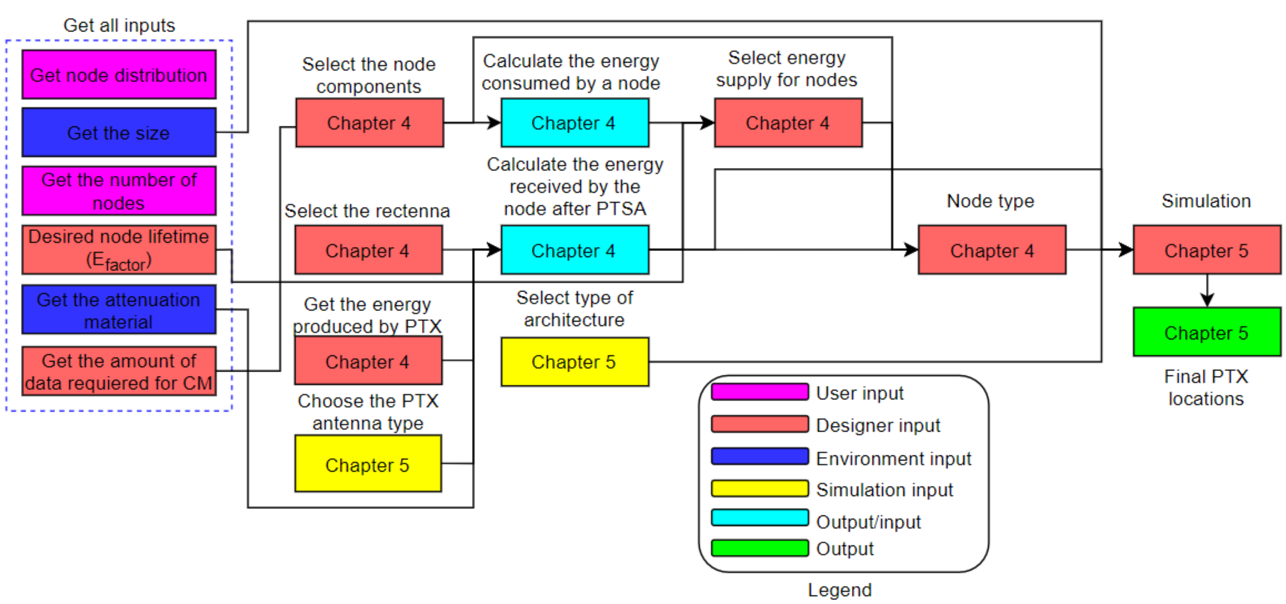


Figure 3-9. Flow chart of steps to implement RFEH-based WSN

3.8 Summary

This chapter provides an overview of all the inputs and parameters that need to be taken into consideration when implementing a WSN using RFEH. The inputs that are defined by the user are analyzed, including the node type, the node density, the node distribution, and the PTSA. These inputs will affect the behaviour of the system and the values of parameters required to attain desired results. The parameters considered for this work includes the PTX antenna type and the power architecture. PTX antennas have been divided into two types: omni-directional and directional antennas. Depending on the environment, one antenna type may be more efficient than the other. Power architectures have also been divided into two types: stationary and mobile. Both architectures can be implemented using the defined powering strategies (STSN, STMN, MT). By combining the PTX antenna type and the power architecture, the final PTX locations for a given environment can be determined by following the steps in Figure 3-9.

Chapter 4

Low-Power Wireless CM Devices

This chapter analyzes the components of a typical CM device to reduce the total amount of operation energy required making the device greener. A design framework for selecting the necessary components will be proposed alongside a study on the impact of RFEH to provide wireless power to condition monitoring devices. Different types or combinations of energy sources will be studied, with an overall goal of determining how different technologies can increase the lifetime of wireless CM devices and the maximum allowable distance between a PTX and a node.

4.1 CM Device Components

The purpose of this section is to provide a general understanding of how the parameters of each component affects the device's total energy consumption. To perform CM, the device might only need to collect data at intervals, rather than continuously, on the monitored equipment [11]. Therefore, the CM device can have two basic power states: active and standby. The device will sleep (standby) most of the time and wake up (active) periodically at scheduled times.

However, not all components have just these two basic modes of operation or there may be considerable variation in power consumption, depending on the device design. For example, each component might also have different operating modes (e.g. transmission mode for the wireless component) with varying power consumption levels. Despite this, components are defined as being used in only active and standby modes for this study. The time it takes for components to switch modes is included in standby mode. For this work, the values that will be considered are for cases where the device is in active mode (t_a) for 1, 10 and 60 seconds every hour. This defines a device active/standby "duty cycle" (high percentage values indicate mostly active operation). Comparatively speaking, active mode current requirements are about three orders of magnitude larger than standby power requirements (i.e. measured in microamps, not milliamps). Consequently, energy harvesting methods are less useful when duty cycles are very high. Therefore, we assume that application functions can still be fulfilled with low duty cycle values.

Typical wireless CM devices are comprised of a power management unit (PMU), a microcontroller (MCU), a wireless communication component, a sensor, and an energy source (

Figure 4-1). By carefully selecting each component, the total energy consumed can be reduced. When considering a wireless device powered by RFEH, other components are added (e.g. a power receiver). Each of these components is discussed in the following subsections. The detailed specifications of each component used is provided in Appendix A.

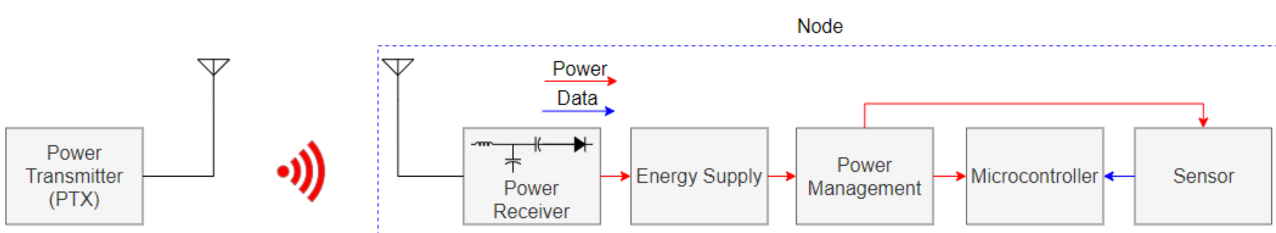


Figure 4-1. Block diagram of a PTX and the node components

4.1.1 Microcontroller (MCU)

The MCU controls the data collection process. It instructs the sensor when it is time to collect data and then stores collected data before transferring it for analysis. The MCU stores this data in non-volatile flash memory (or low-power RAM that can retain its contents in standby mode). The rate at which data arrives and must be stored in memory is determined by the analog to digital converter (ADC) conversion bit rate and resolution, which is usually related to the sampling requirements of the application. Therefore, the flash memory must be sized to store enough bits, while operating in isolation from computers with larger storage capacities that can download the stored data, allowing it to be erased.

This data must be stored fast enough to keep up with the ADC data rate, a product function of the ADC bit rate (ADC_{br}), which is determined by the required sampling rate (s_r) for the data being monitored in the application, and the ADC bit resolution (ADC_{bre}). A wireless communication unit allows for more flexible placement of devices than when wires are used for the purpose. Bluetooth low energy (BLE) is a common radio communication technology for low-complexity/power-efficient MCUs [83]. The latest version of the Bluetooth low energy protocol has low energy consumption and high data transfer rate, compared with previous versions [32]. Other technologies could also be used (e.g. Zigbee or Wi-Fi) [33]. Figure 4-2 gives a comparison of energy consumption for different wireless technologies. Based on this, Bluetooth low energy shows the most promise, from a power perspective, albeit with lower data rates and wireless range.

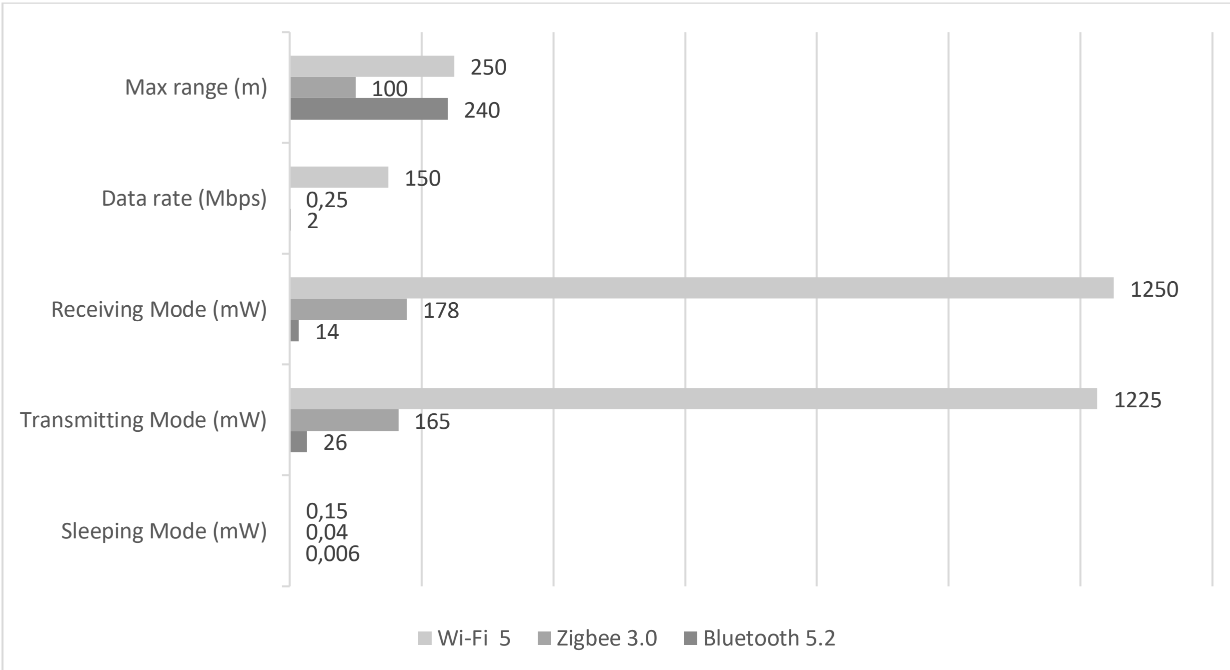


Figure 4-2. Energy consumption comparison for Bluetooth, Zigbee and Wi-Fi [32], [33]

Table 4-1 provides the important specifications for MCUs that possess the capabilities required for this study.

Table 4-1. Important MCU specifications

Parameters	Description
V_{MCU} (V)	MCU voltage
I_{fw} (mA)	Flash program current (mA)
I_{MCU_a} (mA)	Active mode current (mA)
I_{MCU_s} (mA)	Standby mode current (mA
I_{wr} (mA)	Bluetooth PTX current (mA)
ADC_{br} (bit/s)	ADC bit rate
ADC_{bre} (bit)	ADC bit resolution
E_{MCU_a} (μWh)	Non-wireless energy consumption in active mode
E_{MCU_s} (μWh)	Energy consumption during standby mode
E_{wr} (μWh)	Energy consumption due to wireless communication
r_{dr} (Mbit/s)	Rate at which data is being sent to base station
s_r (sps)	ADC sampling rate
t_{fe} (h)	Time it takes the MCU to erase data in the flash
t_{fp} (h)	Time it takes to write a word in the flash memory
t_{fw} (h)	Amount of time to write in the flash
t_{MCU_a} (h)	Total time spent by the MCU in active mode
t_{MCU_s} (h)	Time spent in standby mode every hour by the MCU
t_{ra} (h)	Duration of data transmission to base station
w_l (bit)	Flash memory word length

The following equations are used to determine the energy consumed by the sensor over one hour of operation (E_{MCU_s}):

$$ADC_{br} = \frac{s_r t_a}{ADC_{bre}} \quad (4\text{-}1)$$

$$t_{fw} = \frac{ADC_{br}}{w_l} t_{fp} \quad (4\text{-}2)$$

$$t_{ra} = \frac{ADC_{br}}{r_{dr}} \quad (4\text{-}3)$$

$$t_{MCU\_a} = t_a(h) + t_{fw}(h) + t_{ra}(h) + t_{fe}(h) \quad (4\text{-}4)$$

$$E_{MCU\_a} = (I_{MCU\_a} + I_{fw}) t_{MCU\_a} V_{MCU} \quad (4\text{-}5)$$

$$E_{wr} = I_{wr} t_{ra} V_{MCU} \quad (4\text{-}6)$$

$$t_{MCU\_s} = 1h - t_{MCU\_a} \quad (4\text{-}7)$$

$$E_{MCU\_s} = I_{MCU\_s} * t_{MCU\_s} * V_{MCU} \quad (4\text{-}8)$$

where all equation variables are defined in Table 4-1. Table 4-2 provides examples of MCUs that possess the required capabilities. The EFR32BG22 is chosen as the MCU for this study because it not only performs all the necessary tasks for this application, but it is one of the most energy-efficient MCUs currently on the market. It also includes the latest Bluetooth 5.2 networking chip for IoT devices. All other specifications of the MCU are shown in Appendix A**Erreur ! Source du renvoi introuvable.**. Simulated results for different active mode durations are tabulated in Table 4-3.

Table 4-2. General specifications of various MCUs [84], [85], [86]

Parameters	EFR32BG22 (2020)	QN9022 (2018)	STM32WB55xx (2021)
V_{MCU} (V)	3.3	3.6	3.6
I_{MCU_a} (mA)	0.64	1.35	5
I_{MCU_s} (mA)	0,00194	0.003	0.74
I_{fw} (mA)	1.45	1.35	3.4
I_{wr} (mA)	8.5	8.8	12.7
r_{dr} (Mbit/s)	2	1	2
t_{fw} (s)	0.0000456	-	0.0000908
t_{fe} (s)	0.0144	-	0.0245
w_l (bit)	32	32	64

Table 4-3. Simulated values for the EFR32BG22 microcontroller

Parameters	Value		
V_{MCU} (V)	3.3		
s_r (sps)	76800		
t_a (s/h)	1	10	60
t_{MCU_a} (s/h)	1.02	10.11	60.57
E_{MCU_a} (µWh)	2	18.3	109.9
E_{MCU_s} (µWh)	6.4	6.4	6.3
E_{wr} (µWh)	0.02	0.2	1.1
E_{MCU} (µWh)	2.642	24.9	117.3

4.1.2 Sensors

CM can be done using different sensor types [87], including vibration sensors. Vibration analysis is usually conducted using accelerometers and is the most common method for mechanical fault detection currently used in industry [25]. A vibration sensor produces an analog output proportional to the intensity of mechanical vibration or acceleration.

The two primary types of accelerometers used are based on piezoelectric and MEMS technology. Piezoelectric accelerometers are the most common sensors because of their high linearity, signal-to-noise ratio, and wide bandwidth [88]. However, MEMS accelerometers outperform piezoelectric accelerometers in terms of their size, power consumption, and integration of intelligent features. The specifications for three commonly available MEMS accelerometers suitable for this study are listed in Table 4-4.

Table 4-4. Condition monitoring MEMS accelerometers [89], [90], [91]

Parameters	ADXL1002	MODEL 8102	LIS25BA
Measurement range (g)	±50	±50	±4.16
Sensitivity (mV/g)	40	25	40
Resonant Frequency (Hz)	21000	6000	5150
Operating Voltage Range (V)	3.3 – 5.25	3 – 5.5	1.71 – 1.99
Quiescent Supply Current (mA)	1 – 1.15	0.022	3.3
Standby Current (μA)	225 – 285	0.022	0.010

The ADXL1002 is chosen for this study because it has low power consumption, while having a high resonant frequency. The sensor specifications are shown in Table 4-4. As with the MCU, these devices have a standby mode, where the device is turned off and all measurements are suspended, thereby consuming no energy. Equation (4-11) is used to determine the energy consumed by the sensor over one hour of operation.

$$E_{sa} = I_{sa} t_a V_s \tag{4-9}$$

$$E_{ss} = I_{ss} t_s V_s \quad (4\text{-}10)$$

$$E_{st} = E_{sa} + E_{ss} \quad (4\text{-}11)$$

where E_{sa} is the energy consumed by the sensor in active mode, I_{sa} is the current drawn by the sensor in active mode, E_{ss} is the energy consumed by the sensor in standby mode, I_{ss} is the current drawn by the sensor in standby mode, t_s is the amount of time spent by the sensor in standby mode and V_s is the voltage of the sensor. Table 4-5 summarizes the sensor's current consumption and the amount of data collected when the sensor is active for 1, 10 and 60 seconds of every hour at a sampling rate of 76,800 samples/second (lowest value for MCU to save energy).

Table 4-5. Simulated values for the ADXL1002 accelerometer

Specifications	Value		
s_r (sps)	76,800		
t_a (s/h)	1	10	60
E_{sa} (µWh)	1.4	14	83
E_{ss} (µWh)	0	0	0
E_{st} (µWh)	1.4	14	83

4.1.3 Power Management Unit (PMU)

The PMU regulates and distributes the power from the power receiver and storage unit to the remaining components. PMUs are widely available, but the ability to select their output voltage levels is useful depending on the voltage requirements of downstream devices. Different PMUs are compared in Table 4-6.

Table 4-6. PMU specifications [92],[93], [94]

Specifications	ADP5300	LTC3107	DA9061-A
Input Voltage Range (V)	2.15 – 6.5	2 – 4	2.8 – 5.5
Standby Current (nA)	18 – 130	6 – 7.5	1.5
Active Mode Current (µA)	2.3 – 3.2	6 – 7.5	9
Output Voltage Options (V)	0.8 – 5		0.9 – 3.6

None of these devices are 100% efficient and dissipate energy themselves, whether in their active or standby modes. The ADP5300 is chosen as the PMU in this application because it is the most efficient, consuming less current than the alternative devices, when in active mode. The PMU must be in active mode when other components of the device are in active mode, especially the MCU (t_{MCU_a}). Knowing this time and the required current supply of the PMU in active mode (I_{PMU_a}), as well as the PMU voltage (V_{PMU}), the total energy consumption in active mode (E_{PMU_a}) can be determined:

$$E_{PMU\_a} = I_{PMU\_a} * t_{MCU\_a} * V_{PMU} \tag{4-12}$$

$$E_{PMU\_s} = I_{PMU\_s} * t_{MCU\_s} * V_{PMU} \tag{4-13}$$

where I_{PMU_s} is the required current for the PMU in standby mode. Table 4-7 illustrates the ADP5300's current consumption for one hour of work.

Table 4-7. Simulated values of the ADP5300's energy consumption

Specifications	Value		
Voltage (V)	3		
t_a (s/h)	1	10	60
t_{MCU_a} (s)	1.02	10.11	60.57
E_{PMU_a} (μWh)	0.002	0.019	0.12
E_{PMU_s} (μWh)	0.054	0.054	0.053
E_{PMU} (μWh)	0.056	0.073	0.173

4.1.4 Device Energy Consumption Analysis

With the energy consumption of each device component known, the total energy consumption of the device in active or standby mode can be calculated (Figure 4-3). From this figure, it is clear that the higher the duty cycle, the more energy is consumed in active mode while the energy consumed in standby mode is similar. The device's total energy consumption (E) is the sum of the energy consumption of all individual components.

$$E_a = E_{MCU\_a} + E_{wr} + E_{PMU\_a} + E_{sa} \quad (4\text{-}14)$$

$$E_s = E_{MCU\_s} + E_{PMU\_s} + E_{ss} \quad (4\text{-}15)$$

$$E = E_a + E_s \quad (4\text{-}16)$$

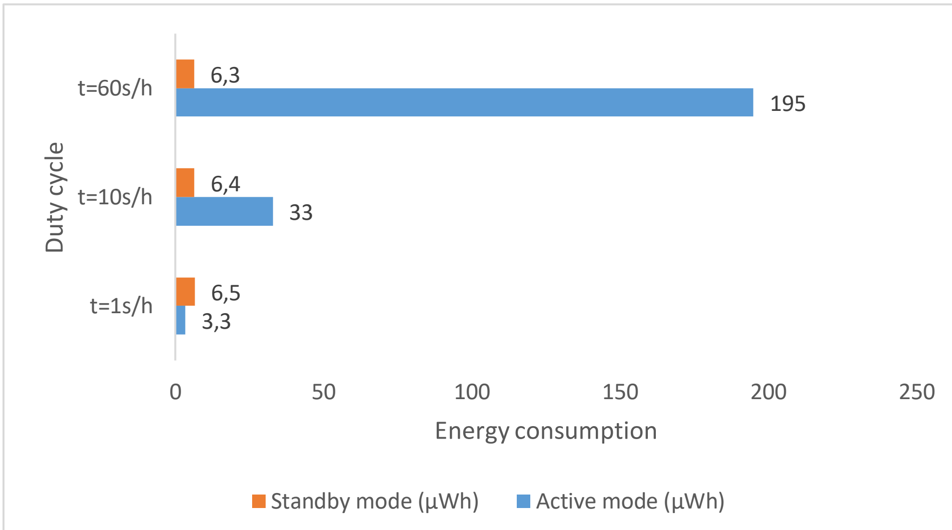
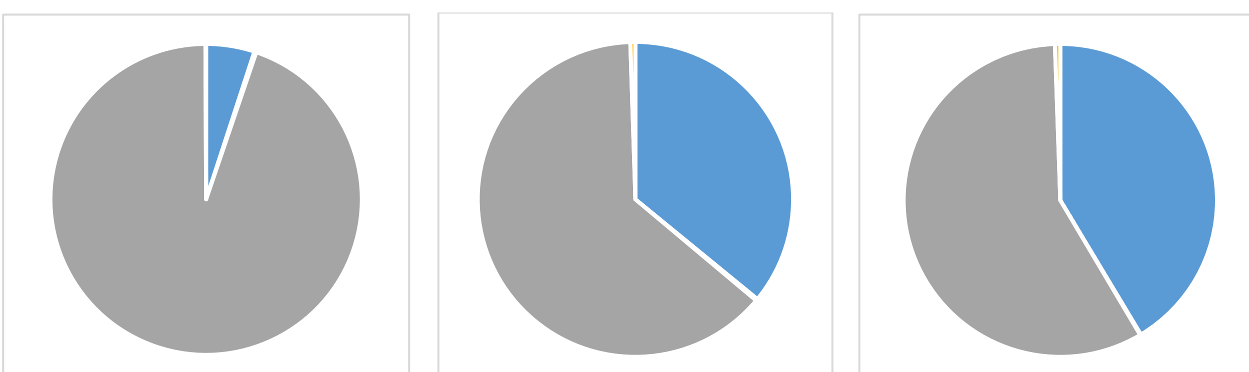


Figure 4-3. Energy consumption of all device components combined in different modes

The MCU dominates the power consumption of a wireless sensor device, as shown in Figure 4-4, but the difference when compared to the sensor energy consumption becomes less significant as more data is gathered (higher duty cycle). This is because the energy consumption of the wireless unit is not linear like the one for the sensor.



(a) t_a = 1s/h (b) t_a = 10s/h (c) t_a = 60s/h

Figure 4-4. Component energy consumption for different duty cycles. Grey: MCU, blue: sensor, red: PMU (too low to be visible), wireless: yellow

Table 4-8 provides a comparison of the energy consumption of other published sensors.

Table 4-8. Active mode energy consumption of other published sensors

Reference	Sensor	Energy Consumed
[95]	Combined soil moisture sensor, temperature sensor, and rain sensor,	0.6J
[96]	Temperature sensor	3.44J
[97]	Ambient light sensor	0.00050mJ
[98]	Position tracking sensor	56mJ
This work	Vibration sensor	34.92mJ

4.2 Energy Supply Technology

Devices can be powered from different sources: non-rechargeable energy storage units, rechargeable energy storage units, or directly scavenged from an energy source (e.g. RF wireless energy in our work) in real time. When considering energy harvesting strategies, combinations of these different methods are also possible and might even be required, depending on the available amounts of scavenge-able energy and the required active/standby duty cycles for specific applications.

4.2.1 RFEH

When choosing a dedicated power transmitter, equation (2-3) needs to be considered to control the output power that is generated. The output power needs to be large enough to be harvested by the device, but still compliant with wireless energy safety regulations, usually expressed as maximum transmitter power for data communications applications or as maximum launched power for power charging applications [61], [99]. As an example, the PowerCast TX91501B is a RF power transmitter that supplies power to end devices at a maximum effective transmit power of 3 watts and a frequency of 915 MHz [100]. The power receiver is a rectifying antenna (rectenna) that converts RF power into DC power. A rectenna is typically composed of an antenna, an impedance matching circuit, and a rectifier. In the absence of intervening obstacles, the received power is linked to the transmitted power by equation (2-3). The carrier frequency of both components needs to be equal, and their antennas need to be chosen carefully to improve the efficiency of power transfer. The PowerCast P2110B power harvester is an example of a RFEH module that converts RF energy from the PTX into DC energy [100]. PowerCast provides a simulator tool for determining the power received (P_r) under various conditions [100]. For this study, the simulation consists of the PowerCast TX91501B transmitting 3 W of power to the PB2110 receiver using a 6.1 dBi patch antenna. The amount of power that can be harvested by the receiver for different transmission distances between the transmitter and receiver is given in Figure 4-5. This figure shows that the practical energy harvesting distance is less than 5 m and that the power exponentially increases when the device gets closer to the PTX.

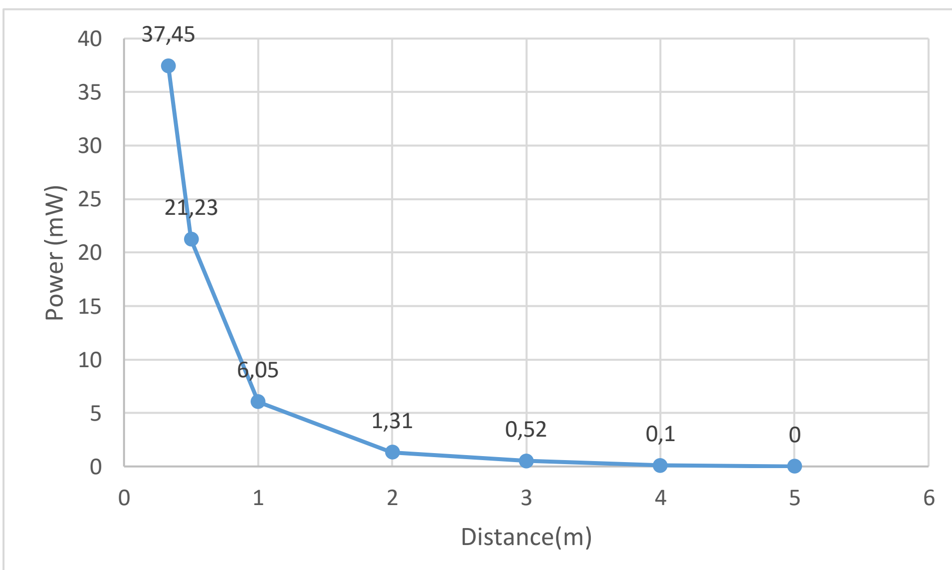


Figure 4-5. Power harvested by the P2110B receiver [100]

Because the simulation is in a far field WPT condition, the transmission distance needs to be greater than the wavelength, as per Friis' equation conditions [49]. By using equation (2-2), with a frequency of 915 MHz, the PTX-node distance (*d*) needs to be greater than 0.328m, but cannot be more than 4m (Figure 4-5) for charging purposes. It is assumed that RFEH is continuously available. Figure 4-6 compares the power harvested at different distances to the power required by the device at different values of duty cycle. Based on these figures, the transmitter can be placed at:

- 4m or less to power the device specified in this work with a duty cycle of less than 1s/h,
- 3m or less to power the device specified in this work with a duty cycle of less than 120s/h,
- 2m or less to power the device specified in this work with a duty cycle of less than 300s/h,
- 1m or less to power the device specified in this work with a duty cycle of less than 1800s/h.

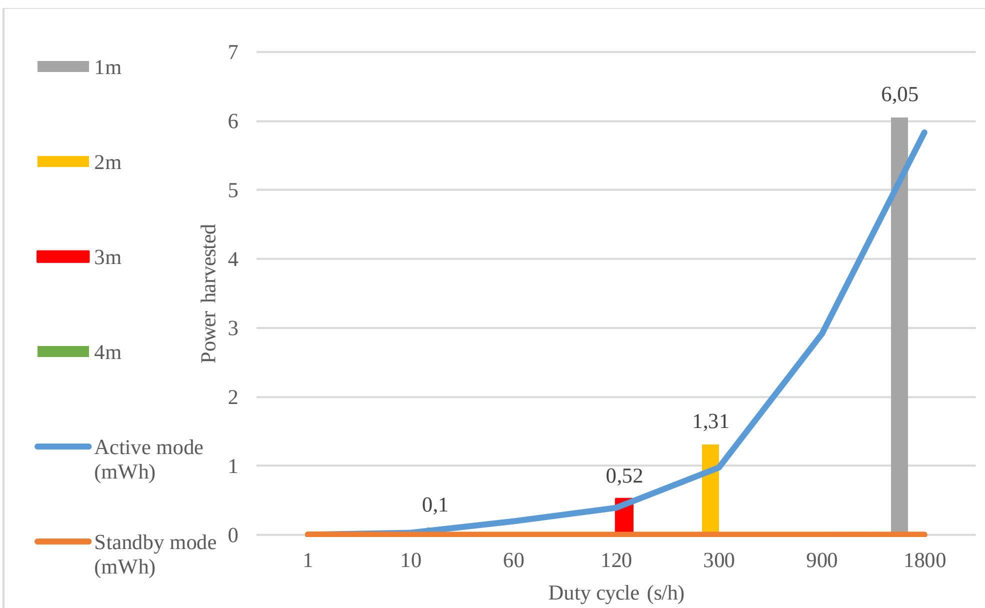


Figure 4-6. Comparison of power harvested at different PTX-node distances with device energy consumption under different duty cycles

To power a device only with RFEH, the energy source must generate more current than all device components required in both standby and active modes. Given the dramatic difference in active and standby current requirement for these components, this depends critically on the active/standby duty cycle, which is dictated by the data collection requirements of the specific CM application. Standalone RFEH (without energy storage) is limited by the MCU's Bluetooth module's 8.5 mA current requirement, as shown in Table 4-2. To generate at least 8.5 mA, d must be a maximum of 0.45m (30.6mW at 3.6V).

4.2.2 Energy Storage

For energy storage, the most common options are batteries and capacitors. Batteries are classified into non-rechargeable and rechargeable categories. Non-rechargeable batteries (NRB) are a significant issue for the environment and the elimination of batteries from landfill sites has been prioritized by several government organizations [101]. By comparing the device simulated in section 4.1 to existing products in section 2.6, it can be seen that, a NRB life is significantly longer than the battery life of a rechargeable battery (RB). While in active mode for a duty cycle of 60s/h and using the same NRB as used in [10] (3.6V, 2400 mAh), the proposed device will remain powered for 4.9 years (equation (4-17), compared to 3 years in [10] and 2 years in [11], based on the specified energy consumption.

$$d_L = \frac{V_{batt} I_{batt}}{E} \tag{4-17}$$

If the battery used in [102] (shelf life of 10 years) is a RB instead, and if it is being continuously recharged by the RFEH source over the years, the device would last 60.5, 13.1, 5.2, and 1 year longer if d is equal to 1m, 2m, 3m, and 4m, respectively. However, all batteries have a shelf life, which limits their overall lifetime. Equation (4-18) is used to find the device lifetime values. With the energy consumption of the device and power transmitted known, it can be seen that the time to fully charge a RB will be expressed in terms of hours, day or weeks depending on the specification of the RB.

$$d_L = \begin{cases} S_l, & if \ \frac{V_{batt} I_{batt}}{E} + P_r S_l \geq S_l \\ \frac{V_{batt} I_{batt}}{E} + P_r S_l, & otherwise \end{cases} \tag{4-18}$$

Combining a NRB and a RFEH source enables the device to be powered by the NRB during active mode and by the RFEH source during standby mode. Using the same example as before and equation (4-18), a 3.6V, 2400 mAh battery will last 5.05 years with a duty cycle of 60s/h. For this case, d could be greater than 0.45m because the standby energy requirement is 6.3 µWh, which is achievable over PTX-node distances of up to 4 m (section **Erreur ! Source du renvoi introuvable.**). Similar to a NRB, the lower the active/sleep duty cycle, the longer the device lasts using this method.

$$d_L = \frac{V_{batt} I_{batt}}{E_a} \tag{4-19}$$

Table 4-9 summarizes all energy types used when powering the simulated devices for 60 seconds every hour. The device is powered by a 3.6V, 2400 mAh battery in all cases. For charging distances of 3 m or less, battery shelf life dominates device lifetime.

Table 4-9. Comparison of different methods for powering wireless CM devices

Type	d_L	d	t_a
ABB [11]	2 years	N/A	Measurements taken once per hour and data collected once per day by a mobile device
Fluke [102]	3 years	N/A	Unknown
NRB	4.9 years	N/A	60s every hour
NRB/RFEH	5.9 years	> 0.45m < 4m	60s every hour
RB/RFEH	5.9 years	4m	60s every hour
RB/RFEH	10 years	3m	60s every hour
RB/RFEH	10 years	2m	60s every hour
RB/RFEH	10 years	1m	60s every hour
RFEH	Unlimited (no battery is required)	< 0.45m	All active mode values

4.2.3 Methodology

To select an adequate energy supply when integrating RFEH, the vibration data logging requirements must be defined. This determines t_a. The MCU and wireless power overhead need to be minimized, without sacrificing application performance requirements. With t_a and the

MCU power consumption known, the energy consumption of the device (*E*) can be calculated. The optimal energy storage can be determined based on the maximum allowable PTX-node distance in the environment (max_d). Based on this distance, the device powering method can then be chosen. If there are no physical objects that restrict the placement of the PTXs and there are no restrictions on the number of PTX units required in multi-node networks, using RFEH alone is the best way to power the device because it will last longer in theory. This means placing the PTX within the maximum allowable range of the device to permit the use of RFEH as the only source of power (i.e. less than 0.45m, as mentioned in the example provided). Practically, it will often be impossible to place the PTX this close to the device, requiring a similar analysis to the one given in Figure 4-6. Assuming a combination of RB/RFEH power, the maximum possible distance to the device from the PTX must be determined. For combined RB/RFEH power to be worthwhile, the lifetime improvement needs to satisfy some sort of user efficiency factor (E_{factor}), which can be an accepted value of lifetime improvement. The efficiency factor can be the amount of time the user would like the device to work before needing to replace it. In this case RB/RFEH will need to allow the device to work for longer than the user efficiency factor. If neither of the two previous methods are feasible, a NRB can be used, at the cost of a reduction in environmental sustainability. As an interesting way of extending the lifetime of the NRB resource, assuming the availability of "enough" (in both power level and timing) scavenge-able power at a node, a combination of NRB and RFEH can be used, since standby mode energy consumption can be large after a long period of time. In the example provided herein, using such a hybrid mode only gave an increase of 1.9 years in lifetime. There is a trade-off between the energy consumption of a device in standby mode and the awake/standby duty cycle. This is summarized using the following equations:

$$\begin{cases} RFEH, & if\ at\ max_d, P_r > E\ \&\ \dfrac{P_r}{V_{MCU}} > I_{wr} \quad (a) \\ \dfrac{RB}{RFEH}, & if\ at\ max_d, P_r > E_s\ \&\ \dfrac{V_{batt}I_{batt}}{E} + P_r S_l > E_{factor} \quad (b) \\ \dfrac{NRB}{RFEH}, & if\ \neg(a/b)\ \&\ P_r > E_s\ \&\ \dfrac{V_{batt}I_{batt}}{E_a} > E_{factor} \quad (c) \\ NRB, & otherwise \quad (d) \end{cases} \tag{4-20}$$

In general, the following steps can be followed to design a CM device powered by RFEH:

Step 1	Determine the amount of data needed to detect failure. This will help determine the duty cycle (t_a)
Step 2	Using equations (4-1) to (4-8), find the total energy consumed, the energy consumed in active mode and standby mode, by the MCU.
Step 3	Using equations (4-9) to (4-11), find the total energy consumed, the energy consumed in active mode and standby mode, by the sensor.
Step 4	Using equations (4-12) and (4-13) , find the energy consumed in active mode and standby mode by the PMU.
Step 5	Using equations (4-14) to (4-16) , find the total energy consumed, the energy consumed in active mode and standby mode, by the device.
Step 6	Using equation (2-3), calculate the maximum allowable distance between the PTX and any given node.
Step 7	Using equation (4-20), select the possible energy storage strategy.

4.3 Summary

In this chapter, the components required to power a low-power CM device have been analyzed. The defined options are: RFEH alone, energy storage units or a combination of the two. Each component affects the total energy consumed by a node and the device's energy consumption is greatly affected by the standby/active mode duty cycle, which determines the feasibility of using RFEH, either alone or combined with other energy supply types. Power transmission is analyzed for directional antennas to determine the maximum allowable distance between a PTX and any given node. Knowing this distance, the type of energy storage between NRB, NRB/RFEH, RB/RFEH, or RFEH can then be determined. This distance can then be used as an input for the energy transfer strategy proposed in Chapter 3.

With ongoing research in RFEH and device components, this work can serve as a framework for determining the optimal energy source type. As energy consumption of individual components is improved, and the specific parameters of different applications are considered, this optimality will change. However, the analysis progression, starting with NRB when no wireless power is available for recharging, progressing through combinations of NRB, RB and RFEH to pure RFEH, when lots of continuous wireless power is available, is expected to remain valid. With all of the application inputs assumed (i.e. populated with “reasonable” values), Chapter 5 will go through the algorithm used to find possible PTX location(s) in a WSN.

Chapter 5

Simulations

This chapter will simulate values for the inputs and parameters that were determined to affect the implementation of RFEH in a WSN. The simulations will determine the minimum number and optimal placement strategy/configuration of PTXs so that all WSN nodes in a network remain powered during machine CM. All simulation assumptions will be defined and the simulation algorithm described before case examples are discussed. The impact of node distribution, node energy consumption, and PTSA will also be discussed in this chapter. The MATLAB code developed for this simulation is provided in Appendix B. A flow chart describing the process, including every element used in the simulation, is shown in Figure 5-1.

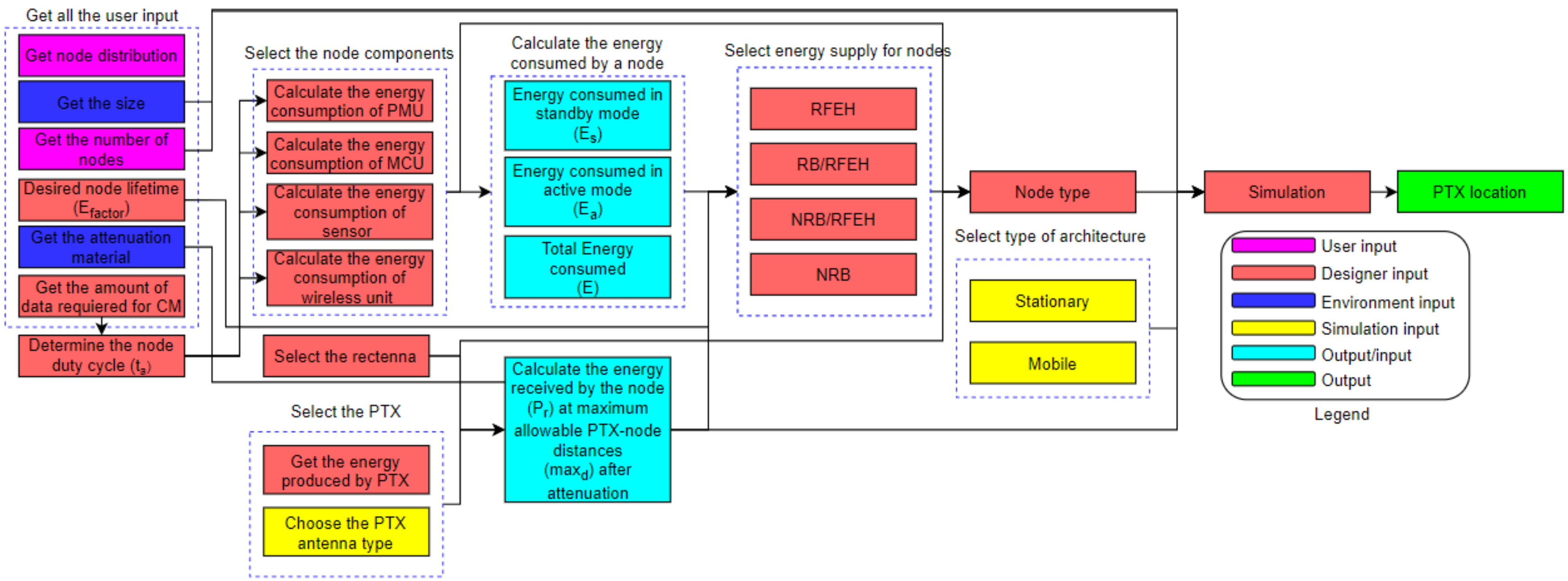


Figure 5-1. Flow chart of simulation elements

5.1 Assumptions and Simulation Objectives

In these simulations, inputs and parameters presented in Chapter 3 and Chapter 4 will be assumed to be as close to real-world values as possible. The inputs, parameters, and expected outputs for the simulations are tabulated in Table 5-1.

Table 5-1. List of inputs, parameters, and outputs

Parameters	**Description**
A	Antenna type
PA	Power architecture
p	Energy received by a node from a PTX
Inputs	**Description**
a	Power transmission signal attenuation
E	Energy required by a node
N	Node density
ND	Node distribution
P	Plant size
r	Coordinates of a node
s	Storage capacity
t_a	Duty cycle
Outputs	**Description**
k	Number of nodes powered by a given transmitter
l	Coordinates of PTX

5.1.1 Inputs

PTSA

PTSA will be calculated in accordance with the material types listed in Table 3-1. The material attenuation values will be determined randomly within the following ranges:

- LOS: 0dB
- Brick: 3.5 – 7 dB
- Concrete: 12 – 35 dB

All other material types listed in Table 3-1 are included in the selected range of values. Therefore, they are not explicitly included as simulation inputs.

Node distribution

Nodes are distributed according to three different distributions: clumped, randomly distributed, and uniformly spaced, as described in section 3.2. Node locations can also be entered manually into the simulator.

Node energy consumption

The energy consumption of each node is based on the node design, as defined by the design framework in Chapter 4.

Plant size

For this simulation, the plant size can be manually entered by the user, but is assumed to be a three-dimensional rectangular prism of arbitrary size having length, width, and height values of (l, w, h).

Node density

The number of nodes in the plant is calculated as follows:

- For a high node density: 0.2 nodes/m^2 [103]
- For a low nodes density: 0.044 nodes/m^2 [103]

Power produced by a PTX

The power produced by a PTX can be manually entered by the user. The maximum power that can be transmitted with respect to all safety and regulatory requirements described in [99] cannot exceed 4W.

Energy storage unit

The simulation only simulates rechargeable storage. The voltage value of the storage is fixed at 3.6V for all cases (standard for CM devices) and the capacity can be decided by the user.

Duty Cycle

The simulation can accept any value of duty cycle ranging from 1s/h – 3600s/h.

Implementation Cost

The following costs have been considered in this simulator:

- Cost of individual transmitters
- Cost of individual nodes
- Cost of installing each PTX station
- Cost of human labour (hourly wage × labour time)
- Cost of energy to power each transmitter: $0.179 × energy consumption [104]

5.1.2 Parameters

Antenna type

Two different types of PTX antennas are simulated: omni-directional and directional antennas.

Powering architecture types

There are two types of power architectures used in the simulator: stationary and mobile. The stationary architecture is composed of fixed PTX devices that continuously supply energy to nodes. The mobile architecture is comprised of PTX devices, whose location can vary, but which remain in fixed locations while charging a node. Mobile PTX devices can provide energy either continuously or intermittently. STSN and STMN strategies are used to simulate the stationary architecture. For this work, only STSN is simulated when using a mobile architecture.

5.2 Simulation Algorithm

The following algorithm is used to obtain simulation results:

Step 1:	Simulation parameter setup
Step 2:	Generate initial PTX locations
Step 3:	Determine feasible PTX locations
Step 4a:	Determine final PTX locations (stationary architecture)
Output 1	Number of PTXs required by the system for stationary architecture Location and orientation of each PTX Cost of implementation (C)
Step 4b:	Determine PTX trajectories (mobile architecture)
Output 2	Number of PTXs required by the system for mobile architecture Location and orientation of each PTX Cost of implementation (C)

5.2.1 Step 1: Simulation Parameter Setup

This step consists of setting the following parameters to the desired values. All combinations of these values have been simulated:

- Plant size (w, l, h)
- Node density (N) (e.g. high or low)
- Node distribution (ND) (e.g. clumped, random, uniform spacing)
- Attenuation for each node (a) (e.g. random assignment of intervening material type of LOS, brick, or concrete)
- Energy consumed by each device (E)
- Energy storage capacity (s)
- PTX energy input (P_t)
- Greatest allowable PTX-node distance, based on the energy consumed calculated using equation (**2-3**), and applying random PTSA to each node

5.2.2 Step 2: Generate Initial PTX Locations

The environment defines initial PTX locations. Essentially, the simulator defines locations where it is physically possible to place PTXs.

Antenna radiation coverage generation

The MATLAB antenna toolbox extension is used to generate PTX coverage regions. The required types of antenna radiation can be generated, automatically, by selecting the antenna type in the toolbox.

To generate antenna radiation manually:

- A sphere of radius r is used to represent omni-directional radiation. The radius r is equal to the greatest allowable PTX-node distance, as determined in step 1.
- An ellipsoid with radiuses (r_x, r_y, r_z) is used to represent directional radiation. r_x represents the radius in the x direction, r_y represents the radius in the y direction and r_z represents the radius in the z direction. r_x and r_y are calculated using the HPBW of an antenna. r_z is the maximum PTX-node distance, as determined in step 1.

Cover entire required space with PTX

The next step of the simulation is to know which node is covered when a PTX is placed at a certain location.

1. The PTX is placed in several locations before recording the nodes that it covers. The principle is to first place the PTX at the origin (position 1), record the nodes that are covered by that PTX and move the PTX by a few cm in the x direction and record the nodes that are covered.
2. The previous step is repeated until the end of the plant is reached in the x direction.
3. Then from position 1, move the PTX by a few cm in the y direction and repeat all the previous steps.
4. The same is done in the z direction.
5. For simplicity, that movement between positions is done uniformly (i.e. at a distance of d_{ant}).

6. Finally, a list of all initial PTX positions, along with the nodes covered in each position, are made available.
7. A node is considered to be within a PTX coverage region if the distance between the PTX and the node is less than the greatest calculated distance between the PTX and node in step 1. If a PTX does not have any node in its coverage region, it is not considered.

In summary, for a plant of coordinates ($l \times w \times h$) and a PTX with an omni-directional antenna of spherical radius r, equation (5-1) represents the list of all PTX position considered.

$$PTX_{omni} = (n \times d_{ant}, m \times d_{ant}, \mathrm{p} \times d_{ant})$$
$$where \left\{n = \left(0,1,\ldots,\left(\frac{l}{d_{ant}}\right)+1\right), m = \left(0,1,\ldots,\left(\frac{w}{d_{ant}}\right)+1\right), p = \left(0,1,\ldots,\left(\frac{h}{d_{ant}}\right)+1\right)\right\} \quad (5\text{-}1)$$

For a directional antenna, each PTX with radiuses (r_x, r_y, r_z) will undergo a 360-degree rotation with steps of $\mathrm{r_{ant}}$ degrees. The equation for a directional antenna is:

$$PTX_{direct} = PTX_{omni} \times \begin{bmatrix} \cos\theta_x\cos\theta_y & \cos\theta_x\sin\theta_y\sin\theta_z - \sin\theta_x\cos\theta_z & \cos\theta_x\sin\theta_y\cos\theta_z + \sin\theta_x\sin\theta_z \\ \sin\theta_x\cos\theta_y & \sin\theta_x\sin\theta_y\sin\theta_z + \cos\theta_x\cos\theta_z & \sin\theta_x\sin\theta_y\cos\theta_z - \cos\theta_x\cos\theta_z \\ -\sin\theta_y & \cos\theta_y\sin\theta_z & \cos\theta_y\cos\theta_z \end{bmatrix} \quad (5\text{-}2)$$
$$\text{Where}\left\{\theta_x, \theta_y, \theta_z = \mathrm{r_{ant}} \times q \text{ and } q = \left(0,1,\ldots,\frac{360}{\mathrm{r_{ant}}}\right)\right\}$$

Suppose there is a 2 × 2 m^2 rectangular plant, with 10 randomly placed nodes. PTXs used have an omni-directional antenna of radius 0.5m and a d_{ant} of 0.25m. The centre of the first PTX will be placed at (0,0) as illustrated in Figure 5-2a. The next n PTXs will be placed at ($n \times 0.25$,0) where $n \times 0.25 < 2$, as illustrated in Figure 5-2b. The next $n \times m$ PTX will be placed at ($n \times 0.25$, $m \times 0.25$) where $m \times 0.25 < 2$, as illustrated in Figure 5-2c. In this case, it can be seen that there are several PTXs that do not have any nodes in their coverage region. These PTX locations will not be considered by the system, as represented in Figure 5-3.

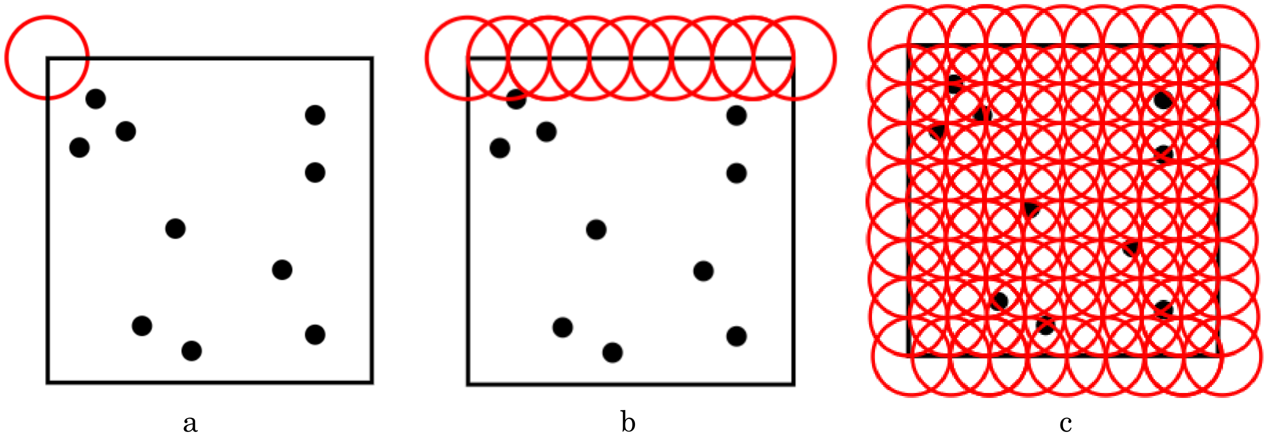


Figure 5-2. Steps to generate PTXs over the entire space supported by a WSN

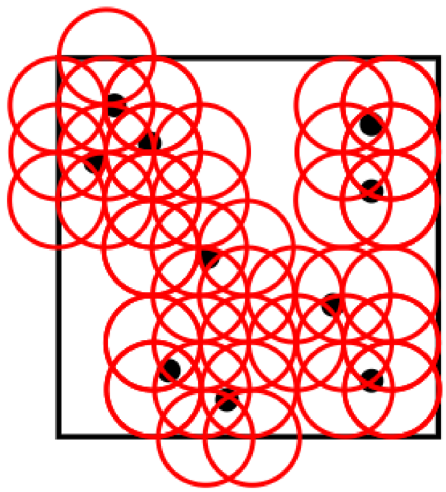


Figure 5-3. Initial PTX locations that have nodes in their coverage region

5.2.3 Step 3: Determine Feasible PTX Locations

A feasible PTX location is a location where a PTX contains nodes in its coverage region after the effect of attenuation is considered. After the initial PTX locations have been determined, feasible locations can be found by following these steps:

1. Using equation (2-3), determine how much power each node receives from each initial PTX location.
2. Using equation (**3-1**), determine the smallest amount of power required to power each node, following attenuation.

3. If the power received from a PTX is less than the power required by a node after attenuation, the node should be removed from the PTX's coverage region.
4. A list of all nodes that are covered by each PTX is created.

Suppose that 2 nodes, N1 and N2, each require 1 mW of received power to operate or be charged. Both nodes are in the coverage region of a PTX that provides them with 1.3 mW when in LOS. Suppose now that after attenuation, the power requirements for N1 and N2 increase to 1.5 mW and 1.2 mW respectively. The PTX will no longer be able to power N1, which means that N1 will be removed from the PTX's coverage region.

At the end of step 3, only feasible PTX locations remain.

5.2.4 Step 4a: Determine Final PTX Locations (Stationary Architecture)

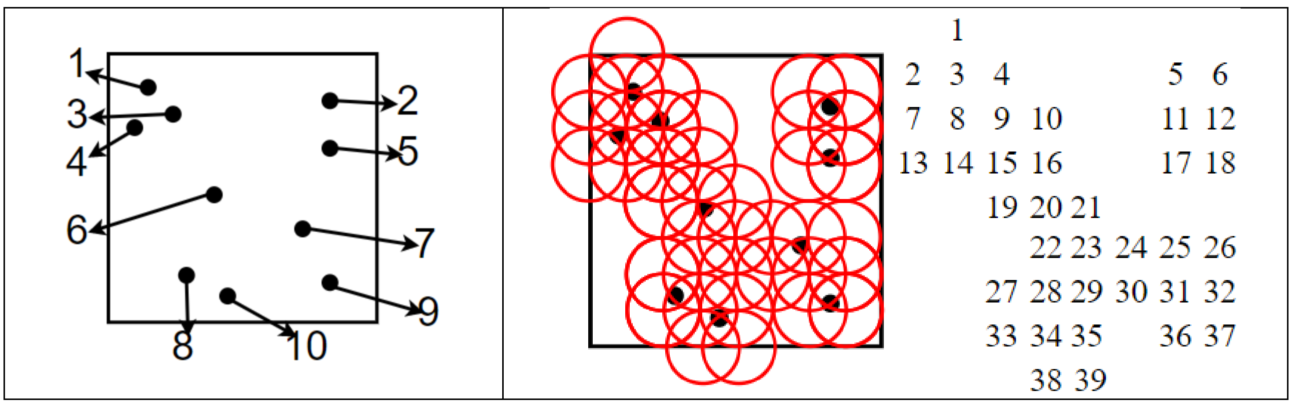
Next, we prioritize the list of PTX devices. For this, a node coverage count can be used. A node coverage count (NCC) refers to the number of nodes that a PTX covers. For instance, if a PTX is capable of simultaneously powering 9 nodes, it has a NCC of 9.

1. Consider the list of feasible PTXs created at the end of step 3. If a node is identified as being associated with a PTX with a NCC of n, remove all PTX that cover that same node if the PTX in question has a lower NCC.
2. Each node will now only be found in PTX device lists that have the same or greater NCC.
    - If multiple PTXs of the same NCC cover the exact same nodes, the simulator will choose the first one in the list. This way of choosing will not have any impact on the system's efficiency for the inputs and parameters considered.
    - If multiple PTXs of the same NCC cover one or more common nodes, the simulator will select the combination of PTXs that result in the highest cost efficiency. Suppose that PTX-1 covers N1, N2, N3 and N4, and PTX-2 covers N4, N5, N6 and N7. Those

two PTXs both cover N4 and have the same NCC. In this case the system will consider 2 choices:

- Locations of PTXs having lower NCC that cover only N1, N2 and N3, while PTX-2 covers N4, N5, N6 and N7.
- Locations of PTXs having lower NCC that cover only N5, N6 and N7, while PTX-1 covers N1, N2, N3 and N4.

Continuing the example from step 3, let the nodes and PTXs be numbered as shown in Figure 5-4. Based on this, the NCC of PTXs are listed in Table 5-2.



a b

Figure 5-4. Numbering of a) nodes and b) PTXs

Table 5-2. List of PTXs, their NCC and the nodes that they cover

PTX #	NCC	Node #	PTX #	NCC	Node #	PTX #	NCC	Node #	PTX #	NCC	Node #
1	1	1	11	2	2,5	21	1	6	31	2	7,9
2	1	1	12	2	2,5	22	1	6	32	1	9
3	3	1,3,4	13	1	4	23	1	6	33	1	8
4	2	1,3	14	1	4	24	1	7	34	2	8,10
5	1	2	15	1	3	25	1	7	35	1	10
6	1	2	16	1	6	26	1	7	36	1	9
7	1	4	17	1	5	27	1	8	37	1	9
8	3	1,3,4	18	1	5	28	1	8	38	1	10
9	3	1,3,4	19	1	6	29	1	10	39	1	10
10	1	3	20	1	6	30	1	7			

- By looking at Table 5-2, the highest NCC is for PTX-3, PTX-8 and PTX-9.
- By applying the second step of step 4, the following PTXs will be removed from the list:

- PTX-1, PTX-2, PTX-4 because N1 is already in PTX-3;
- PTX-8 and PTX-9, which all have a higher NCC;
- PTX-15 because N3 is already in PTX-3, PTX-8 and PTX-9 which all have a higher NCC;
- PTX-7, PTX-13 and PTX-14, because N4 is already in PTX-3, PTX-8 and PTX-9, which all have a higher NCC.

- Now, N1, N3 and N4 are only found in PTX devices with a NCC of 3. In those 3 locations, the PTX devices all have the same (and highest) NCC. All 3 locations are valid. Hence, the simulator will select the first PTX location on the list. The selection process won't affect the system cost efficiency because regardless of the choice, the number of PTXs will stay the same.
- By repeating the same process for the other NCCs, the final table can be made to look like Table 5-3.

Table 5-3. List of PTXs, their NCC and the nodes that they cover after applying step 4

PTX #	Priority level	Node #	PTX #	Priority level	Node #
3	3	1,3,4	20	1	6
8	3	1,3,4	21	1	6
9	3	1,3,4	22	1	6
11	2	2,5	23	1	6
12	2	2,5	31	2	7,9
16	1	6	34	2	8,10
19	1	6			

This can also be represented as in Figure 5-5. Note that if a coverage region touches a node, the node is considered inside that coverage region.

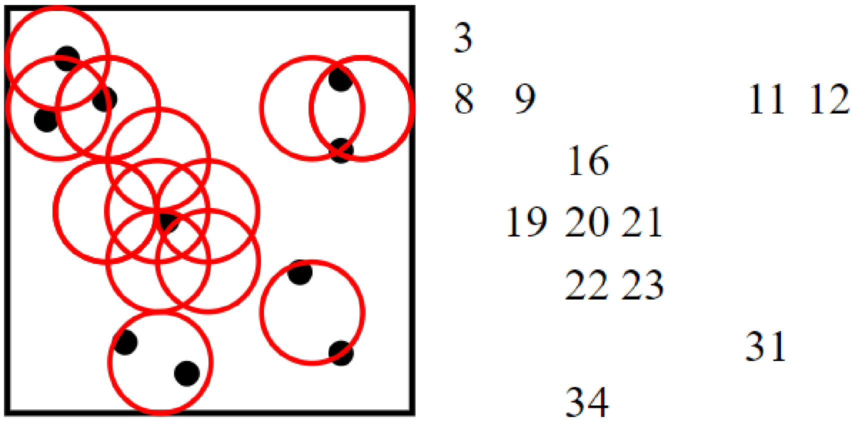


Figure 5-5. PTXs with the nodes covered after applying step 5

As mentioned above, in cases where multiple PTX devices cover the same nodes, the system will randomly select one of the PTX locations. All locations are valid and selecting one over the other does not affect the system cost efficiency. For this example, PTX-8, PTX-11 and PTX-20 will be kept. The final result is shown in Figure 5-6.

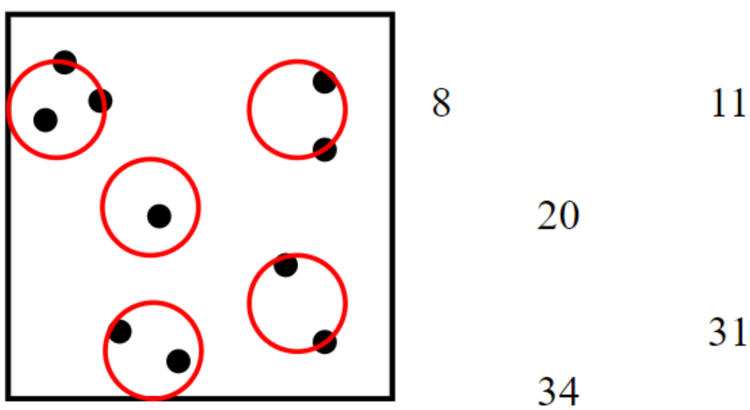


Figure 5-6. PTXs with the node they cover

5.2.5 Step 4b: Determine PTX Trajectories (Mobile Architecture)

As discussed in Chapter 2, mobile architectures for powering nodes with RFEH have already been developed. Some of these are based on the Hamiltonian cycle (approaches based on the shortest path between nodes), where the distance between power sources and the nodes that they are powering is being minimized, while others are based on predetermined schedules or on energy storage unit levels. In this simulation, a predetermined schedule for all PTXs will be used because, in this case, the system can predict when the node will require energy. The objective is to determine the path and time required for a single PTX to charge as many nodes as possible before returning to charge previous nodes before they are discharged.

Consider a system with five nodes that requires charging via the mobile architecture (Table 5-4). Each node has a charge of 100 percent in its initial state. Because of node homogeneity, nodes are assumed to each discharge at the same rate. In reality and in our simulation, nodes will each charge at a different rate, depending on intervening attenuation and distance from the PTX to the nodes in question, but it is assumed that discharging and charging rates are identical just to simplify the example.

With these simplifying assumptions, at the end of the second charging cycle, all nodes will be charged to 75%, and the PTX will begin charging N1. N1 is fully charged after the third charging cycle, and the PTX begins charging N2. N2 is fully charged after the fourth charging cycle, and the PTX begins charging N3. N3 is fully charged after the fifth charging cycle, and the PTX begins charging N4. However, at the same time N5 is also discharged. In this case, N5 will be excluded from the charging zone and will require charging via another PTX.

Table 5-4. Example of a mobile architecture charging cycle*

Cycle	N1	N2	N3	N4	N5
1	100%	100%	100%	100%	100%
2	75%	75%	75%	75%	75%
3	100%	50%	50%	50%	50%
4	75%	100%	25%	25%	25%
5	50%	75%	100%	1%	1%
6	25%	50%	75%	100%	
7	100%	25%	50%	75%	
8	75%	100%	25%	50%	
9	50%	75%	100%	25%	
10	25%	50%	75%	100%	

*Cells highlighted in yellow indicate that the node is charging, while cells highlighted in red indicate that the node is low on charge. Please note that the charge percentage applied to each node in the simulation will differ from those in this example and will depend on both distance between PTX devices and nodes as well as intervening sources of attenuation.

In this thesis, the positions of PTXs when charging a node are known. There are additional criteria for choosing the trajectory, as explained in section 2.7, such as the Hamiltonian cycle, or the nodes with the lowest charge time. The shortest path is not used because the distance between two nodes has little effect on the overall result in this case. This is due to the long charging time. Indeed, because the charging time for nodes is measured in days, a few minutes of travel time between nodes will have no discernible effect on the outcome.

a) STSN

1. Identify the location of each node that receives the greatest amount of power from a specific PTX.
2. Calculate the time required for each node to be fully charged.
3. Calculate the time required for each node to be fully discharged.
4. Determine the resting position and time required for a single PTX, in that resting position, to charge as many nodes as possible before returning to a different position to charge other nodes before they are discharged.
5. Determine the required number of PTX.

a) STMN

1. Establish charging zones. A charging zone is the area within which a single PTX can power groups of nodes sequentially, prior to the complete discharge of any node in that same group. This area is determined by the PTX's coverage region.
2. Generate the travelling trajectory and calculate the time required for a single PTX to travel to as many charging zones as possible in order to charge nodes before returning to previous charging zones before their nodes are discharged. The time required for a single PTX to power a zone is the same as the time required for the PTX to power the node with

the lowest charge rate plus the travel time to move from one charging zone to the other charging zone.

5.2.6 Expected Outcome

The simulator determines the total number of PTXs required by the system for stationary and mobile architectures, as well as the location and orientation of each PTX charging zone centre. The system efficiency for each combination of architecture and antenna type using equation (3-2) can then be calculated.

5.3 General Cases and Impact of Parameters

This section discusses several test cases used in the simulation to determine the effect of node distribution, duty cycle, and PTSA. The following fixed values will be used:

- Plant size: 15 × 15 × 10 m
- Node density: High (46 nodes for described plant size)
- Node energy consumption: 35 mJ (Chapter 4)

5.3.1 First Step: Simulation Parameter Setup

The general description of each example case is as follows:

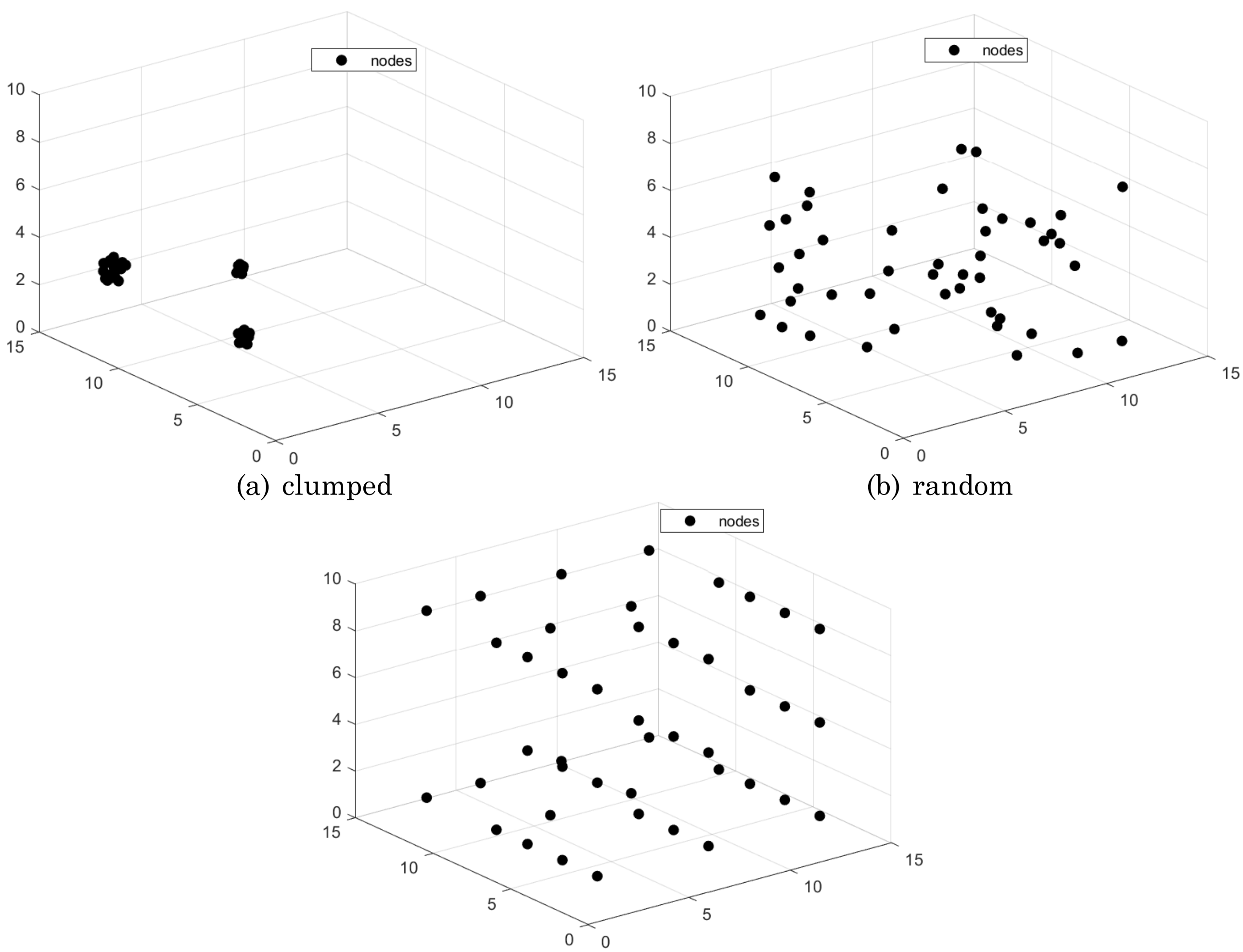
Case *i*: A *rectangular* manufacturing plant measuring *15 × 15 × 10 meters*, with *46* CM nodes placed *[*using *node distribution]* and that are affected by attenuation caused by *[attenuation]*. The nodes are homogeneous, and each node consumes *9.7 µWh (35 mJ)*, for a duty cycle of *1s/h*, an energy storage capacity of *3.6V* x *10mAh = 36 WAh (129 600 J)*, and an antenna receiver having a *6.1 dBi gain*. Each PTX has a power output of *3W* at a frequency of *915 MHz*. The cost of implementation is the following: cost of individual transmitters *$300* [105], cost of individual receiver *$300* [106], [107], [108], [109], [110], cost of installing each PTX station *$200*, cost of human labour *$25/h* [111]. For all nodes, the lifetime is expected to be *10 years*.

Table 5-5 contains the mix of power architectures, antenna types, and node types used for the simulation.

Table 5-5. Architecture and antenna used for the simulation

Id	Architecture	Antenna	Node type
SO	Stationary	Omni-directional	Storage/zero storage
SD	Stationary	Directional	Storage/zero storage
MO	Mobile	Omni-Directional	Storage
MD	Mobile	Directional	Storage

Figure 5-7 illustrates the node distribution in each case.



(a) clumped

(b) random

(c) uniformly spaced (3D view makes it hard to clearly see all nodes uniformly spaced, but they are)

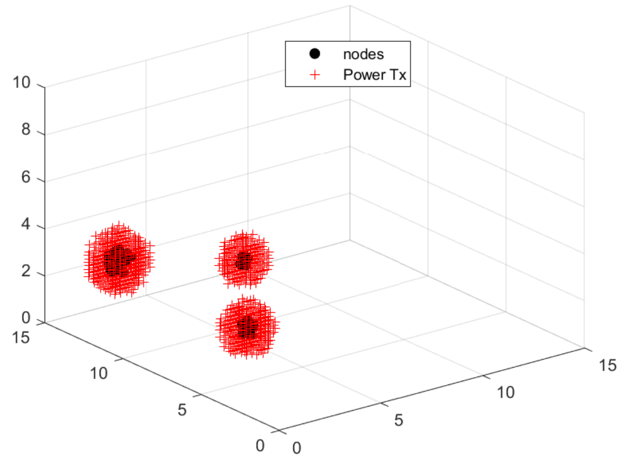
Figure 5-7. 3D visualization of node locations for different distribution cases

By using equation (**2-3**), the maximum PTX-node distance is calculated. For this case, the values are calculated to be 0.92m for an omni-directional antenna and 2.3m for a directional antenna.

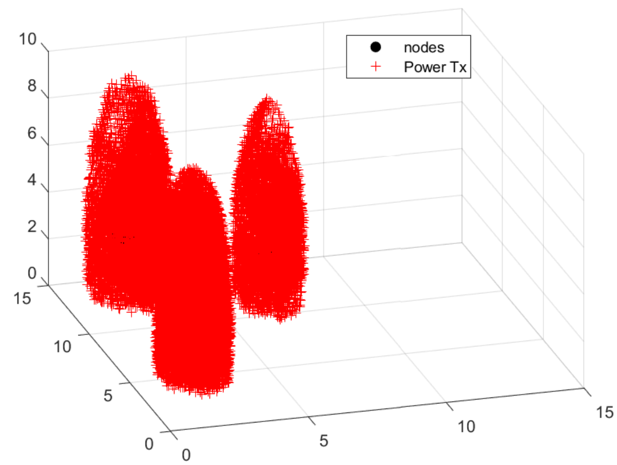
5.3.2 Second Step: Generate Initial PTX Locations

In the second step, the PTX initial location and their coverage region is generated. This is done using equation (5-1) and equation (5-2) where d_{ant} is set to 0.3 meter and r_{ant} is set to 15 degrees. The values for d_{ant}, r_{ant} were chosen to allow the simulation to be completed in a reasonable time. For better precision, lower values are recommended, but at the cost of increased simulation runtime. After several iterations, the chosen values are deemed to be accurate enough for this thesis and give a good trade-off between execution speed and value accuracy. For this step, the initial PTX locations generated for all 3 distributions will be the same because they are generated from the same equations (i.e. equation (5-1) and equation (5-2)). There are more PTXs for a directional antenna than for the omni-directional antenna because when using the directional antenna there are many possible rotation angles. To be precise, for every initial PTX location for omni-directional antennas, there are $(360/15) \times 3 = 72$ initial PTX location for directional antennas. This figure shows that with initial PTX locations, the entire plant space is covered.

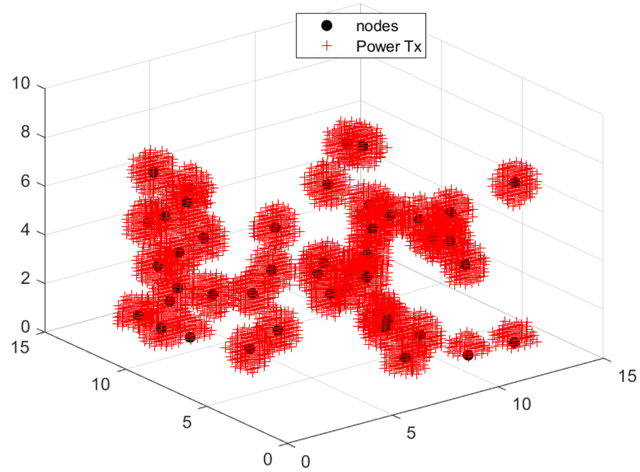
Then, all PTX locations that do not cover any nodes are removed from the list, as illustrated in Figure 5-8. From the figure, it can be seen that the number of initial PTX locations decreases dramatically, as expected. Now, all PTX locations present in the figure power at least one node in a LOS environment. At this point, the list of all initial PTX locations that cover at least one node can be generated.



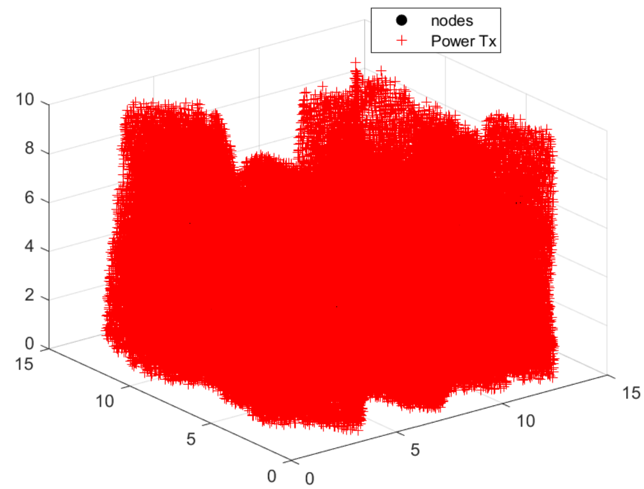
(a) clumped with omni-directional antennas



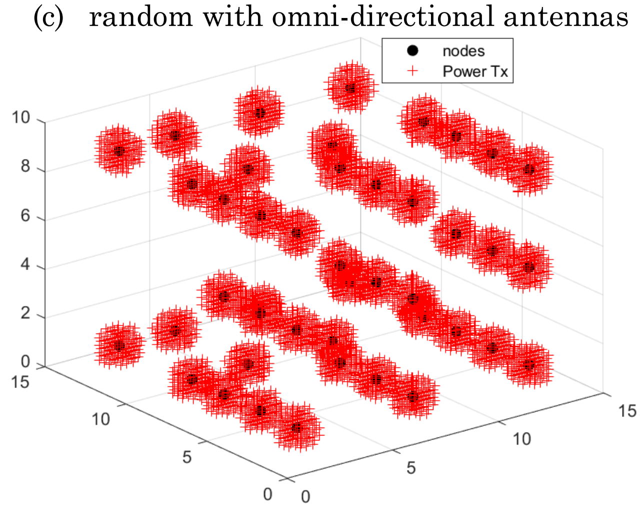
(b) clumped with directional antennas



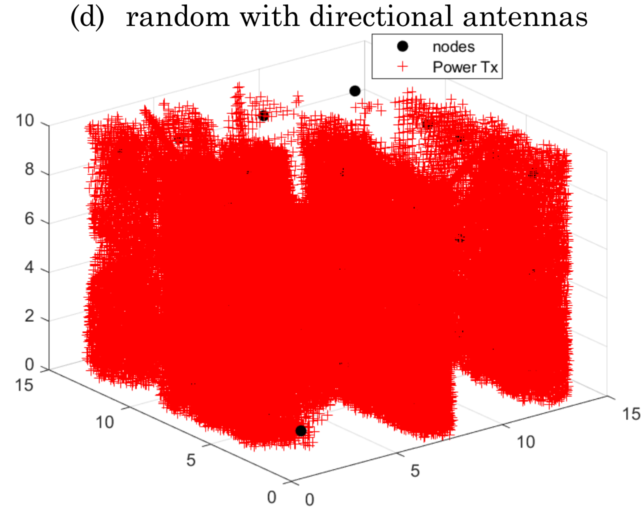
(c) random with omni-directional antennas



(d) random with directional antennas



(e) uniform pattern with omni-directional antennas

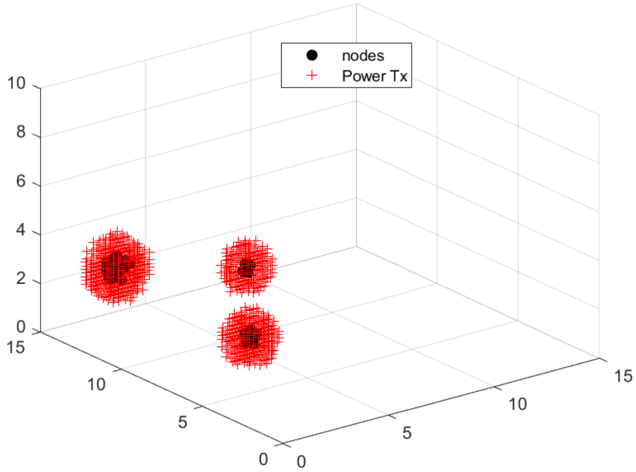


(f) uniform pattern with directional antennas

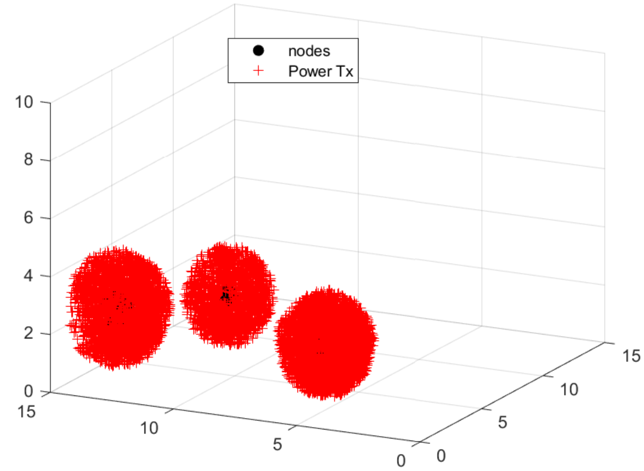
Figure 5-8. 3D visualization of feasible PTX locations in LOS for different distributions and antenna types

5.3.3 Third Step: Determine Feasible PTX Locations

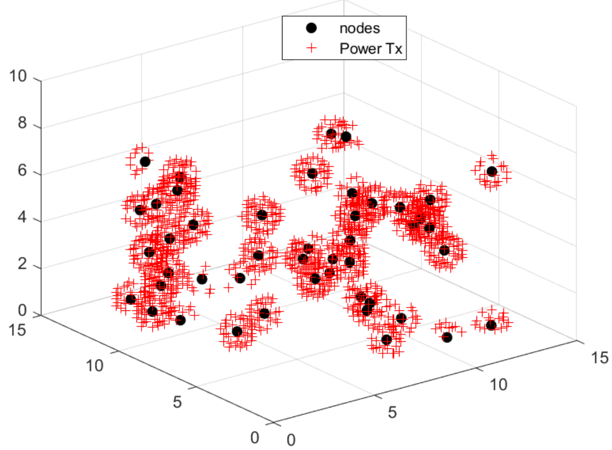
In the third step, attenuation is applied, and all PTX locations that cannot provide sufficient power to at least one node are removed, as illustrated Figure 5-9. In fact, step 2 only considers the distance between the PTX and the node to verify if a node is within a PTX coverage region. In this step, the power required by the node is compared with the power needed from the PTX. At this point the distance between a node and a PTX has been replaced with required and actual incident power at each node. At the end of this step, only the feasible PTX locations remain.



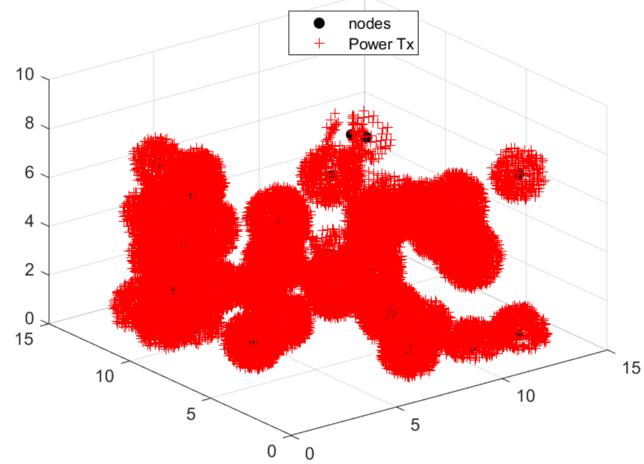
(a) clumped with omni-directional antennas



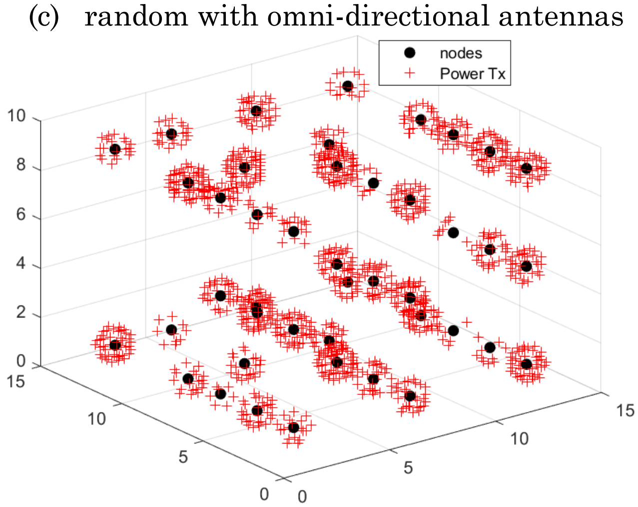
(b) clumped with directional antennas



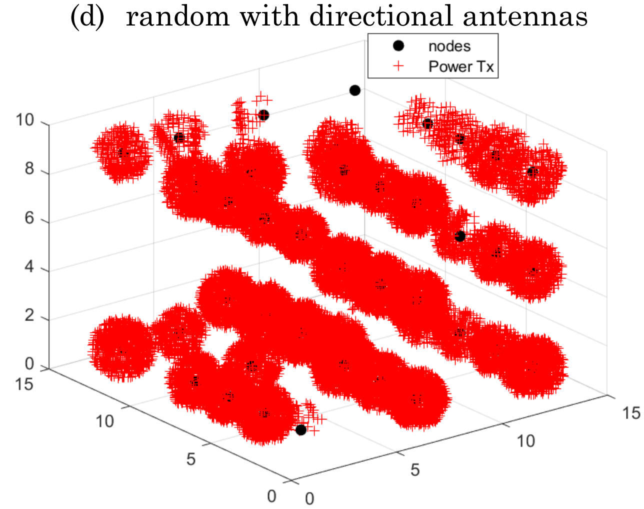
(c) random with omni-directional antennas



(d) random with directional antennas



(e) uniform pattern with omni-directional antennas



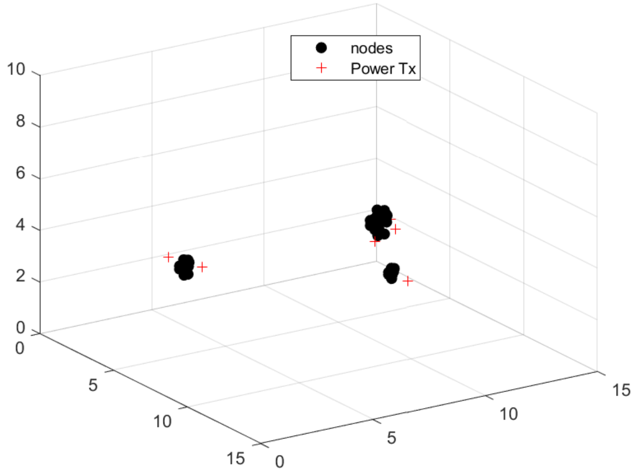
(f) uniform pattern with directional antennas

Figure 5-9. 3D visualization of feasible PTX locations that cover nodes for different distributions and antenna types

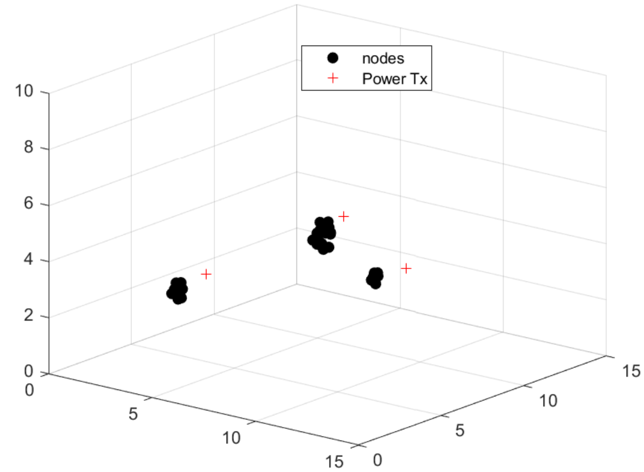
5.3.4 Discussion

In the fourth step, the NCCs are generated and the final PTX locations are defined, as illustrated in Figure 5-10. This last step selects the optimal PTX locations that maximises the cost efficiency of the system.

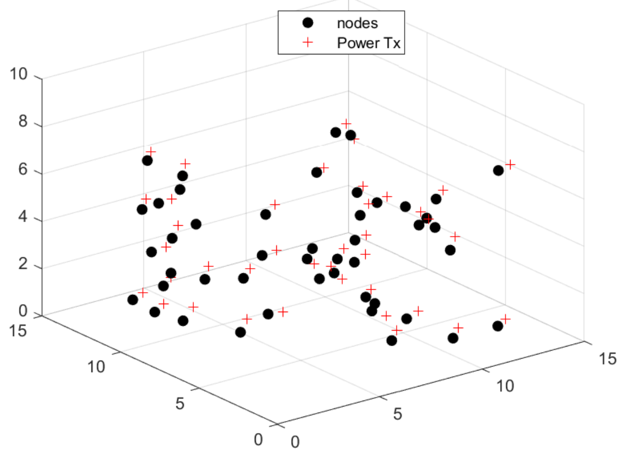
Results are illustrated in Figure 5-11, where the cost per node for an expected lifetime of 10 years is determined. In general, the more attenuation the system encounters, the less cost efficient the system is. This is to be expected, as attenuation results in an increase in the number of required PTX nodes. When an omni-directional antenna is used, nodes cannot be powered via RFEH regardless of architecture when the attenuation material is concrete. The PTX is simply incapable of providing sufficient power to nodes for any physical configuration. Some nodes are capable of being powered using a directional antenna, but this is likely due to low enough attenuation values for the particular power signals being received by those nodes, making a power transmission link feasible.



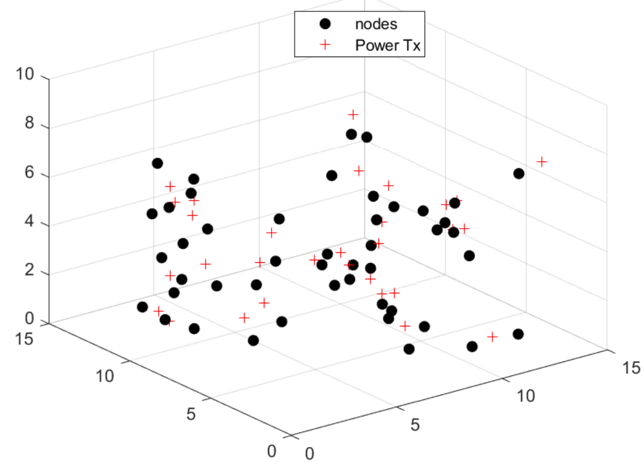
(a) clumped with omni-directional antennas



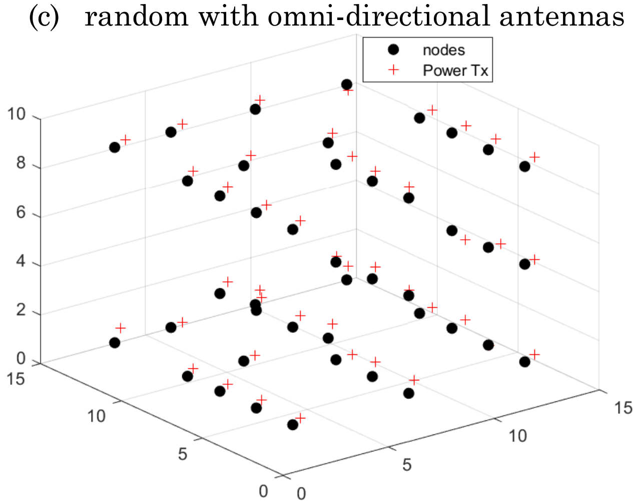
(b) clumped with directional antennas



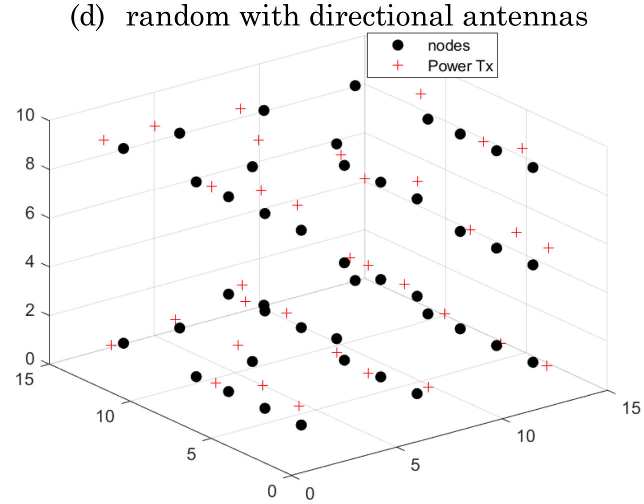
(c) random with omni-directional antennas



(d) random with directional antennas

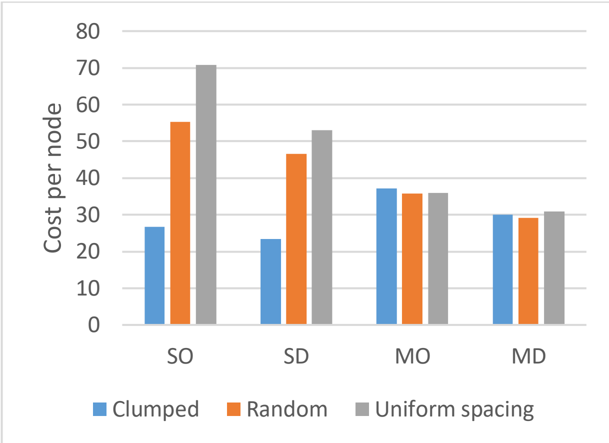


(e) uniform pattern with omni-directional antennas

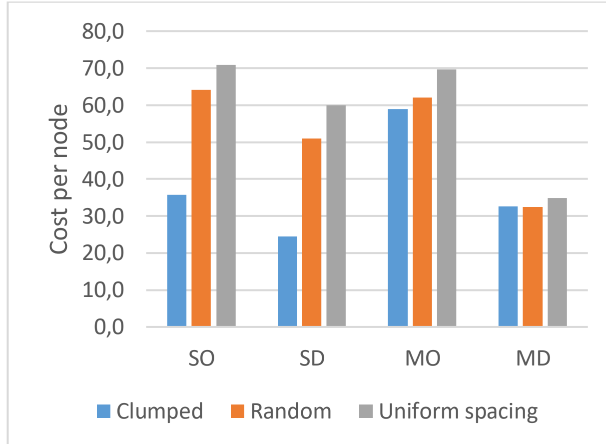


(f) uniform pattern with directional antennas

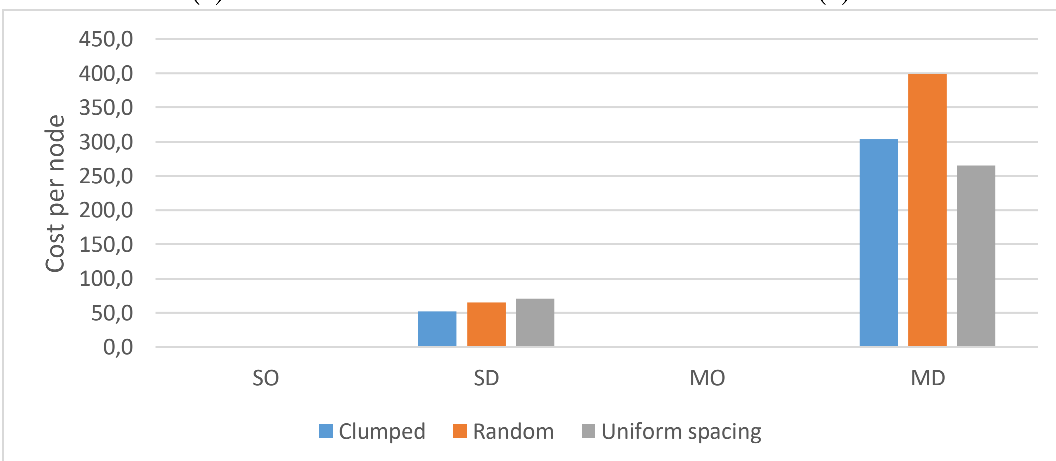
Figure 5-10. 3D visualization of final PTX locations relative to the nodes for stationary architecture for different distributions and antenna types



(a) LOS



(b) brick



(c) concrete

Figure 5-11. System efficiency in different architectures for different PTSA materials

Erreur ! Référence non valide pour un signet., Table 5-7 and Table 5-8 list the number of PTXs for each NCC value for different types of architectures/antennas, attenuation, and distribution types. These tables are followed by a detailed explanation about why the results in Figure 5-11 have been obtained.

Table 5-6. Number of PTXs in each NCC for clumped distribution

Attenuation	SO		SD		MO		MD	
	# PTX	**NCC #**	**# PTX**	**NCC #**	**# PTX**	**NCC #**	**# PTX**	**NCC #**
LOS	1	17	1	23	6	7	1	45
	1	13	1	13	1	3		
	1	10	1	9				
	1	5						
	1	3						
	1	2						
Total	6	45	3	45	7	45	1	45
Brick	1	11	1	22	3	3	1	17
	1	8	1	13	14	2	2	14
	1	4	1	9	8	1		
	2	3	1	1				
	1	2						
	7	1						
Total	13	45	4	45	25	45	3	45
Concrete	NA	NA	1	4	NA	NA	1	2
			4	1			43	1
Total	0	0	5	8	0	0	44	45

Erreur ! Référence non valide pour un signet., Table 5-7 and Table 5-8 list the number of PTXs for each NCC value for different types of architectures/antennas, attenuation, and distribution types. These tables are followed by a detailed explanation about why the results in Figure 5-11 have been obtained.

Table 5-6 tabulates the following results (**R1** through **R15**):

- **R1.** For LOS there are 6 PTXs required for SO, 3 PTXs for SD, 7 PTXs for MO and 1 PTX for MD.
- **R2.** For bricks there are 13 PTXs required for SO, 4 PTXs for SD, 25 PTXs for MO and 3 PTXs for MD.
    - **R2.1.** The increase in number of PTXs required for each type of architecture compared to the result in R1 is normal because in LOS there is no attenuation, which allows PTXs to be placed further from nodes and lead to higher NCC

values. In LOS for instance, the highest NCC for a PTX is 17 compared to 11 for bricks.

- **R2.2.** The biggest jump between R1 and R2 occurs for MO, where the number of PTXs goes from 7 to 21. This is explained by the increase in charge time for individual nodes. Indeed, after simulation, the charging time of nodes for LOS was between 436 – 540 hours and the charging time of nodes for LOS range between 78 – 83 hours. This means that PTXs have spent more time charging nodes when the attenuation material was brick.

- **R3.** For concrete, there are no nodes that can be powered when using SO and MO. This is because all power launched by PTXs is completely attenuated before it can arrive at a node.
  - **R3.1.** For SD, only 8/45 nodes can be successfully powered because the power is greatly attenuated. Nonetheless, because power launched via a directional antenna is concentrate in a narrower direction, some nodes (those with lower attenuation within the range) can be powered.
  - **R3.2.** Finally, with MD, all nodes are successfully powered. However, 43 of them are stationary. This is because the charge time of individual nodes range between 1253 – 3683 hours.
- **R4.** The number of PTXs required for SO is lower than the number required for MO for all attenuation materials. This is because in SO, nodes are in close proximity, which results in higher NCC values. Hence, less PTXs. In addition, the simulator's mobile algorithm only considers STSN, which means that a PTX can only charge one node at a time. If STMN was used, the results would certainly be different.

- **R5.** For this case, directional antennas require less PTXs than omni-directional antennas for stationary architecture. This is surprising, as it was expected that omni-directional antennas would be more efficient due to the close proximity of nodes within a group. This result is explained by a small maximum allowable distance between the PTX and the nodes, which decreased the coverage regions of the omni-directional antennas, and thus, the number of nodes that could fit within the coverage region of a single PTX. Increasing the PTX's input power might change this result.

Table 5-7. Number of PTXs in each NCC for random distribution

Attenuation	SO		SD		MO		MD	
	# PTX	NCC #	# PTX	NCC #	# PTX	NCC #	# PTX	NCC #
LOS	3	3	1	5	5	7	1	45
	8	2	2	4	1	5		
	20	1	3	3	1	5		
			6	2				
			11	1				
Total	31	45	23	45	7	45	1	45
Brick	7	2	1	4	2	3	1	16
	31	1	3	3	17	2	1	14
			7	2	5	1	1	10
			19	1			1	5
Total	40	45	30	45	24	45	4	45
Concrete	NA	NA	12	1	NA	NA	1	2
							43	1
Total	0	0	12	12	0	0	44	45

Table 5-7 tabulates the following results (R):

- **R6.** For LOS there are 31 PTXs required for SO, 23 PTXs for SD, 7 PTXs for MO and 1 PTX for MD.
- **R7.** For bricks there are 40 PTXs required for SO, 30 PTXs for SD, 24 PTXs for MO and 4 PTXs for MD.
  - **R7.1.** Results similar to R2.1 can be observed for the random distribution (i.e. there is an increase in the number of PTXs for brick compared to LOS).

- **R7.2.** Results similar to R2.2 can be observed for the random distribution. In this case the charge time for MO range between 436 – 559 hours and for MD range between 78 – 87 hours.

- **R8.** Similar to R3, when using concrete there are no nodes that can be powered when using SO and MO. This is because all the power launched by the PTXs is completely attenuated before arriving at a node.
  - **R8.1.** Only 12/45 nodes are successfully powered. This is due to the same reason as in R3.1.
  - **R8.2.** With MD, all nodes are successfully powered, but 43 of them are stationary, exactly as in R3.2. The charge time in this case ranges between 1280 – 2905 hours.
- **R9.** The number of PTXs required for SO is higher than the number required for MO for all attenuation materials. This is different from what was found in R4. This shows the difference between the clumped and random distributions. When nodes are randomly distributed, they are not necessarily in close proximity, which results in PTXs with lower NCC values. Hence more PTXs.
- **R10.** For this case, directional antennas require less PTXs than omni-directional antennas for stationary architecture. This is expected because the power launched via a directional antenna is concentrated in a narrower direction that allows the PTXs to be placed further away from nodes. Hence, higher NCC values are obtained. Nodes are not necessarily clumped together, which is where omni-directional antennas work well.

Table 5-8. Number of PTXs in each NCC for uniform distribution

Attenuation	SO		SD		MO		MD	
	# PTX	**NCC #**	**# PTX**	**NCC #**	**# PTX**	**NCC #**	**# PTX**	**NCC #**
LOS	45	1	15	2	6	7	1	28
			15	1	1	3	1	17
Total	45	45	30	45	7	45	1	45
Brick	45	1	11	2	2	3	2	14
			23	1	17	2	1	13
					5	1	1	4
Total	45	1	34	45	24	45	4	45
Concrete	NA	NA	10	1	NA	NA	2	2
							41	1
Total	0	0	12	10	0	0	44	45

Table 5-8 tabulates the following results (R):

- **R11.** For LOS there are 45 PTXs required for SO, 30 PTXs for SD, 7 PTXs for MO and 1 PTX for MD.
- **R12.** For bricks there are 45 PTXs required for SO, 34 PTXs for SD, 24 PTXs for MO and 4 PTXs for MD.
    - **R12.1.** For the uniform distribution, the increase in the number of PTXs for brick compared to LOS only happens for the mobile architecture. In stationary architecture, there is no difference. This is because when nodes are far from each other, the PTX coverage regions are not large enough to cover multiple nodes.
    - **R12.2.** For MO and MD, the increase in the number of PTXs between LOS and brick attenuation still happens similar to the results in R2.2 and R7.2 In this case, the charge time for MO ranges between 438 – 590 hours and MD ranges between 78 – 100 hours.
- **R13.** Similar to R3 and R8 for concrete, there are no nodes that can be powered when using SO and MO. This is because all power launched by the PTXs is attenuated to levels where the power no longer reach the node.

- R13.1. Only 12/45 nodes are successfully powered. This is due to the same reason as in R3.1 and R8.1.
- R13.2. With MD, all nodes are successfully powered. However, 41 of them are stationary, similar to R3.2 and R8.2. The charge time in this case ranges between 1258 – 3416 hours.

- **R14.** The number of PTXs required for SO is higher than the number required for MO for all attenuation materials, similar to R9. This is because of reason R9 (i.e. PTXs have lower NCC values).
- **R15.** For this case, directional antennas require less PTXs than omni-directional antennas for stationary architecture based on the same reason as in R10.

5.3.5 Impact of Duty Cycle

The impact of the device duty cycle can be summarized by saying that the more time a node spends in active mode, the less cost efficient the system is. If the duty cycle is increased to 10s/h, architectures with omni-directional antennas become ineffective while the system efficiency of architectures with directional antennas decreases. When nodes consume more energy, the PTX-node distance must be shortened to account for the higher energy charging requirements, as seen in Table 5-9. Alternatively, additional PTX devices can be used to compensate, further increasing system cost.

Table 5-9. System cost efficiency in different architectures and node distributions for duty cycle of 10s/h

Attenuation	Node distribution		SO	SD	MO	MD
LOS	Clumped	System cost efficiency	NA	24.5	NA	36.57
		Unreachable nodes (%)	100	0	100	0
	Random	System cost efficiency	NA	52.01	NA	32.41
		Unreachable nodes (%)	100	0	100	0
	Uniform spacing	System cost efficiency	NA	67.15	NA	37.7
		Unreachable nodes (%)	100	0	100	0
Brick	Clumped	System cost efficiency	NA	30.15	NA	48.38
		Unreachable nodes (%)	100	0	100	0
	Random	System cost efficiency	NA	58.64	NA	47.8
		Unreachable nodes (%)	100	0	100	0
	Uniform spacing	System cost efficiency	NA	70.9	NA	51
		Unreachable nodes (%)	100	0	100	0
Concrete	Clumped	System cost efficiency	NA	NA	NA	NA
		Unreachable nodes (%)	100	100	100	100
	Random	System cost efficiency	NA	NA	NA	NA
		Unreachable nodes (%)	100	100	100	100
	Uniform spacing	System cost efficiency	NA	NA	NA	NA
		Unreachable nodes (%)	100	100	100	100

5.4 Case Study Example

This section examines the case of randomly fixed nodes and provides a visual development of the node locations. For this case, a *rectangular* manufacturing plant measuring, *15 × 15 × 10 meters*, with *46* CM nodes placed *randomly* and affected by attenuation caused by *bricks*. The nodes are homogeneous, and each of them consumes *9.7 µWh (35 mJ)*, have a duty cycle of *1s/h*, an energy storage of *3.6V, 10mAh (129 600 J)*, and an antenna receiver having a *6.1 dBi gain*. Each PTX has a power output of *3W* and works at a frequency of *915 MHz*. The cost of implementation is the following: cost of individual transmitters is *$300* [105], cost of individual receivers is *$300* [106], [107], [108], [109], [110], cost of installing each PTX station is *$200*, and cost of human labour is *$25/h* [111]. By using equation (**2-3**), the maximum PTX-node distances are calculated. For this case, the values are calculated to be 0.92m for an omni-directional antenna and 2.3m for a directional antenna.

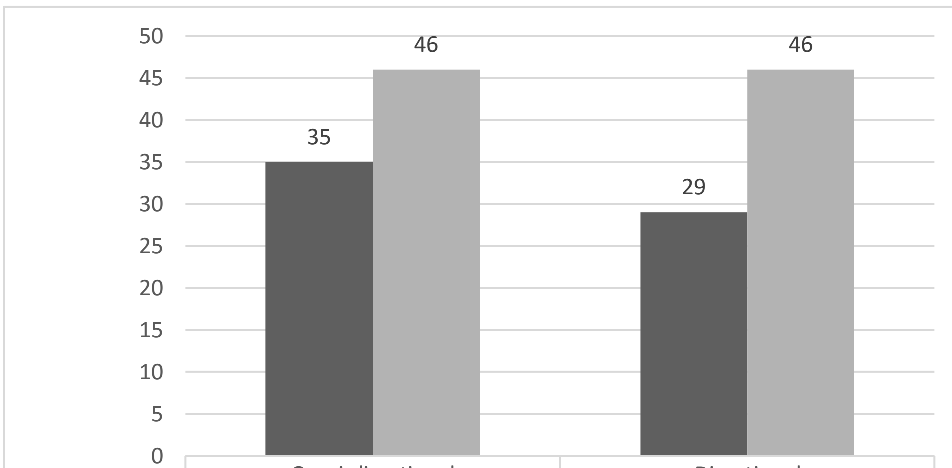
Table 5-10 provides the coordinates of each node with their attenuation values. The step-by-step implementation procedure, with associated figures, for this case can be found in Appendix C.

Table 5-10. Spatial location of nodes and attenuation values

#	Location (m)	Attenuation (dB)	#	Location (m)	Attenuation (dB)
1	[2.5,12.8,7.7]	4	24	[10.4,5.7,1.4]	5
2	[2.4,10.4,3.6]	5	25	[10.4,6.3,3.5]	3
3	[3.6,12.8,2.9]	7	26	[8.3,2.2,3.3]	7
4	[4.4,11.9,5.5]	3	27	[10.4,4,1.18]	7
5	[4.5,8.5,8.2]	7	28	[9.2,4.6,3.9]	7
6	[2.6,11.6,8.2]	6	29	[3.3,5.1,0.8]	4
7	[3.3,11.6,5.3]	4	30	[7.5,5.2,5.5]	5
8	[3.3,11.4,3]	3	31	[5.8,4.9,0.1]	3
9	[3.4,11,7.7]	5	32	[3,5.2,5.1]	5
10	[2.7,9.5,4]	5	33	[3.6,3.7,7.1]	4
11	[2.2,11.9,8.1]	3	34	[4.2,4.6,2.1]	3
12	[3.1,9.5,0.3]	5	35	[9.4,2.5,4]	3
13	[2.4,10.1,4.8]	4	36	[6.7,2.2,5.1]	5
14	[2.1,12.5,6.4]	4	37	[9.6,6.4,0.5]	3
15	[9.4,9.6,1.7]	5	38	[4.8,3.3,4.4]	5
16	[10.1,9.5,0]	4	39	[2.1,4.7,5.8]	5
17	[9.5,9.1,2.8]	4	40	[7.8,6.3,2]	6
18	[10.1,12.8,6.3]	6	41	[5.0,2.8,7.5]	6
19	[12.4,12.7,5.6]	4	42	[12.4,2.8,8.7]	6
20	[11,12.2,4.9]	7	43	[12,3.5,7.6]	3
21	[8.9,11.8,3.9]	7	44	[6.3,6.2,4.5]	3
22	[11,9.3,2.6]	6	45	[2.3,3.7,2.5]	4
23	[9.8,10.1,4.5]	4	46	[9.4,3.2,6.7]	5

5.4.1 Stationary Architecture Output Discussion

After generating the PTX coverage regions and identifying the nodes covered by each PTX, the most efficient locations for PTX devices, based on the NCC values, are determined. When using stationary architecture, the system requires 35 omni-directional PTX antennas and 29 directional antennas, as illustrated in Figure 5-12. If nodes are placed far apart from each other, it prevents the use of STMN. In this example, there are fewer PTXs required to power multiple nodes because of their proximity. In this example, directional antennas give more efficient system costs than omni-directional antennas. This makes sense because directional antennas concentrate power in a single direction, and the nodes are not in close proximity to each other. As mentioned in step 2 (generation of initial PTX locations), the maximum PTX-node distance for directional antennas is greater than the maximum distance for omni-directional antennas, allowing PTX devices with directional antennas to be placed further apart. Each PTX powers more nodes, resulting in fewer required PTX devices.



	Omni-directional	Directional
# PTX	35	29
# of nodes	46	46

Figure 5-12. Number of final PTXs for different antenna types and stationary architecture

5.4.2 Mobile Architecture Output Discussion

For mobile architecture, optimal PTX locations need to be determined to maximize received power at each node. Then, the time required for each node to charge fully and discharge, can be determined, as well as the trajectory of all PTXs. Finally, the number of PTX devices that can charge the greatest number of nodes possible before returning to charge previous nodes before they are discharged is determined, as discussed in section 5.2.5.

When using the mobile architecture, the system requires 27 PTXs when using omni-directional antennas and 3 PTXs when using directional antennas, as illustrated in Figure 5-13. In this case, a directional antenna is more efficient than an omni-directional antenna, because it requires less time for the PTX to charge individual nodes, allowing it to travel to many more nodes before returning to the initial node. Additionally, individual nodes take several days to charge fully, implying that PTXs will not need to move frequently, but only a few times per year. The charging time obtained is reasonable due to the low power input received by each PRX.

While this simulated algorithm requires that each node be charged fully before the PTX can be transferred to another node, another possibility would be to partially charge nodes, potentially requiring fewer PTXs. Only the STSN strategy has been tested in this example. Additional possibilities will be explored in future studies.

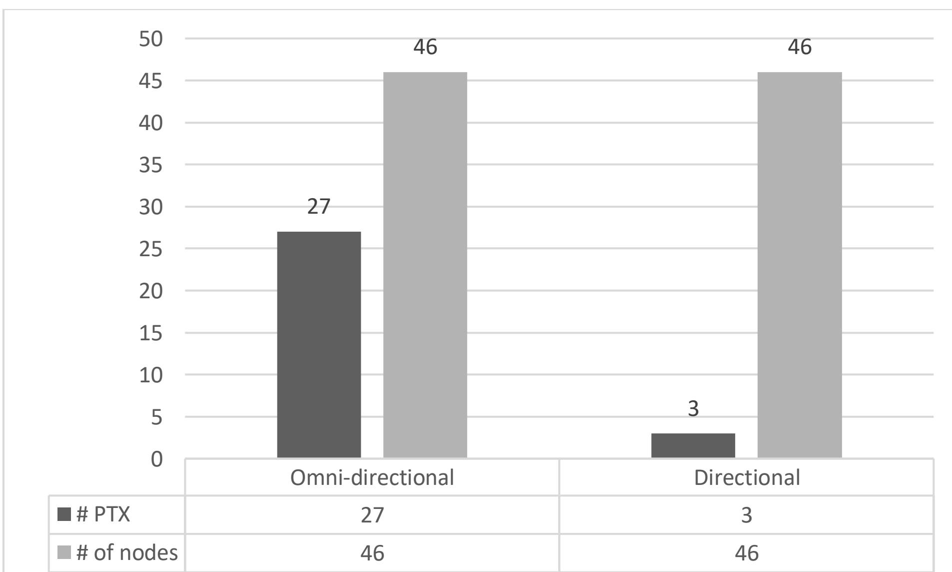


Figure 5-13. Number of final PTXs for different antenna type for mobile architecture

5.4.3 Example Case Discussion

By comparing the results in Figure 5-14**Erreur ! Source du renvoi introuvable.**, it is clear that a directional antenna in a mobile architecture requires the fewest PTXs for the example case presented. This was expected for this case, as a directional antenna provides more power to a specific node or set of nodes, allowing faster far field charging. Based on the obtained results, the system's cost efficiency over time can be calculated. As a reminder, system cost efficiency is defined as the total system cost per node for a specific lifetime. As expected, the results (Figure 5-15**Erreur ! Source du renvoi introuvable.**) indicate that the mobile architecture with a directional antenna is the least expensive of all architectures for the example case presented. However, mobile architectures require the use of a rechargeable battery, which is not always the case with stationary architecture. Battery-less architectures can outlast architectures that require batteries in theory, due to limited battery shelf life. System cost efficiency is increased by lowering the overall cost over time. It then becomes critical to understand the maximum lifetime of a node in the absence of energy storage. The system lifetime would then be limited to the time required for a single component to fail, and thus, for the entire device to fail.

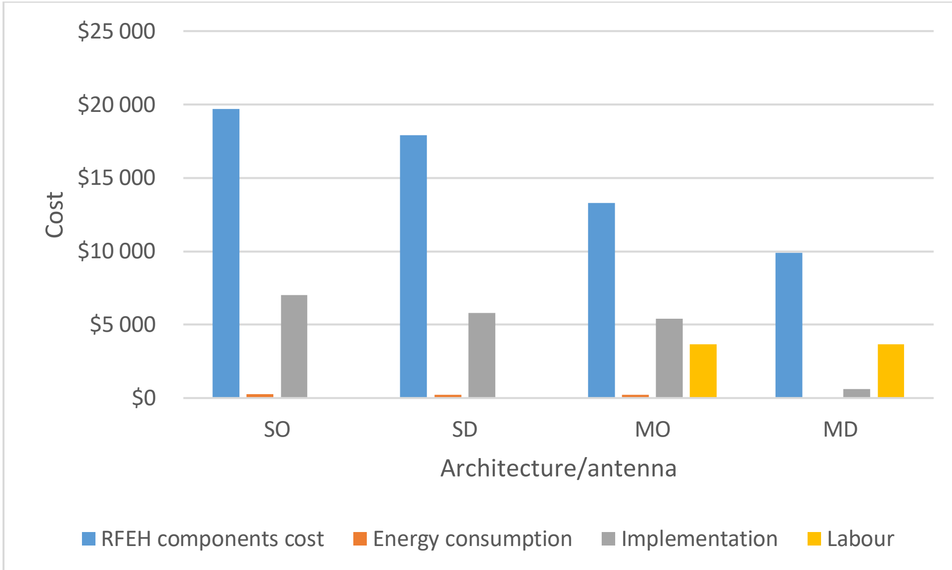


Figure 5-14. Cost of implementation for each architecture

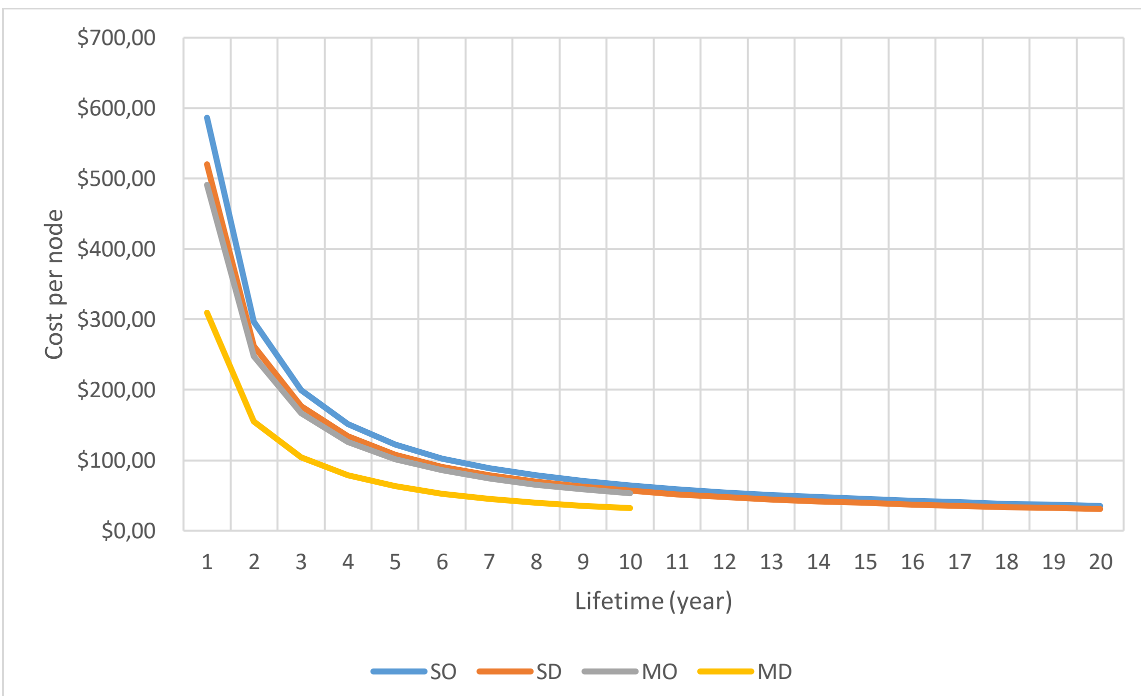


Figure 5-15. Yearly cost per node for different system lifetimes

5.5 Summary

This chapter describes the simulation for the implementation of a WSN powered by RFEH, including plant size, node density, attenuation material, duty cycle, node distribution, and node type. These inputs were used to determine the optimal combination of architecture types and antenna types, for powering a given set of condition monitoring nodes. The simulation then employs the proposed algorithm to determine the system's response to co-variations in node distribution and attenuation material.

There are some limitations to the simulation. In the case of mobile architectures, the simulator does not simulate the STMN strategy. Moreover, for STSN, the trajectory is determined entirely by user input without optimization. This is because the time required to charge a node is expressed in days, minimizing the importance of the PTX's travel time. Nonetheless, three distinct node distributions: clumped, random, and uniform spacing, each with three distinct attenuations: LOS, brick, and concrete were compared.

Input	Result	Reason
Clumped distribution	• SD is more cost efficient for LOS and brick • SO and MO not possible for concrete	• Nodes are at close proximity • Attenuation is too high
Random distribution	• MD is more cost efficient for LOS and brick • Omni-directional antenna is not cost efficient • SO and MO not possible for concrete	• Nodes are not within close proximity of each other • Attenuation is too high
Uniform distribution	• MD is more cost efficient for LOS and brick • Omni-directional antenna is not cost efficient • SO and MO not possible for concrete	• Nodes are not within close proximity of each other • Attenuation is too high
LOS	• Works well for all architectures	• No attenuation
Brick	• MD is more cost efficient • Affects omni-directional antennas • Directional antennas work well	• Localized gain for omni-directional antennas is smaller than directional antennas
Concrete	• Only works for MD	• PTXs can be closer to nodes

In many real plants, nodes are typically randomly distributed, which allows for both clumped and uniform spacing distributions. One method for determining the optimal architecture to use is to divide the entire plant into subzones with a single node distribution (either clumped or uniform spacing). By subdividing the plant, the most cost-efficient architecture for each zone can be utilized to create a more cost-efficient system overall. If this is the case, the simulator can simply be run for each subzone and the optimal power transmission locations for each zone can be determined.

PTSA is the primary impediment to large-scale implementation of RFEH in a WSN. When attenuation is high, the amount of energy transmitted is significantly reduced, making wireless power transfer much less effective. Nonetheless, developing, and optimizing power delivery architectures can help mitigate the effects of attenuation and ensure that nodes receive sufficient power.

This simulator addresses only a subset of the possible scenarios that could occur in a real-world setting. The simulator could be enhanced to account for additional possible scenarios, including the following:

- Analyzing various node types. The nodes in this work are assumed to be homogeneous. This simulator can be enhanced to analyse heterogeneous nodes with varying power consumptions and duty cycles.
- Developing alternate methods for mobile architecture modelling. As previously stated, mobile architectures can be enhanced by incorporating STMN and other optimal path finders.
- The establishment of subzones. The simulator could be enhanced to automatically divide a plant into subzones and determine optimal PTX locations within these zones. To divide the zones efficiently, the simulator must handle multiple parameters, such as the PTX-node distance and power consumption of nodes.

This simulator is used to determine a system's cost efficiency, based on critical parameters when RFEH is implemented in a WSN. The simulation in this study demonstrates only the behaviour of a system by generating optimal PTX locations in a space where all nodes are homogeneous, clumped, randomly or uniformly distributed, and have distinct attenuation ranges. Additionally, this tool can be used to monitor the impact of changing/updating individual components/parameters on the overall system.

Chapter 6

Conclusions and Recommendations for Future Work

6.1 Conclusions

This research was conducted to understand large-scale implementation of RFEH in a WSN for CM applications. The constant advancement of technology, automation, and wireless communication indicates that RFEH has a promising future. Indeed, the production and installation of energy-efficient devices is on the rise, particularly in industrial settings. When these facts are combined with the pressing need to reduce the use of non-rechargeable batteries, this technology might very well find a place in industry in the near future. The goal is to determine whether RFEH can be used to power a CM device, and whether RFEH can be implemented in a large-scale WSN in an efficient and cost-effective manner. CM is an industrial application that is used in a variety of harsh environments, including those with and without LOS, as well as those that are small or large in size. Even with so many environmental parameters to consider, it is possible to make an informed decision about the use of RFEH in a WSN. To address this, a thesis question was presented:

Given n nodes in a WSN, what is the optimal placement of PTXs for use with RFEH to minimize the total number of PTXs required to power all nodes, minimize the total cost of the system and maximize network lifetime.

The PTX is one of the most important components for the implementation of RFEH because it is the primary source of energy for the devices that must capture data related to the condition of a machine. Important specifications related to the PTX have been considered in this thesis, including the amount of power it can provide, as well as the type of antenna that can be used for energy propagation. This study evaluated two types of antennas: omni-directional and directional. Each antenna has properties that allows it to maximise a specific characteristic. For example, an omnidirectional antenna is advantageous when the nodes to be powered are close together around a PTX, while a directional antenna can power nodes that are further away from the PTX, but only in one direction. To determine whether a PTX's energy is sufficient to power a node, two factors must be considered: the energy required for the node to function, and the attenuation of the energy sent by the PTX. Knowing the energy required by a node and the resultant PTSA, the distance at which a PTX must be placed in order to supply this node can be determined. To understand the source and impact of PTSA, various types of materials that can cause PTSA have been included in the study. This work has also demonstrated the importance of node component selection and it effect on the node's energy consumption. Finally, knowing how much energy a node consumes aids in the selection of the appropriate type of energy supply. This study looked at two types of energy supply: harvest-use (RFEH) and harvest-store-use (RFEH/RB or RFEF/NRB). When neither of these two types of energy supply satisfy the demand, the system can fall back on the standard NRB solution already being used by others in industry.

Environmental inputs considered include PTSA, plant size and node location. Plant size, which along with the number of nodes, determines node density. On the other hand, node locations, along with three different node distribution patterns (clumped, random, and uniform) affect final PTX locations within a plant. Finally, two power architectures have been developed for PTX placement: stationary and mobile. By combining these two architectures with different antenna types and implementing a custom simulator, optimal and cost effective PTX locations were determined that minimize the total number of nodes and maximize system lifetime.

This work leads to the conclusion that it is indeed feasible to use RFEH in a WSN, but further enhancements could be included, which would allow it to be even more cost efficient. Several of these enhancements are suggested for future work.

6.2 Recommendations for Future Work

With the rapid development of wireless IoT devices, it is critical to find a way to power them wirelessly as well. To consider RFEH on a large scale, the PTX-node distance must be significantly increased. This can be accomplished by either increasing the amount of energy transmitted, by reducing the amount of energy consumed by the devices themselves, or by using more directional antennas. Each of these strategies should be explored further in the future.

PTSA is a significant impediment to the implementation of RFEH, even more so when transmission distances are long enough to reduce received power levels to very low values. Developing a method to minimise attenuation will be critical for increasing the possible use of RFEH for IoT devices. Additionally, power architectures can be enhanced by combining them, in different ways, depending on the node distribution and environment. Based on the work here, overall cost efficiency of RFEH in WSNs can be improved by:

- Improving the rectenna specifications, as the efficiency of the RF-DC conversion is still insufficient for environments with a duty cycle greater than a few seconds. Enhancing these specifications will result in an increase in the amount of energy received by the PRX.

- Reducing the effect of PTSA, the power received by nodes is increased. Time reversal techniques should be explored as a way to work around the impact of attenuation.
- Increasing the maximum allowable PTX transmitted power (which is currently limited to 4W in North America) will result in increased input power received by the node. Europe has approved the first transmitter with a power output of 10 watts [112]. Further research is required to determine how such higher power levels can be transmitted safely and whether additional techniques are required to disable wireless transmission when humans are too close to such transmitters.
- Reducing the energy consumption of nodes. While great improvements have been made, if this energy consumption can be reduced further, a wider range of power architectures will be feasible than those possible with current technology.
- Using engineered surface with metamaterials can reduce transmission loss through intervening objects. Metamaterials are artificially engineered materials that possess properties not found in nature. Metamaterials can be used as a means of "channeling" the energy from the transmitter to the receiver more efficiently [113].
- Implementing specific dynamic and adaptive techniques such as antenna aperture illumination, or phased arrays to adjust the phase/polarization of antennas for a better beaming efficiency on the transmitter side, can increase the received power on the receiver side.

With more time, many other inputs and parameters could be considered. For this simulation, the node type could be expanded to include heterogeneous nodes, PTSA could be generated around specific areas rather than only being assigned to individual nodes, STMN strategies can be developed for mobile architectures, as well as MT strategies for all architectures. Finally, different types of architecture can be combined to create hybrid architectures that take advantage of each. With the rapid advancement of IoT technology, this simulator can be used beyond CM applications where RFEH can be considered as a power source.

References

[1] S. C. Vegunta and J. V. Milanovic, "Estimation of Cost of Downtime of Industrial Process Due to Voltage Sags," *IEEE Trans. Power Deliv.*, vol. 26, no. 2, pp. 576–587, Apr. 2011, doi: 10.1109/TPWRD.2009.2035366.

[2] P. Brown, "Principal Causes of Failure in Machinery," p. 5.

[3] C. Murphey, "Choosing The Most Suitable Predictive Maintenance Sensor," p. 5.

[4] A. H. Pesch and P. N. Scavelli, "Condition Monitoring of Active Magnetic Bearings on the Internet of Things," *Actuators*, vol. 8, no. 1, Art. no. 1, Mar. 2019, doi: 10.3390/act8010017.

[5] S. J. Watson, B. J. Xiang, W. Yang, P. J. Tavner, and C. J. Crabtree, "Condition Monitoring of the Power Output of Wind Turbine Generators Using Wavelets," *IEEE Trans. Energy Convers.*, vol. 25, no. 3, pp. 715–721, Sep. 2010, doi: 10.1109/TEC.2010.2040083.

[6] D. Goyal, Vanraj, B. S. Pabla, and S. S. Dhami, "Non-contact sensor placement strategy for condition monitoring of rotating machine-elements," *Eng. Sci. Technol. Int. J.*, vol. 22, no. 2, pp. 489–501, Apr. 2019, doi: 10.1016/j.jestch.2018.12.006.

[7] G. Loubet, A. Takacs, and D. Dragomirescu, "Towards the Design of Wireless Communicating Reinforced Concrete," *IEEE Access*, vol. 6, pp. 75002–75014, 2018, doi: 10.1109/ACCESS.2018.2883434.

[8] W. Zhou, T. G. Habetler, and R. G. Harley, "Bearing Condition Monitoring Methods for Electric Machines: A General Review," in *2007 IEEE International Symposium on Diagnostics for Electric Machines, Power Electronics and Drives*, Sep. 2007, pp. 3–6. doi: 10.1109/DEMPED.2007.4393062.

[9] S. Orhan, N. Aktürk, and V. Çelik, "Vibration monitoring for defect diagnosis of rolling element bearings as a predictive maintenance tool: Comprehensive case studies," *NDT E Int.*, vol. 39, no. 4, pp. 293–298, Jun. 2006, doi: 10.1016/j.ndteint.2005.08.008.

[10] "Fluke 3560 FC Vibration Sensor and Fluke 3502 FC Gateway," p. 4.

[11] "ABB Ability Smart Sensor for mechanical products." https://search.abb.com/library/Download.aspx?DocumentID=9AKK107592&LanguageCode=en&DocumentPartId=&Action=Launch (accessed Jul. 24, 2020).

[12] P. Saffari, A. Basaligheh, V. J. Sieben, and K. Moez, "An RF-Powered Wireless Temperature Sensor for Harsh Environment Monitoring With Non-Intermittent Operation," *IEEE Trans. Circuits Syst. Regul. Pap.*, vol. 65, no. 5, pp. 1529–1542, May 2018, doi: 10.1109/TCSI.2017.2758327.

[13] W. Saeed, N. Shoaib, H. M. Cheema, and M. U. Khan, "RF Energy Harvesting for Ubiquitous, Zero Power Wireless Sensors," *Int. J. Antennas Propag.*, vol. 2018, p. e8903139, Apr. 2018, doi: 10.1155/2018/8903139.

[14] S. Y. Hui, "Planar Wireless Charging Technology for Portable Electronic Products and Qi," *Proc. IEEE*, vol. 101, no. 6, pp. 1290–1301, Jun. 2013, doi: 10.1109/JPROC.2013.2246531.

[15] L. Meile, A. Ulrich, and M. Magno, "Wireless Power Transmission Powering Miniaturized Low Power IoT devices: A Revie," in *2019 IEEE 8th International Workshop on Advances in Sensors and Interfaces (IWASI)*, Jun. 2019, pp. 312–317. doi: 10.1109/IWASI.2019.8791433.

[16] M. Aboualalaa and H. Elsadek, "Rectenna Systems for RF Energy Harvesting and Wireless Power Transfer," *Recent Wirel. Power Transf. Technol.*, Oct. 2019, doi: 10.5772/intechopen.89674.

[17] J. Zhang, X. Bai, W. Han, B. Zhao, L. Xu, and J. Wei, "The design of radio frequency energy harvesting and radio frequency-based wireless power transfer system for battery-less self-sustaining applications," *Int. J. RF Microw. Comput.-Aided Eng.*, vol. 29, no. 1, p. e21658, 2019, doi: 10.1002/mmce.21658.

[18] S. Kim *et al.*, "Ambient RF Energy-Harvesting Technologies for Self-Sustainable Standalone Wireless Sensor Platforms," *Proc. IEEE*, vol. 102, no. 11, pp. 1649–1666, Nov. 2014, doi: 10.1109/JPROC.2014.2357031.

[19] X. Lu, P. Wang, D. Niyato, D. I. Kim, and Z. Han, "Wireless Networks With RF Energy Harvesting: A Contemporary Survey," *IEEE Commun. Surv. Tutor.*, vol. 17, no. 2, pp. 757–789, 2015, doi: 10.1109/COMST.2014.2368999.

[20] R. La Rosa, P. Livreri, C. Trigona, L. Di Donato, and G. Sorbello, "Strategies and Techniques for Powering Wireless Sensor Nodes through Energy Harvesting and Wireless Power Transfer," *Sensors*, vol. 19, no. 12, p. 2660, Jun. 2019, doi: 10.3390/s19122660.

[21] Z. Zhang, H. Pang, A. Georgiadis, and C. Cecati, "Wireless Power Transfer—An Overview," *IEEE Trans. Ind. Electron.*, vol. 66, no. 2, pp. 1044–1058, Feb. 2019, doi: 10.1109/TIE.2018.2835378.

[22] "i-Alert | Sensor Portal," *i-Alert | Sensor Portal*. https://www.i-alert.com/ (accessed Apr. 21, 2022).

[23] "Treon Industrial Node 6 Ex." https://www.treonstore.com/treon-industrial-node-6-ex.html (accessed Apr. 21, 2022).

[24] "FLIR SV87 Vibration and Temperature Remote Sensor." https://www.itm.com/product/flir-sv87-vibration-and-temperature-remote-sensor?gclid=Cj0KCQjw3v6SBhCsARIsACyrRAkvyxpMIicHRoCCFgHnxf1nF8LcR-287BRaFLXRRemnRAMaV84tP4kaAjAgEALw_wcB (accessed Apr. 21, 2022).

[25] P. Wright, D. Dornfeld, and N. Ota, "CONDITION MONITORING IN END-MILLING USING WIRELESS SENSOR NETWORKS (WSNs)," vol. 36, p. 9, 2008.

[26] S. Giannoulis, C. Koulamas, C. Emmanouilidis, P. Pistofidis, and D. Karampatzakis, "Wireless Sensor Network Technologies for Condition Monitoring of Industrial Assets," 2013, pp. 33–40.

[27] V. Hodge, S. O'Keefe, M. Weeks, and A. Moulds, "Wireless Sensor Networks for Condition Monitoring in the Railway Industry: A Survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, pp. 1088–1106, Jun. 2015, doi: 10.1109/TITS.2014.2366512.

[28] A. Tripathi, H. P. Gupta, T. Dutta, R. Mishra, K. K. Shukla, and S. Jit, "Coverage and Connectivity in WSNs: A Survey, Research Issues and Challenges," *IEEE Access*, vol. 6, pp. 26971–26992, 2018, doi: 10.1109/ACCESS.2018.2833632.

[29] J. K. Sandhu, A. K. Verma, and P. S. Rana, "A Novel Framework for Reliable Network Prediction of Small Scale Wireless Sensor Networks (SSWSNs)," *Fundam. Informaticae*, vol. 160, no. 3, pp. 303–341, Jan. 2018, doi: 10.3233/FI-2018-1685.

[30] X.-L. Zhang, F. Zhang, J. Yuan, J. Weng, and W. Zhang, "Sensor fault diagnosis and location for small and medium-scale wireless sensor networks," in *2010 Sixth International Conference on Natural Computation*, Aug. 2010, vol. 7, pp. 3628–3632. doi: 10.1109/ICNC.2010.5584224.

[31] A. S. Rostami, M. Badkoobe, F. Mohanna, H. keshavarz, A. A. R. Hosseinabadi, and A. K. Sangaiah, "Survey on clustering in heterogeneous and homogeneous wireless sensor networks," *J. Supercomput.*, vol. 74, no. 1, pp. 277–323, Jan. 2018, doi: 10.1007/s11227-017-2128-1.

[32] O. Kazeem, O. Akintade, and L. Kehinde, “Comparative Study of Communication Interfaces for Sensors and Actuators in the Cloud of Internet of Things,” *Int. J. Internet Things*, vol. 6, pp. 9–13, Jun. 2017, doi: 10.5923/j.ijit.20170601.02.

[33] T. Øvrebekk, “The Importance of Average Power Consumption to Battery Life.” https://blog.nordicsemi.com/getconnected/the-importance-of-average-power-consumption-to-battery-life (accessed May 07, 2021).

[34] “Wi-Fi 6 For Low-Power IoT,” *Semiconductor Engineering*, Jun. 25, 2020. https://www.imgtec.com/blog/how-wi-fi-6-unlocks-the-potential-of-low-power-iot/ (accessed May 23, 2021).

[35] “2.4 vs. 5 GHz WiFi | CenturyLink.” https://www.centurylink.com/home/help/internet/wireless/which-frequency-should-you-use.html (accessed May 23, 2021).

[36] G. Loubet, A. Takacs, and D. Dragomirescu, “Implementation of a Battery-Free Wireless Sensor for Cyber-Physical Systems Dedicated to Structural Health Monitoring Applications,” *IEEE Access*, vol. 7, pp. 24679–24690, 2019, doi: 10.1109/ACCESS.2019.2900161.

[37] “Power Sources for Wireless Sensor Networks,” p. 22.

[38] J. P. Amaro, F. J. T. E. Ferreira, R. Cortesão, and J. Landeck, “Powering Wireless Sensor Networks Nodes for Complex Protocols on Harvested Energy,” *Procedia Technol.*, vol. 5, 2012, Accessed: Apr. 05, 2021. [Online]. Available: https://cyberleninka.org/article/n/1011372

[39] N. Sahraei, E. E. Looney, S. M. Watson, I. M. Peters, and T. Buonassisi, “Adaptive power consumption improves the reliability of solar-powered devices for internet of things,” *Appl. Energy*, vol. 224, pp. 322–329, août 2018, doi: 10.1016/j.apenergy.2018.04.091.

[40] “An indoor power line based magnetic field energy harvester for self-powered wireless sensors in smart home applications - ScienceDirect.” https://www.sciencedirect.com/science/article/pii/S0306261918315265?casa_token=Zc5Dphf1uxoAAAAA:TWPZ_8M9MqixXKaPnBzGxAgu_A8eKQA_XWsq3Hs6g1VkK5EXjEXOx5WDr_TouD_0kyTFJr8kK7U (accessed Mar. 07, 2022).

[41] A. L. Cottrill *et al.*, “Persistent energy harvesting in the harsh desert environment using a thermal resonance device: Design, testing, and analysis,” *Appl. Energy*, vol. 235, pp. 1514–1523, février 2019, doi: 10.1016/j.apenergy.2018.11.045.

[42] “Vibration energy harvesting to power condition monitoring sensors for industrial and manufacturing equipment,” *ResearchGate*. https://www.researchgate.net/publication/258177183_Vibration_energy_harvesting_to_power_condition_monitoring_sensors_for_industrial_and_manufacturing_equipment (accessed Jul. 11, 2020).

[43] C. R. Valenta and G. D. Durgin, “Harvesting Wireless Power: Survey of Energy-Harvester Conversion Efficiency in Far-Field, Wireless Power Transfer Systems,” *IEEE Microw. Mag.*, vol. 15, no. 4, pp. 108–120, Jun. 2014, doi: 10.1109/MMM.2014.2309499.

[44] C. Tong, *Introduction to Materials for Advanced Energy Systems*. Springer, 2018.

[45] S. Sudevalayam and P. Kulkarni, “Energy Harvesting Sensor Nodes: Survey and Implications,” *IEEE Commun. Surv. Tutor.*, vol. 13, no. 3, pp. 443–461, Third 2011, doi: 10.1109/SURV.2011.060710.00094.

[46] G. Anastasi, M. Conti, M. Francesco, and A. Passarella, “Energy Conservation in Wireless Sensor Networks: A Survey,” *Ad Hoc Netw.*, vol. 7, pp. 537–568, May 2009, doi: 10.1016/j.adhoc.2008.06.003.

[47] T. Sanislav, G. D. Mois, S. Zeadally, and S. C. Folea, "Energy Harvesting Techniques for Internet of Things (IoT)," *IEEE Access*, vol. 9, pp. 39530–39549, 2021, doi: 10.1109/ACCESS.2021.3064066.

[48] V. T. Nguyen, S. H. Kang, J. H. Choi, and C. W. Jung, "Magnetic resonance wireless power transfer using three-coil system with single planar receiver for laptop applications," *IEEE Trans. Consum. Electron.*, vol. 61, no. 2, pp. 160–166, May 2015, doi: 10.1109/TCE.2015.7150569.

[49] T. S. Rappaport, *Wireless Communications: Principles and Practice*, Subsequent edition. Upper Saddle River, N.J: Prentice Hall, 2001.

[50] X. Lu, D. Niyato, P. Wang, D. I. Kim, and Z. Han, "Wireless charger networking for mobile devices: fundamentals, standards, and applications," *IEEE Wirel. Commun.*, vol. 22, no. 2, pp. 126–135, Apr. 2015, doi: 10.1109/MWC.2015.7096295.

[51] M. Budhia, G. A. Covic, and J. T. Boys, "Design and Optimization of Circular Magnetic Structures for Lumped Inductive Power Transfer Systems," *IEEE Trans. Power Electron.*, vol. 26, no. 11, pp. 3096–3108, Nov. 2011, doi: 10.1109/TPEL.2011.2143730.

[52] A. Karalis, J. D. Joannopoulos, and M. Soljačić, "Efficient wireless non-radiative mid-range energy transfer," *Ann. Phys.*, vol. 323, no. 1, pp. 34–48, Jan. 2008, doi: 10.1016/j.aop.2007.04.017.

[53] M. P. Aparicio, A. Bakkali, J. Pelegri-Sebastia, T. Sogorb, and V. L. and A. Bou, "Radio Frequency Energy Harvesting - Sources and Techniques," *Renew. Energy - Util. Syst. Integr.*, May 2016, doi: 10.5772/61722.

[54] P. Nintanavongsa, U. Muncuk, D. R. Lewis, and K. R. Chowdhury, "Design Optimization and Implementation for RF Energy Harvesting Circuits," *IEEE J. Emerg. Sel. Top. Circuits Syst.*, vol. 2, no. 1, pp. 24–33, Mar. 2012, doi: 10.1109/JETCAS.2012.2187106.

[55] M. Piñuela, P. D. Mitcheson, and S. Lucyszyn, "Ambient RF Energy Harvesting in Urban and Semi-Urban Environments," *IEEE Trans. Microw. Theory Tech.*, vol. 61, no. 7, pp. 2715–2726, Jul. 2013, doi: 10.1109/TMTT.2013.2262687.

[56] "Rectenna Serves 2.45-GHz Wireless Power Transmission (.PDF Download)," *Microwaves & RF*, Sep. 18, 2014. https://www.mwrf.com/home/whitepaper/21847447/rectenna-serves-245ghz-wireless-power-transmission-pdf-download (accessed Feb. 10, 2021).

[57] E. Falkenstein, M. Roberg, and Z. Popovic, "Low-Power Wireless Power Delivery," *IEEE Trans. Microw. Theory Tech.*, vol. 60, no. 7, pp. 2277–2286, Jul. 2012, doi: 10.1109/TMTT.2012.2193594.

[58] A. Costanzo and D. Masotti, "Smart Solutions in Smart Spaces: Getting the Most from Far-Field Wireless Power Transfer," *IEEE Microw. Mag.*, vol. 17, no. 5, pp. 30–45, May 2016, doi: 10.1109/MMM.2016.2525119.

[59] A. M. Zungeru, L.-M. Ang, S. Prabaharan, and K. P. Seng, "Radio Frequency Energy harvesting and Management for Wireless Sensor Networks," p. 40.

[60] K. Rajeswari, Sujitha. S, Hinduja. S, and Vivitha. C, "Analysis of the Efficiency of RF to DC Conversion using an Energy harvesting device," in *2019 International Conference on Recent Advances in Energy-efficient Computing and Communication (ICRAECC)*, Mar. 2019, pp. 1–7. doi: 10.1109/ICRAECC43874.2019.8995099.

[61] W. A. Serdijn, A. L. R. Mansano, and M. Stoopman, "Chapter 4.2 - Introduction to RF Energy Harvesting," in *Wearable Sensors*, E. Sazonov and M. R. Neuman, Eds. Oxford: Academic Press, 2014, pp. 299–322. doi: 10.1016/B978-0-12-418662-0.00019-2.

[62] C. Song, Y. Huang, J. Zhou, J. Zhang, S. Yuan, and P. Carter, "A High-Efficiency Broadband Rectenna for Ambient Wireless Energy Harvesting," *IEEE Trans. Antennas Propag.*, vol. 63, no. 8, pp. 3486–3495, Aug. 2015, doi: 10.1109/TAP.2015.2431719.

[63] F. Xie, G. Yang, and W. Geyi, "Optimal Design of an Antenna Array for Energy Harvesting," *IEEE Antennas Wirel. Propag. Lett.*, vol. 12, pp. 155–158, 2013, doi: 10.1109/LAWP.2013.2243697.

[64] D. Ho, I. Kharrat, V. Ngo, T. Vuong, Q. Nguyen, and M. Le, "Dual-band rectenna for ambient RF energy harvesting at GSM 900 MHz and 1800 MHz," in *2016 IEEE International Conference on Sustainable Energy Technologies (ICSET)*, Nov. 2016, pp. 306–310. doi: 10.1109/ICSET.2016.7811800.

[65] "Energy Harvesting Using a Low-Cost Rectenna for Internet of Things (IoT) Applications - IEEE Journals & Magazine." https://ieeexplore.ieee.org/document/8355964 (accessed May 13, 2020).

[66] Z. Hameed and K. Moez, "Design of impedance matching circuits for RF energy harvesting systems," *Microelectron. J.*, vol. 62, pp. 49–56, Apr. 2017, doi: 10.1016/j.mejo.2017.02.004.

[67] D. A. Govind, "Antenna Impedance Matching – Simplified," p. 5.

[68] "Understanding Matching Networks | Selected Topics | Electronics Textbook." https://www.allaboutcircuits.com/textbook/radio-frequency-analysis-design/selected-topics/understanding-matching-networks/ (accessed Mar. 03, 2021).

[69] "Full Wave Rectifier and Bridge Rectifier Theory," *Basic Electronics Tutorials*, Aug. 13, 2013. https://www.electronics-tutorials.ws/diode/diode_6.html (accessed Apr. 09, 2021).

[70] R. Ge, H. Pan, Z. Lin, N. Gong, J. Wang, and X. Chen, "RF-powered battery-less Wireless Sensor Network in structural monitoring," in *2016 IEEE International Conference on Electro Information Technology (EIT)*, May 2016, pp. 0547–0552. doi: 10.1109/EIT.2016.7535298.

[71] S. D. Assimonis, S.-N. Daskalakis, and A. Bletsas, "Efficient RF harvesting for low-power input with low-cost lossy substrate rectenna grid," in *2014 IEEE RFID Technology and Applications Conference (RFID-TA)*, Tampere, Finland, Sep. 2014, pp. 1–6. doi: 10.1109/RFID-TA.2014.6934190.

[72] V. Talla, B. Kellogg, B. Ransford, S. Naderiparizi, S. Gollakota, and J. R. Smith, "Powering the Next Billion Devices with Wi-Fi," p. 13.

[73] L. Xie, Y. Shi, Y. T. Hou, and H. D. Sherali, "Making Sensor Networks Immortal: An Energy-Renewal Approach With Wireless Power Transfer," *IEEEACM Trans. Netw.*, vol. 20, no. 6, pp. 1748–1761, Dec. 2012, doi: 10.1109/TNET.2012.2185831.

[74] L. Xie, Y. Shi, Y. T. Hou, W. Lou, H. D. Sherali, and S. F. Midkiff, "Multi-Node Wireless Energy Charging in Sensor Networks," *IEEEACM Trans. Netw.*, vol. 23, no. 2, pp. 437–450, Apr. 2015, doi: 10.1109/TNET.2014.2303979.

[75] L. He, L. Kong, Y. Gu, J. Pan, and T. Zhu, "Evaluating the On-Demand Mobile Charging in Wireless Sensor Networks," *IEEE Trans. Mob. Comput.*, vol. 14, no. 9, pp. 1861–1875, Sep. 2015, doi: 10.1109/TMC.2014.2368557.

[76] L. Persons, "The Basics of Signal Attenuation," p. 9.

[77] F. Cangialosi, T. Grover, P. Healey, T. Furman, A. Simon, and S. M. Anlage, "Time reversed electromagnetic wave propagation as a novel method of wireless power transfer," in *2016 IEEE Wireless Power Transfer Conference (WPTC)*, Aveiro, Portugal, May 2016, pp. 1–4. doi: 10.1109/WPT.2016.7498832.

[78] "Antenna Theory - Radiation Pattern." https://www.tutorialspoint.com/antenna_theory/antenna_theory_radiation_pattern.htm (accessed Sep. 25, 2021).

[79] K. F. Lee, *Principles of antenna theory*. 1984. Accessed: Oct. 17, 2021. [Online]. Available: https://ui.adsabs.harvard.edu/abs/1984jwl..book.....L

[80] S. Shao, K. Gudan, and J. J. Hull, “A Mechanically Beam-Steered Phased Array Antenna for Power-Harvesting Applications [Antenna Applications Corner],” *IEEE Antennas Propag. Mag.*, vol. 58, no. 3, pp. 58–64, Jun. 2016, doi: 10.1109/MAP.2016.2541605.
[81] J. J. Maciel, J. F. Slocum, J. K. Smith, and J. Turtle, “MEMS Electronically Steerable Antennas for Fire Control Radars,” *IEEE Aerosp. Electron. Syst. Mag.*, vol. 22, no. 11, pp. 17–20, Nov. 2007, doi: 10.1109/MAES.2007.4408522.
[82] F. Xie, G.-M. Yang, and W. Geyi, “Optimal Design of an Antenna Array for Energy Harvesting,” *IEEE Antennas Wirel. Propag. Lett.*, vol. 12, pp. 155–158, 2013, doi: 10.1109/LAWP.2013.2243697.
[83] K. Nair *et al.*, “Optimizing power consumption in iot based wireless sensor networks using Bluetooth Low Energy,” in *2015 International Conference on Green Computing and Internet of Things (ICGCIoT)*, Oct. 2015, pp. 589–593. doi: 10.1109/ICGCIoT.2015.7380533.
[84] “efr32bg22-datasheet.pdf.” Accessed: Mar. 23, 2021. [Online]. Available: https://www.silabs.com/documents/public/data-sheets/efr32bg22-datasheet.pdf
[85] “QN902X.pdf.” Accessed: Apr. 21, 2021. [Online]. Available: https://www.nxp.com/docs/en/data-sheet/QN902X.pdf
[86] “Multiprotocol wireless 32-bit MCU Arm®-based Cortex®-M4 with FPU, Bluetooth® 5.2 and 802.15.4 radio solution,” p. 193.
[87] S. Nandi, H. A. Toliyat, and X. Li, “Condition Monitoring and Fault Diagnosis of Electrical Motors—A Review,” *IEEE Trans. Energy Convers.*, vol. 20, no. 4, pp. 719–729, Dec. 2005, doi: 10.1109/TEC.2005.847955.
[88] P. Regtien and E. Dertien, “8 - Piezoelectric sensors,” in *Sensors for Mechatronics (Second Edition)*, P. Regtien and E. Dertien, Eds. Elsevier, 2018, pp. 245–265. doi: 10.1016/B978-0-12-813810-6.00008-2.
[89] “ADXL1001-1002.pdf.” Accessed: Apr. 21, 2021. [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/ADXL1001-1002.pdf
[90] “MODEL 8102 ACCELEROMETER,” p. 4.
[91] “en.DM00476949.pdf.” Accessed: May 23, 2021. [Online]. Available: https://www.st.com/content/ccc/resource/technical/document/datasheet/group3/4e/2a/c2/d1/ab/e0/4c/49/DM00476949/files/DM00476949.pdf/jcr:content/translations/en.DM00476949.pdf
[92] “ADP5300.pdf.” Accessed: Apr. 21, 2021. [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/ADP5300.pdf
[93] “da9061-a_datasheet_2v3.pdf.” Accessed: May 03, 2021. [Online]. Available: https://www.dialog-semiconductor.com/sites/default/files/da9061-a_datasheet_2v3.pdf
[94] “1792082.pdf.” Accessed: May 03, 2021. [Online]. Available: http://www.farnell.com/datasheets/1792082.pdf?_ga=2.43669677.943397721.1620048428-175010406.1601925870&_gac=1.116396788.1620048428.Cj0KCQjw5eX7BRDQARIsAMhYLP-a7bylv-n0N_MGI10vjHDGEgjP2W3qrC5KF8sZaAfLctiFHN12qRYaAsNpEALw_wcB
[95] A. Eraliev and G. Bracco, “Design and Implementation of ZigBee Based Low-Power Wireless Sensor and Actuator Network (WSAN) for Automation of Urban Garden Irrigation Systems,” in *2021 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS)*, Apr. 2021, pp. 1–7. doi: 10.1109/IEMTRONICS52119.2021.9422568.

[96] L. Wang *et al.*, "System level design of wireless sensor node powered by piezoelectric vibration energy harvesting," *Sens. Actuators Phys.*, vol. 310, p. 112039, Aug. 2020, doi: 10.1016/j.sna.2020.112039.
[97] R. L. Rosa, C. Dehollain, A. Burg, M. Costanza, and P. Livreri, "An Energy-Autonomous Wireless Sensor With Simultaneous Energy Harvesting and Ambient Light Sensing," *IEEE Sens. J.*, vol. 21, no. 12, pp. 13744–13752, Jun. 2021, doi: 10.1109/JSEN.2021.3068134.
[98] T. Jang *et al.*, "Circuit and System Designs of Ultra-Low Power Sensor Nodes With Illustration in a Miniaturized GNSS Logger for Position Tracking: Part II—Data Communication, Energy Harvesting, Power Management, and Digital Circuits," *IEEE Trans. Circuits Syst. Regul. Pap.*, vol. 64, no. 9, pp. 2250–2262, Sep. 2017, doi: 10.1109/TCSI.2017.2730638.
[99] I. Government of Canada, "RSS-216 — Wireless Power Transfer Devices," Sep. 15, 2020. https://www.ic.gc.ca/eic/site/smt-gst.nsf/eng/sf10871.html#s6.4 (accessed Apr. 27, 2022).
[100] "Wireless Power Products - Powercastco.com," *Powercast Co.* https://www.powercastco.com/ (accessed Feb. 24, 2022).
[101] "Law Document English View," *Ontario.ca*, Jul. 24, 2014. https://www.ontario.ca/laws/view (accessed Aug. 03, 2021).
[102] "GettingStarted_3561FC_3502FC.book," p. 20, 2019.
[103] I. Slama, B. Jouaber, and D. Zeghlache, "Energy Efficient Scheme for Large Scale Wireless Sensor Networks with Multiple Sinks," in *2008 IEEE Wireless Communications and Networking Conference*, Mar. 2008, pp. 2367–2372. doi: 10.1109/WCNC.2008.417.
[104] "Electricity Prices in Canada 2021," *energyhub.org*, Mar. 11, 2021. https://www.energyhub.org/electricity-prices/ (accessed Apr. 24, 2022).
[105] "TX91501B-1W-ID Powercast Corporation | RF/IF and RFID | DigiKey." https://www.digikey.ca/en/products/detail/powercast-corporation/TX91501B-1W-ID/9992798 (accessed Mar. 28, 2022).
[106] "P2110B Powercast Corporation | Integrated Circuits (ICs) | DigiKey." https://www.digikey.ca/en/products/detail/powercast-corporation/P2110B/9385206 (accessed Apr. 24, 2022).
[107] "PA-915-01 Powercast | Mouser Canada." https://www.mouser.ca/ProductDetail/Powercast/PA-915-01?qs=Xb96R0AS9ef%2FYCmYWPYjEg%3D%3D (accessed Apr. 24, 2022).
[108] "EFR32BG22C224F512GN32-C Silicon Labs | RF/IF and RFID | DigiKey." https://www.digikey.ca/en/products/detail/silicon-labs/EFR32BG22C224F512GN32-C/11610897 (accessed Apr. 24, 2022).
[109] "ADXL1002BCPZ Analog Devices Inc. | Sensors, Transducers | DigiKey." https://www.digikey.ca/en/products/detail/analog-devices-inc/ADXL1002BCPZ/7041452 (accessed Apr. 24, 2022).
[110] "ADP5300ACPZ-2-R7 Analog Devices Inc. | Integrated Circuits (ICs) | DigiKey." https://www.digikey.ca/en/products/detail/analog-devices-inc/ADP5300ACPZ-2-R7/5583189 (accessed Apr. 24, 2022).
[111] "Manufacturing Salary in Canada - Average Salary," *Talent.com*. https://ca.talent.com/salary (accessed Apr. 24, 2022).
[112] "Energous' Latest FCC Certification Nearly Doubles The Power That Can Now Be Transmitted Wirelessly," Dec. 01, 2021. https://www.businesswire.com/news/home/20211201005148/en/Energous%E2%80%99-

Latest-FCC-Certification-Nearly-Doubles-The-Power-That-Can-Now-Be-Transmitted-Wirelessly (accessed Apr. 11, 2022).
[113] "Wireless power transfer with metamaterials | Cambridge Consultants." https://www.cambridgeconsultants.com/insights/wireless-power-transfer-with-metamaterials (accessed Jun. 07, 2022).

Appendix A – Components Datasheet

This appendix provides the datasheet of the components used or considered for this work.

Table A-1. General specifications of the Fluke 3502 FC gateway

Temperature range	-25°C to +65°C
Size	57.3 mm x 39.3 mm x 46.1 mm
Weight	35 g
Power	AC input: 100 V ac to 240 V ac, 50/60 Hz
Bluetooth type	Low energy 4.1
Bluetooth frequency range	2405 MHz to 2480 MHz
Wi-Fi type	802.11 b/g/n
Wi-Fi Frequency range	2412 MHz to 2462 MHz

Table A-2. Complete datasheet for the TX91501B Powercast transmitter

The status LED indicates the following operating conditions:

LED	Status	Description
Off	Off	5VDC is not applied
Green	Active	Transmitting normally
Red[1]	Fault	Transmission stopped due to an obstruction in close proximity[2]

[1]During fault status, the LED will momentarily blink about once per second until the obstruction is removed.

[2]If operating on a flat surface for testing purposes, such as a table, it is recommended that the TX91501B is elevated off the surface with a non-conducting material, such as foam or a cardboard box, OR the TX91501B is positioned near the edge for unobstructed transmission.

SPECIFICATIONS

Item	Description
Modulation	Power: Direct Sequence Spread Spectrum (DSSS) Data: Amplitude Shift Keying (ASK)
Frequency	915 MHz Center Frequency
Power output (not user adjustable)	TX91501B-3W-XX: 3 watts EIRP TX91501B-1W-XX: 1 watt EIRP
Beam pattern	60° width, 60° height Vertical Polarization (with logo in upright position)
Approvals	FCC Part 15.247 Industry Canada
Firmware Version	Located on the TX91501B label
Data Communications	An 8-bit, factory-set, TX91501B identification (ID) number is broadcast with random intervals up to 10ms using Amplitude Shift Keying (ASK) modulation. The ID number can be found on the TX91501B label.
Operating Temperature	-20°C to 50°C
Installation Environment	Indoor use only
Mounting	6 bracket mounting holes (See mechanical drawing for dimensions) 2 keyholes (See mechanical drawing for dimensions)
Power	5VDC/1A through included Power Supply
Color	Black
Dimensions	6.75" x 6.25" x 1.63" (171.45 x 158.75 x 41.4 mm)
Weight	1 lb
Warranty	1 year

Rev A-1 – 2018/10 User's Manual - TX91501B Transmitter Page 2
www.powercastco.com

Table A-3. Complete datasheet for the P2110B Powerharvester receiver

P2110B

915 MHz RF Powerharvester® Receiver

SPECIFICATIONS

T_A = 25°C, RF_{IN} = 915MHz unless otherwise noted.

Parameter	Symbol	Condition	Min	Typ	Max	Unit
RF Characteristics[1]						
Input Power	RF_{IN}		-12		10	dBm
Frequency			902		928	MHz
DC Characteristics						
Output Voltage	V_{OUT}		2.0	3.3	5.5	V
Output Current	I_{OUT}				50	mA
V_{CAP} Maximum	V_{MAX}			1.25		V
V_{CAP} Minimum	V_{MIN}			1.02		V
Signal Strength	D_{OUT}	RF_{IN} = 0dBm		275		mV
Boost Efficiency		I_{OUT} = 20mA		85		%
Maximum INT Current				0.1		mA
Digital Characteristics						
RESET Input High				1		V
D_{SET} Input High			1.8			V
INT Output High			V_{MIN}		V_{MAX}	V
Timing Characteristics						
D_{SET} Delay				50		µs
RESET Delay				6.6		µs
RESET Pulse Width			20			ns

[1] See typical performance graphs for operation at other frequencies or power levels.

TYPICAL PERFORMANCE GRAPHS T_A = 25°C, V_{OUT} = 3.3V, V_{CAP} = 1.2V, unless otherwise noted

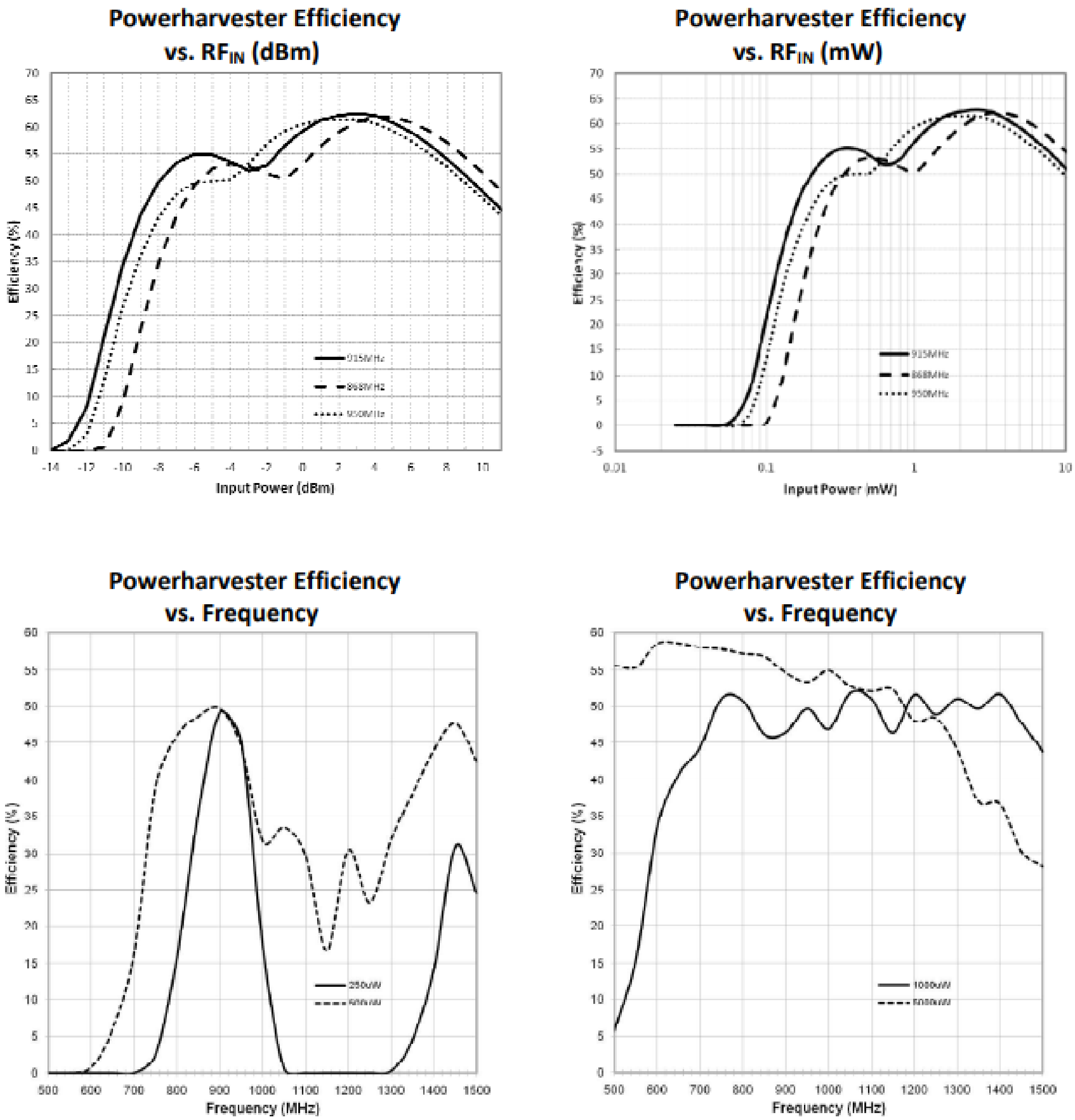


Figure A-1. Powerharvester efficiency vs frequency

Table A-4. Complete datasheet for the ADXL1002 sensor

SPECIFICATIONS

T_A = 25°C, V_{DD} = 5.0 V, acceleration = 0 *g*, unless otherwise noted.

Table 1.

Parameter[1]	Test Conditions/ Comments	ADXL1001 Min	ADXL1001 Typ	ADXL1001 Max	ADXL1002 Min	ADXL1002 Typ	ADXL1002 Max	Unit
SENSOR								
Measurement Range			±100			±50		*g*
Linearity	Percentage of full scale		±0.1			±0.1		%
Cross Axis Sensitivity[2]	ZX cross axis		±1.0			±1.0		%
	YX cross axis		±1.0			±1.0		%
SENSITIVITY (RATIOMETRIC TO V_{DD})								
Sensitivity	DC		20			40		mV/*g*
Sensitivity Change Due to Temperature[3]	T_A = −40°C to +125°C		±5			±5		%
ZERO *g* OFFSET (RATIOMETRIC TO V_{DD})								
0 *g* Output Voltage			$V_{DD}/2$			$V_{DD}/2$		V
0 *g* Output Range over Temperature[4]	−40°C to +125°C		5			5		*g*
NOISE								
Noise Density	100 Hz to 10 kHz		30			25		µg/√Hz
1/f Frequency Corner			0.1			0.1		Hz
FREQUENCY RESPONSE								
Sensor Resonant Frequency			21			21		kHz
5% Bandwidth[5]			4.7			4.7		kHz
3 dB Bandwidth[5]			11			11		kHz
SELF TEST								
Output Change (Ratiometric to V_{DD})	ST low to ST high	235	275		510	545		mV
Input Level								
High, V_{IH}		V_{DD} × 0.7			V_{DD} × 0.7			V
Low, V_{IL}				V_{DD} × 0.3			V_{DD} × 0.3	V
Input Current			25			25		µA
OUTPUT AMPLIFIER								
Short-Circuit Current			3			3		mA
Output Impedance			<0.1			<0.1		Ω
Maximum Resistive Load			20			20		MΩ
Maximum Capacitive Load[6]	No external resistor		100			100		pF
	With external resistor		22			22		nF
POWER SUPPLY (V_{DD})								
Operating Voltage Range		3.3	5.0	5.25	3.3	5.0	5.25	V
Quiescent Supply Current			1.0	1.15		1.0	1.15	mA
Standby Current			225	285		225	285	µA
Standby Recovery Time (Standby to Measure Mode)	Output settled to 1% of final value		<50			<50		µs
Turn On Time[7]			<550			<550		µs
OPERATING TEMPERATURE RANGE		−40		+125	−40		+125	°C

[1] All minimum and maximum specifications are guaranteed. Typical specifications may not be guaranteed.
[2] Cross axis sensitivity is defined as the coupling of excitation along a perpendicular axis onto the measured axis output.
[3] Includes package hysteresis from 25°C.
[4] Difference between maximum and minimum values in temperature range.
[5] Specified as frequency range that is within a deviation range relative to dc sensitivity, range is limited by an increase in response due to response gain at the sensor resonant frequency.
[6] For capacitive loads larger than 100 pF, an external series resistor must be connected (minimum 8 kΩ). The output capacitance must not exceed 22 nF.
[7] Measured time difference from the instant V_{DD} reaches half its value to the instant at which the output settles to 1% of its final value.

TYPICAL PERFORMANCE CHARACTERISTICS

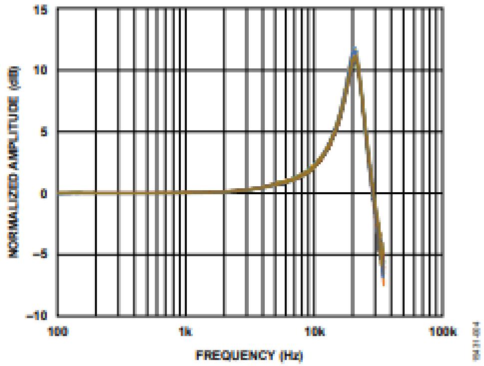


Figure 4. Frequency Response of ADXL1001/ADXL1002, High Frequency (>5 kHz) Vibration Response; a Laser Vibrometer Controller Referencing the ADXL1002 Package Used for Accuracy

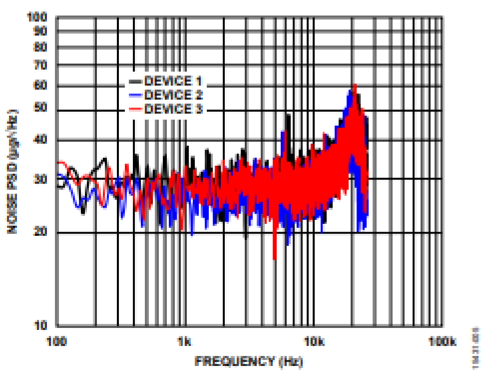


Figure 5. ADXL1001 Noise Power Spectral Density (PSD) vs. Frequency

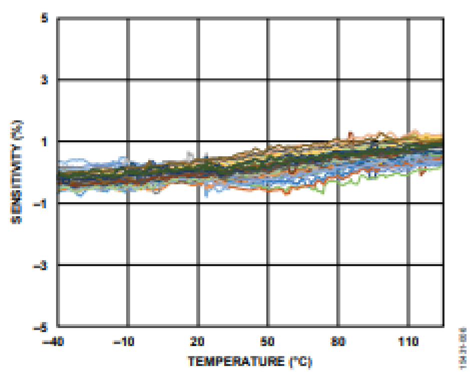


Figure 6. ADXL1001 Sensitivity vs. Temperature

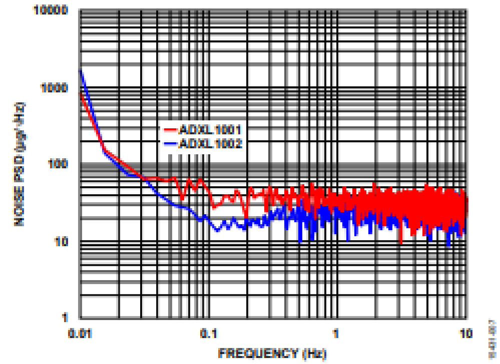


Figure 7. ADXL1001/ADXL1002 Noise Power Spectral Density (Noise PSD) Below 10 Hz

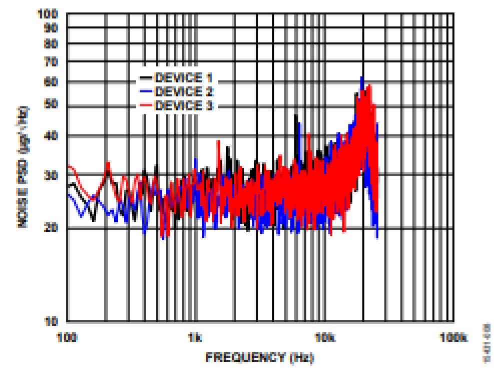


Figure 8. ADXL1002 Noise Power Spectral Density (PSD)

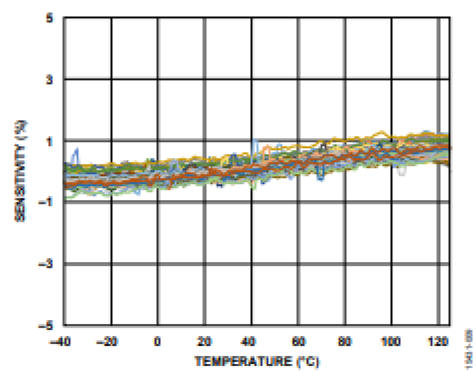


Figure 9. ADXL1002 Sensitivity vs. Temperature

Figure A-2. Typical performance characteristics of the ADXL1002

Table A-5. Complete datasheet specifications for the ADP5300 sensor

SPECIFICATIONS

V_{IN} = 3.6 V, V_{OUT} = 2.5 V, T_J = −40°C to +125°C for minimum and maximum specifications, and T_A = 25°C for typical specifications, unless otherwise noted.

Table 1.

Parameter	Symbol	Min	Typ	Max	Unit	Test Conditions/Comments
INPUT SUPPLY VOLTAGE RANGE	V_{IN}	2.15		6.50	V	
SHUTDOWN CURRENT	$I_{SHUTDOWN}$		18	40	nA	V_{EN} = 0 V, −40°C ≤ T_J ≤ +85°C
			18	130	nA	V_{EN} = 0 V, −40°C ≤ T_J ≤ +125°C
QUIESCENT CURRENT						
Operating Quiescent Current in Hysteresis Mode	I_{Q_HYS}		180	260	nA	−40°C ≤ T_J ≤ +85°C
			180	350	nA	−40°C ≤ T_J ≤ +125°C
			570	1400	nA	100% duty cycle operation, V_{IN} = 3.0 V, V_{OUT} set to 3.3 V
Operating Quiescent Current in Hysteresis Mode	$I_{Q\text{-}HYS2}$		2.3	3.2	µA	V_{IN} = 3.6 V, V_{STOP} = 3.6 V
Operating Quiescent Current in PWM Mode	I_{Q_PWM}		425	630	µA	
UNDERVOLTAGE LOCKOUT	UVLO					
UVLO Threshold						
Rising	V_{UVLO_RISING}		2.06	2.14	V	
Falling	$V_{UVLO\_FALLING}$	1.90	2.00		V	
OSCILLATOR CIRCUIT						
Switching Frequency in PWM Mode	f_{SW}	1.7	2.0	2.3	MHz	
Feedback (FB) Threshold of Frequency Fold	V_{OSC_FOLD}		0.3		V	
SYNCHRONIZATION THRESHOLD						
SYNC Clock Range	$SYNC_{CLOCK}$	1.5		2.5	MHz	
SYNC High Level Threshold	$SYNC_{HIGH}$	1.2			V	
SYNC Low Level Threshold	$SYNC_{LOW}$			0.4	V	
SYNC Duty Cycle Range	$SYNC_{DUTY}$	100		$1/f_{SW}$ − 150	ns	
SYNC/MODE Leakage Current	$I_{SYNC\_LEAKAGE}$		50	150	nA	$V_{SYNC/MODE}$ = 3.6 V
MODE TRANSITION						
Transition Delay from Hysteresis Mode to PWM Mode	$t_{HYS\_TO\_PWM}$		20		Clock cycles	SYNC/MODE goes logic high from logic low
EN PIN						
Input Voltage Threshold						
High	V_{IH}	1.2			V	
Low	V_{IL}			0.4	V	
Input Leakage Current	$I_{EN\_LEAKAGE}$			25	nA	
STOP Switching						
PWM Switching Stop Delay	$t_{STOP\text{-}RISE\text{-}DELAY}$		10		ns	STOP goes logic high from low
PWM Switching Resume Delay	$t_{STOP\text{-}FALL\text{-}DELAY}$		20		ns	STOP goes logic low from high
FB PIN						
Output Options by VID Resistor	V_{OUT_OPT}	0.8		5.0	V	0.8 V to 5.0 V in various factory options
PWM Mode						
Fixed VID Code Voltage Accuracy	$V_{FB\_PWM\_FIX}$	−0.6		+0.6	%	T_J = 25°C, output voltage setting via factory fuse
		−1.2		+1.2	%	−40°C ≤ T_J ≤ +125°C
Adjustable VID Code Voltage Accuracy	$V_{FB\_PWM\_ADJ}$	−1.5		+1.5	%	Output voltage setting via VID resistor

Parameter	Symbol	Min	Typ	Max	Unit	Test Conditions/Comments
Hysteresis Mode						
Fixed VID Code Threshold Accuracy from Active Mode to Standby Mode	$V_{FB\_HYS\_FIX}$	−0.75		+0.75	%	T_J = 25°C
		−2.5		+2.5	%	−40°C ≤ T_J ≤ +125°C
Adjustable VID Code Threshold Accuracy from Active Mode to Standby Mode	$V_{FB\_HYS\_ADJ}$	−3		+3	%	−40°C ≤ T_J ≤ +125°C
Hysteresis of Threshold Accuracy from Active Mode to Standby Mode	$V_{FB\_HYS\ (HYS)}$		1		%	
Feedback Bias Current	I_{FB}		66	95	nA	Output Option 0, V_{OUT} = 2.5 V
			25	45	nA	Output Option 1, V_{OUT} = 1.3 V
SW PIN						
High-Side Power FET On Resistance	$R_{DS\ (ON)\ H}$		386	520	mΩ	Pin to pin measurement
Low-Side Power FET On Resistance	$R_{DS\ (ON)\ L}$		299	470	mΩ	Pin to pin measurement
Current Limit in PWM Mode	I_{LIM_PWM}	800	1000	1200	mA	SYNC/MODE = high
Peak Current in Hysteresis Mode	I_{LIM_HYS}		265		mA	SYNC/MODE = low
Minimum On Time	t_{MIN_ON}		40	70	ns	
VOUTOK PIN						
Monitor Threshold	$V_{OUTOK\ (RISE)}$	87	90	93	%	
Monitor Hysteresis	$V_{OUTOK\ (HYS)}$		3		%	
Monitor Rising Delay	t_{VOUTOK_RISE}		40		µs	
Monitor Falling Delay	t_{VOUTOK_FALL}		10		µs	
Leakage Current	$I_{VOUTOK\_LEAKAGE}$		0.1	1	µA	
Output Low Voltage	V_{OUTOK_LOW}		50	80	mV	I_{VOUTOK} = 100 µA
SOFT START						
Default Soft Start Time	t_{SS}		350		µs	Factory trim, 1 bit (350 µs and 2800 µs)
Start-Up Delay	t_{START_DELAY}		2		ms	Delay from the EN pin being pulled high
C_{OUT} DISCHARGE SWITCH ON RESISTANCE	R_{DIS}		290		Ω	
THERMAL SHUTDOWN						
Threshold	T_{SHDN}		142		°C	
Hysteresis	T_{HYS}		127		°C	

Appendix B – Matlab code

This appendix contains the Matlab code for the simulator used in Chapter 5.

main.m

```
clc;
clear;
for jj = 1:1
for ii = 1:3

%% User Input
% Plant size
plant_width = 15; %% input %% width of plant (meter)
plant_length = 15; %% input %% length of plant (meter)
plant_height = 9; %% input %% heigth of plant (meter)
%Plant shape (rectangle)
r1 = [2, (plant_length-8)*(2/3); plant_width-2-(plant_width-6)/2,plant_width-
2];
r2 = [(plant_length-8)*(2/3)+4,plant_length-2; plant_width-2-(plant_width-
6)/2,plant_width-2];
r3 = [2,plant_length-2; 2,(plant_width-2)/2];
%Duty cycle
duty_cycle = 10; %s/h
%Nodes density
high_density = 0.2*plant_width*plant_length;
low_density = 0.044*plant_width*plant_length;

% if ii == 1 || ii == 3 || ii == 5
%    node_density = low_density; %x/m^2
% else
%     node_density = high_density; %x/m^2
% end

number_nodes =
46;%round(node_density);%round(plant_width*plant_lenght*sensor_density); %%
input %% Number of devices required

%Nodes configuration

if ii == 1
    placement = 'clumped';
elseif ii == 2
    placement = 'random';
else
    placement = 'uniform spacing';
end

[xl,yl,zl] = getNodeLocation(r1,r2,r3,plant_height,number_nodes,placement);
%random, clumped, uniform spacing
number_nodes_2 = size(xl,2); %actual number of nodes
nodes_location = cell(1,number_nodes_2); %initialization of node location
```

```
%000000000000000000000000000000000000000000
% figure(9);
% scatter3(xl,yl,zl,'MarkerEdgeColor','black','MarkerFaceColor','black');
% %axis equal

%Attenuation
[plant_attenuation_min, plant_attenuation_max] =
getAttenuation('concrete');%choice between LOS, concrete, brick, Faraday
% Nodes generation
for i3 = 1:number_nodes_2 % randomnly generated coordinates of equipment to
monitor
    attenuation = randi([plant_attenuation_min plant_attenuation_max],1,1); %
randomnly generated attenuatiuon
    %manual%
    %attenuation =
[4,5,7,3,7,6,4,3,5,5,3,5,4,4,5,4,4,6,4,7,7,6,4,5,3,7,7,7,4,5,3,5,4,3,3,5,3,5,
5,6,6,6,3,3,4,5];
    nodes_location{i3} = [xl(i3), yl(i3), zl(i3), attenuation];
end
%Cost of implementation
min_cost_imp = 200; %$
desire_lifetime = 10; %year

M = 1; %%Minimum amount of device per transmitte
%% Variable input
storage_capacity = 10;%mA %0 if there is no storage
%% Designer Input
%Friis equation and device consumption
[Pt,Pr1,Pr2,d1,d2,Gt1, Gt2, Gr, Freq, Gt_lin,E_active,E_standby,E_device] =
getFriisEquation(duty_cycle);
%Maximum distance between transmitter and receiver omnidirectional
d_tx_rx_1 = 0;
for i2 = 1:length(Pr1)
    if Pr1(i2) < E_device && d1(i2) ~= 0.5
        d_tx_rx_1 = d1(i2);
        break;
    end
end
%Maximum distance between transmitter and receiver directional
d_tx_rx_2 = 0;
for j2 = 1:length(Pr2)
    if Pr2(j2) < E_device
        d_tx_rx_2 = d2(j2);
        break;
    end
end
% Graph of device energy consumption for 10 years
t = 0:1:10; %year
active_ten_year = E_active*8760*t;
standby_ten_year = E_standby*8760*t;
lifetime1 = Pr1/E_device;
lifetime2 = Pr2/E_device;

%% Variable input - Create the spheres for omnidirectional antennas
```

```
[coordinates_1,counter_sphere_1] = createOmniAntenna(d_tx_rx_1, plant_width,
plant_length, plant_height);

%% Variable input - Create the ellipsoid for directional antennas
[coordinates_2,counter_sphere_2,x_ell,y_ell] = createDirectAntenna(Gt_lin,
plant_width, plant_length, plant_height);

%% Find if the device is within the sphere/ellispoid
%Sphere
sphere_device_coord_LOS_1 =
getDeviceCoveredByOmni(coordinates_1,counter_sphere_1,number_nodes_2,xl,yl,zl
,d_tx_rx_1,nodes_location);
% Ellipsoid
sphere_device_coord_LOS_2 =
getDeviceCoveredByDirect(coordinates_2,counter_sphere_2,number_nodes_2,xl,yl,
zl,x_ell,y_ell,Gt_lin,nodes_location);

%% Delete devices based on attenuation
%Sphere
sphere_device_coord_NLOS_1_1 =
deleteNLOSCoverageOmni(sphere_device_coord_LOS_1,Pt,Gt1,Gr,Freq,E_device);
%Ellipsoid
sphere_device_coord_NLOS_2_2 =
deleteNLOSCoverageDirect(sphere_device_coord_LOS_2,Pt,Gt2,Gr,Freq,E_device);

%% Arrange the spheres by amount of device covered
%Omnidirectional
[sphere_device_coord_arranged_1,sphere_device_coord_arranged_1_2] =
arrangeOmniList(sphere_device_coord_NLOS_1_1,number_nodes_2);
%Directional
[sphere_device_coord_arranged_2,sphere_device_coord_arranged_2_2] =
arrangeDirectList(sphere_device_coord_NLOS_2_2,number_nodes_2);

%% Find the most efficient zones spheres%%%
% Omnidirectional
[priority_sphere_A_3,A0_stationary_device_transmitter_omni_number,total_node_
E] = getOmniCoverage(sphere_device_coord_arranged_1_2,nodes_location,M);
%Directional
[priority_sphere_B_3,A1_stationary_device_transmitter_direct_number,total_nod
e_E2] = getDirectCoverage(sphere_device_coord_arranged_2_2,nodes_location,M);

%% Mobile Zones
E_required_for_xtime = storage_capacity * 3.3; %Energy required to fully
charge a device
%battery_max_charge_cycle = 10000;
full_charge_time = E_required_for_xtime/E_device; % time available after
fully charge a device
vitesse = 1;
%Omnidirectional
if size(sphere_device_coord_NLOS_1_1,1) ~= 0
    zone_mobile_A = getZoneMobileOmni(nodes_location,
sphere_device_coord_NLOS_1_1);
    [A2_mobile_device_transmitter_omni_number,node_life_time_A,zone_A4] =
getMobileOmni(zone_mobile_A,E_required_for_xtime,full_charge_time,vitesse);
else
    A2_mobile_device_transmitter_omni_number = [0,0];
```

```
end
%directional
if size(sphere_device_coord_NLOS_2_2,1) ~= 0
    zone_mobile_B = getZoneMobileDirect(nodes_location,
sphere_device_coord_NLOS_2_2);
    [A3_mobile_device_transmitter_direct_number,node_life_time_B,zone_B4] =
getMobileDirect(zone_mobile_B,E_required_for_xtime,full_charge_time,vitesse);
else
    A3_mobile_device_transmitter_omni_number = [0,0]
end

%% Result
A0_Power_tx_cost =
A0_stationary_device_transmitter_omni_number(1)*300+(A0_stationary_device_tra
nsmitter_omni_number(2)*200);
A0_Implementation_cost = A0_stationary_device_transmitter_omni_number(1)*200;
A0_Energy_consumption(jj,ii) =
A0_stationary_device_transmitter_omni_number(1)*5*0.001*3600000; %+energy
drones, joules
A0_Energy_cost =
A0_stationary_device_transmitter_omni_number(1)*5*0.001*0.179*24;
A0_Labour_cost = 0;
A0_Capital_cost =
A0_Power_tx_cost+(A0_Energy_cost*365)+A0_Implementation_cost;
A0_Cost(jj,ii) = A0_Labour_cost+A0_Capital_cost;
A0_Energy_efficiency(jj,ii) =
((total_node_E*0.000001*3600000)/A0_Energy_consumption(jj,ii))*200; %%%
A0_system_lifetime = 10; %%%
A0_system_efficiency(jj,ii) =
A0_Cost(jj,ii)/A0_stationary_device_transmitter_omni_number(2)/A0_system_life
time;
A0_Lost_cause(jj,ii) = (number_nodes_2-
A0_stationary_device_transmitter_omni_number(2))/number_nodes_2*100;
A0_Transmitter(jj,ii) = A0_stationary_device_transmitter_omni_number(1);
A000_Result(ii,:) = [A0_Cost(jj,ii), A0_Energy_consumption(jj,ii),
A0_Energy_efficiency(jj,ii)...
                  A0_Transmitter(jj,ii), A0_Lost_cause(jj,ii),
A0_system_efficiency(jj,ii)];


A1_Power_tx_cost =
A1_stationary_device_transmitter_direct_number(1)*300+(A1_stationary_device_t
ransmitter_direct_number(2)*200);
A1_Implementation_cost =
A1_stationary_device_transmitter_direct_number(1)*200;
A1_Energy_consumption(jj,ii) =
A1_stationary_device_transmitter_direct_number(1)*5*0.001*3600000; %+energy
drones, joules
A1_Energy_cost =
A1_stationary_device_transmitter_direct_number(1)*5*0.001*0.179*24;
A1_Labour_cost = 0;
A1_Capital_cost =
A1_Power_tx_cost+(A1_Energy_cost*365)+A1_Implementation_cost;
A1_Cost(jj,ii) = A1_Labour_cost+A1_Capital_cost;
A1_Energy_efficiency(jj,ii) =
((total_node_E2*0.000001*3600000)/A1_Energy_consumption(jj,ii))*200; %%%
```

```
A1_system_lifetime = 10; %%%
A1_system_efficiency(jj,ii) =
A1_Cost(jj,ii)/A1_stationary_device_transmitter_direct_number(2)/A1_system_li
fetime;
A1_Lost_cause(jj,ii) = (number_nodes_2-
A1_stationary_device_transmitter_direct_number(2))/number_nodes_2*100;
A1_Transmitter(jj,ii) = A1_stationary_device_transmitter_direct_number(1);
A001_Result(ii,:) = [A1_Cost(jj,ii), A1_Energy_consumption(jj,ii),
A1_Energy_efficiency(jj,ii)...
                  A1_Transmitter(jj,ii), A1_Lost_cause(jj,ii),
A1_system_efficiency(jj,ii)];

A2_Power_tx_cost =
A2_mobile_device_transmitter_omni_number(1)*300+(A2_mobile_device_transmitter
_omni_number(2)*200)...
    +(number_nodes_2-A2_mobile_device_transmitter_omni_number(2))*300 +
(number_nodes_2-A2_mobile_device_transmitter_omni_number(2))*200;
A2_Implementation_cost = A2_mobile_device_transmitter_omni_number(1)*200 +
...
    (number_nodes_2-A2_mobile_device_transmitter_omni_number(2))*200; % fixed
node for mobile
A2_Energy_consumption(jj,ii) = ((A2_mobile_device_transmitter_omni_number(1)
+ number_nodes_2-A2_mobile_device_transmitter_omni_number(2))...
    *5*0.001*3600000)+3600; %+energy drones, joules
A2_Energy_cost = (A2_mobile_device_transmitter_omni_number(1)+number_nodes_2-
A2_mobile_device_transmitter_omni_number(2))...
    *5*0.001*0.179*24;
A2_Labour_time = 46;
A2_Labour_cost = 40*A2_Labour_time*2;%40$/hr
A2_Capital_cost =
A2_Power_tx_cost+(A2_Energy_cost*365)+A2_Implementation_cost;
A2_Cost(jj,ii) = A2_Labour_cost+A2_Capital_cost;
A2_Energy_efficiency(jj,ii) =
((total_node_E*0.000001*3600000)/A2_Energy_consumption(jj,ii))*200; %%%
A2_system_lifetime = 10; %%%
A2_system_efficiency(jj,ii) =
A2_Cost(jj,ii)/A2_mobile_device_transmitter_omni_number(2)/A2_system_lifetime
;
A2_Lost_cause(jj,ii) = (number_nodes_2-
A2_mobile_device_transmitter_omni_number(2))/number_nodes_2*100;
A2_Transmitter(jj,ii) = A2_mobile_device_transmitter_omni_number(1);
A002_Result(ii,:) = [A2_Cost(jj,ii), A2_Energy_consumption(jj,ii),
A2_Energy_efficiency(jj,ii)...
                  A2_Transmitter(jj,ii), A2_Lost_cause(jj,ii),
A2_system_efficiency(jj,ii)];

A3_Power_tx_cost =
A3_mobile_device_transmitter_direct_number(1)*300+(A3_mobile_device_transmitt
er_direct_number(2)*200)...
    +(number_nodes_2-A3_mobile_device_transmitter_direct_number(2))*300 +
(number_nodes_2-A3_mobile_device_transmitter_direct_number(2))*200;
A3_Implementation_cost =A3_mobile_device_transmitter_direct_number(1)*200 ...
    +(number_nodes_2-A3_mobile_device_transmitter_direct_number(2))*200;
A3_Energy_consumption(jj,ii) =
((A3_mobile_device_transmitter_direct_number(1) + (number_nodes_2-
A3_mobile_device_transmitter_direct_number(2)))...
```

```
    *5*0.001*3600000)+3600; %+energy drones, joules
A3_Energy_cost = (A3_mobile_device_transmitter_direct_number(1) +
number_nodes_2-A3_mobile_device_transmitter_direct_number(2))...
    *5*0.001*0.179*24;
A3_Labour_time = 46;
A3_Labour_cost = 40*A3_Labour_time*2;%40$/hr
A3_Capital_cost =
A3_Power_tx_cost+(A3_Energy_cost*365)+A3_Implementation_cost;
A3_Cost(jj,ii) = A3_Labour_cost+A3_Capital_cost;
A3_Energy_efficiency(jj,ii) =
((total_node_E2*0.000001*3600000)/A3_Energy_consumption(jj,ii))*200; %%%
A3_system_lifetime = 10; %%%
A3_system_efficiency(jj,ii) =
A3_Cost(jj,ii)/A3_mobile_device_transmitter_direct_number(2)/A3_system_lifeti
me;
A3_Lost_cause(jj,ii) = (number_nodes_2-
A3_mobile_device_transmitter_direct_number(2))/number_nodes_2*100;
A3_Transmitter(jj,ii) = A3_mobile_device_transmitter_direct_number(1);
A003_Result(ii,:) = [A3_Cost(jj,ii), A3_Energy_consumption(jj,ii),
A3_Energy_efficiency(jj,ii)...
                  A3_Transmitter(jj,ii), A3_Lost_cause(jj,ii),
A3_system_efficiency(jj,ii)];


end
end
```

arrangeDirectList.m

```
function [sphere_device_coord_arranged_2,sphere_device_coord_arranged_2_2] =
arrangeDirectList(sphere_device_coord_NLOS_2_2,number_nodes_2)
    sphere_coord_2 = sphere_device_coord_NLOS_2_2(:,1); %Sphere coordinates
that contains devices
    device_coord_in_sphere_2 = sphere_device_coord_NLOS_2_2(:,2); %Devices
that are contained in each sphere
    sphere_device_coord_arranged_2 =
cell(size(sphere_coord_2,1),number_nodes_2);
    count_numb = 1;

    for j11 = 1:size(sphere_coord_2,1) %Find spheres that cover more than 1
device and rearange them
        if count_numb <= 1 %If there are no more duplicate
            count_numb = 0;
            sphere_device_coord_arranged_2{j11,1} = sphere_coord_2{j11};
%place the sphere coordinate in the first column
            for j12 = j11:size(sphere_coord_2,1)
                if (sphere_coord_2{j11}(1) == sphere_coord_2{j12}(1) &&
sphere_coord_2{j11}(2) == sphere_coord_2{j12}(2) && sphere_coord_2{j11}(3) ==
sphere_coord_2{j12}(3)) %if the sphere coordinates are similar
                    sphere_device_coord_arranged_2{j11,(j12-j11+2)} =
device_coord_in_sphere_2{j12}; %Place the devices coordinate in the same row
(only 1 instance of that sphere)
```

```
                    count_numb = count_numb+1; %increase everytime the sphere
coordinates are similar
                end
            end
        else
            count_numb = count_numb - 1; %decrement to leave and not put
duplicate sphere in cells
        end
    end
    sphere_device_coord_arranged_2 =
sphere_device_coord_arranged_2(~cellfun('isempty',sphere_device_coord_arrange
d_2(:,1)), :); % remove any empty cells
    sphere_device_coord_arranged_2_2 =
cell(size(sphere_device_coord_arranged_2,1),1);
    for j12 = 1:size(sphere_device_coord_arranged_2,1) %rearange cells to
only contains actual values, no empty lists.
        temp_2 = sphere_device_coord_arranged_2(j12,:);
        sphere_device_coord_arranged_2_2{j12,:} =
temp_2(~cellfun('isempty',sphere_device_coord_arranged_2(j12,:)));
    end
end
```

arrangeOmniList.m

```
function [sphere_device_coord_arranged_1,sphere_device_coord_arranged_1_2] =
arrangeOmniList(sphere_device_coord_NLOS_1_1,number_nodes_2)
    sphere_coord_1 = sphere_device_coord_NLOS_1_1(:,1); %Sphere coordinates
that contains devices
    device_coord_in_sphere_1 = sphere_device_coord_NLOS_1_1(:,2); %Devices
that are contained in each sphere

    count_numb = 1;
    sphere_device_coord_arranged_1 =
cell(size(sphere_coord_1,1),number_nodes_2);

    for i10 = 1:size(sphere_coord_1,1) %Find spheres that cover more than 1
device and rearange them
        if count_numb <= 1 %If there are no more duplicate
            count_numb = 0;
            sphere_device_coord_arranged_1{i10,1} = sphere_coord_1{i10};
%place the sphere coordinate in the first column
            for i11 = i10:size(sphere_coord_1,1)
                if (sphere_coord_1{i10}(1) == sphere_coord_1{i11}(1) &&
sphere_coord_1{i10}(2) == sphere_coord_1{i11}(2) && sphere_coord_1{i10}(3) ==
sphere_coord_1{i11}(3)) %if the sphere coordinates are similar
                    sphere_device_coord_arranged_1{i10,(i11-i10+2)} =
device_coord_in_sphere_1{i11}; %Place the devices coordinate in the same row
(only 1 instance of that sphere)
                    count_numb = count_numb+1; %increase everytime the sphere
coordinates are similar
                end
            end
        else
            count_numb = count_numb - 1; %decrement to leave and not put
duplicate sphere in cells
```

```
        end
    end
    sphere_device_coord_arranged_1 =
sphere_device_coord_arranged_1(~cellfun('isempty',sphere_device_coord_arrange
d_1(:,1)), :); % remove any empty cells
    sphere_device_coord_arranged_1_2 =
cell(size(sphere_device_coord_arranged_1,1),1);
    for i11 = 1:size(sphere_device_coord_arranged_1,1) %rearange cells to
only contains actual values, no empty lists.
        temp_1 = sphere_device_coord_arranged_1(i11,:);
        sphere_device_coord_arranged_1_2{i11,:} =
temp_1(~cellfun('isempty',sphere_device_coord_arranged_1(i11,:)));
    end
end
```

createDirectAntenna.m

```
function [sphere_device_coord_arranged_1,sphere_device_coord_arranged_1_2] =
arrangeOmniList(sphere_device_coord_NLOS_1_1,number_nodes_2)
    sphere_coord_1 = sphere_device_coord_NLOS_1_1(:,1); %Sphere coordinates
that contains devices
    device_coord_in_sphere_1 = sphere_device_coord_NLOS_1_1(:,2); %Devices
that are contained in each sphere

    count_numb = 1;
    sphere_device_coord_arranged_1 =
cell(size(sphere_coord_1,1),number_nodes_2);

    for i10 = 1:size(sphere_coord_1,1) %Find spheres that cover more than 1
device and rearange them
        if count_numb <= 1 %If there are no more duplicate
            count_numb = 0;
            sphere_device_coord_arranged_1{i10,1} = sphere_coord_1{i10};
%place the sphere coordinate in the first column
            for i11 = i10:size(sphere_coord_1,1)
                if (sphere_coord_1{i10}(1) == sphere_coord_1{i11}(1) &&
sphere_coord_1{i10}(2) == sphere_coord_1{i11}(2) && sphere_coord_1{i10}(3) ==
sphere_coord_1{i11}(3)) %if the sphere coordinates are similar
                    sphere_device_coord_arranged_1{i10,(i11-i10+2)} =
device_coord_in_sphere_1{i11}; %Place the devices coordinate in the same row
(only 1 instance of that sphere)
                    count_numb = count_numb+1; %increase everytime the sphere
coordinates are similar
                end
            end
        else
            count_numb = count_numb - 1; %decrement to leave and not put
duplicate sphere in cells
        end
    end
    sphere_device_coord_arranged_1 =
sphere_device_coord_arranged_1(~cellfun('isempty',sphere_device_coord_arrange
d_1(:,1)), :); % remove any empty cells
    sphere_device_coord_arranged_1_2 =
cell(size(sphere_device_coord_arranged_1,1),1);
```

```
    for i11 = 1:size(sphere_device_coord_arranged_1,1) %rearange cells to
only contains actual values, no empty lists.
        temp_1 = sphere_device_coord_arranged_1(i11,:);
        sphere_device_coord_arranged_1_2{i11,:} =
temp_1(~cellfun('isempty',sphere_device_coord_arranged_1(i11,:)));
    end
end
```

createOmniAntenna.m

```
function [coordinates_1,counter_sphere_1] = createOmniAntenna(d_tx_rx_1,
plant_width, plant_lenght, plant_height)
    inc_dist = 0.3; % Generate sphere every "inc_dist" meter
    r = d_tx_rx_1; % omnidirectional circle radius which is the max distance
between transmitter and receiver
    counter_sphere_1 = 0; %Number of spheres
    coordinates_1 = cell(round((plant_width-2*r)/inc_dist*(plant_lenght-
2*r)/inc_dist*(plant_height-2*r)/inc_dist),1);
     for i4 = r/2:inc_dist:plant_width-r/2
         for i5 = r/2:inc_dist:plant_lenght-r/2
             for i6 = r/2:inc_dist:plant_height+2-r/2
                 counter_sphere_1 = counter_sphere_1 + 1;
                 coordinates_1{counter_sphere_1,1} = [i4,i5,i6]; %location of
all the omni-transmitter
             end
         end
     end
end
```

deleteNLOSCoverageDirect.m

```
function sphere_device_coord_NLOS_2_2 =
deleteNLOSCoverageDirect(sphere_device_coord_LOS_2,Pt,Gt2,Gr,Freq,E_device)
    sphere_device_coord_NLOS_2 = sphere_device_coord_LOS_2;
    for j10 = 1:size(sphere_device_coord_NLOS_2,1) % Check every value in the
list
       d_tx_rx_2_2 = (sphere_device_coord_NLOS_2{j10,1}(1)-
sphere_device_coord_NLOS_2{j10,2}(1))^2+(sphere_device_coord_NLOS_2{j10,1}(2)
-
sphere_device_coord_NLOS_2{j10,2}(2))^2+(sphere_device_coord_NLOS_2{j10,1}(3)
-sphere_device_coord_NLOS_2{j10,2}(3))^2;% Distance between transmitter and
receiver
       Pout =
(Pt*10^(Gt2/10)*10^(Gr/10)*((3*10^8)/(Freq*10^9))^2)/(4*pi()*d_tx_rx_2_2)^2;%
Power received by sensor in LOS
       Pin = Pout/(10^(sphere_device_coord_NLOS_2{j10,2}(4)/10));
       sphere_device_coord_NLOS_2{j10,2}(5) = Pin;
       if Pin < E_device || d_tx_rx_2_2 < (3*10^8)/(Freq*10^9)
           sphere_device_coord_NLOS_2{j10,2} = {};
       end
    end
```

```
    sphere_device_coord_NLOS_2_2 =
sphere_device_coord_NLOS_2(~cellfun('isempty',sphere_device_coord_NLOS_2(:,2)
), :); %Device covered by ellipsoid in NLOS
end
```

deleteNLOSCoverageOmni.m

```
function sphere_device_coord_NLOS_1_1 =
deleteNLOSCoverageOmni(sphere_device_coord_LOS_1,Pt,Gt1,Gr,Freq,E_device)
    sphere_device_coord_NLOS_1 = sphere_device_coord_LOS_1;
    for i9 = 1:size(sphere_device_coord_NLOS_1,1) % Check every value in the
list
       d_tx_rx_1_1 = (sphere_device_coord_NLOS_1{i9,1}(1)-
sphere_device_coord_NLOS_1{i9,2}(1))^2+(sphere_device_coord_NLOS_1{i9,1}(2)-
sphere_device_coord_NLOS_1{i9,2}(2))^2+(sphere_device_coord_NLOS_1{i9,1}(3)-
sphere_device_coord_NLOS_1{i9,2}(3))^2;% Distance between transmitter and
receiver
       Pout =
(Pt*10^(Gt1/10)*10^(Gr/10)*((3*10^8)/(Freq*10^9))^2)/(4*pi()*d_tx_rx_1_1)^2;%
Power received by sensor in LOS
       Pin = Pout/(10^((sphere_device_coord_NLOS_1{i9,2}(4))/10)); %Power
received after attenuation
       sphere_device_coord_NLOS_1{i9,2}(5) = Pin;
       if Pin < E_device || d_tx_rx_1_1 < ((3*10^8)/(Freq*10^9))
           sphere_device_coord_NLOS_1{i9,2} = {};
       end
    end
    sphere_device_coord_NLOS_1_1 =
sphere_device_coord_NLOS_1(~cellfun('isempty',sphere_device_coord_NLOS_1(:,2)
), :); %Device covered by sphere in NLOS
end
```

getAttenuation.m

```
function [plant_attenuation_min, plant_attenuation_max] =
getAttenuation(material_type)
    if strcmp(material_type,'LOS')
        plant_attenuation_min = 0;
        plant_attenuation_max = 0;
    elseif strcmp(material_type,'concrete')
        plant_attenuation_min = 12;
        plant_attenuation_max = 35;
    elseif strcmp(material_type,'brick')
        plant_attenuation_min = 3;
        plant_attenuation_max = 7;
    elseif strcmp(material_type,'faraday')
        plant_attenuation_min = 50;
        plant_attenuation_max = 100;
    end
end
```

getDeviceCoveredByDirect.m

```
function sphere_device_coord_LOS_2 =
getDeviceCoveredByDirect(coordinates_2,counter_sphere_2,number_nodes_2,xl,yl,
zl,x_ell,y_ell,Gt_lin,nodes_location)
    sphere_device_coord_LOS_2 = cell(size(coordinates_2,1),2);
    counter_value = 0;
    for j8 = 1:counter_sphere_2+1
        for j9 = 1:number_nodes_2
            if ((xl(j9)-coordinates_2{j8}(1))/x_ell)^2+((yl(j9)-
coordinates_2{j8}(2))/y_ell)^2+((zl(j9)-coordinates_2{j8}(3))/Gt_lin)^2 < 1
%distance between device and sphere is less than radius^2. Distance between 2
points
              counter_value = counter_value + 1;
              sphere_device_coord_LOS_2{counter_value,1} = coordinates_2{j8};
%coordinates of spheres
              sphere_device_coord_LOS_2{counter_value,2} =
nodes_location{j9}; %coordinates of devices
            end
        end
    end
    sphere_device_coord_LOS_2 =
sphere_device_coord_LOS_2(~cellfun('isempty',sphere_device_coord_LOS_2(:,1)),
:); %Tx that coveres nodes in LOS
end
```

getDeviceCoveredByOmni.m

```
function sphere_device_coord_LOS_1 =
getDeviceCoveredByOmni(coordinates_1,counter_sphere_1,number_nodes_2,xl,yl,zl
,d_tx_rx_1,nodes_location)
    counter_value = 0;
    sphere_device_coord_LOS_1 = cell(size(coordinates_1,1),2);
    for i7 = 1:counter_sphere_1
        for i8 = 1:number_nodes_2
            if (xl(i8)-coordinates_1{i7}(1))^2+(yl(i8)-
coordinates_1{i7}(2))^2+(zl(i8)-coordinates_1{i7}(3))^2 <= d_tx_rx_1^2
%distance between device and sphere is less than radius^2. Distance between 2
points
                counter_value = counter_value + 1;
                sphere_device_coord_LOS_1{counter_value,1} =
coordinates_1{i7}; %coordinates of spheres
                sphere_device_coord_LOS_1{counter_value,2} =
nodes_location{i8}; %coordinates of devices
            end
        end
    end
    sphere_device_coord_LOS_1 =
sphere_device_coord_LOS_1(~cellfun('isempty',sphere_device_coord_LOS_1(:,1)),
:); %Tx that coveres nodes in LOS
end
```

getDirectCoverage.m

```
function
[priority_sphere_B_3,A1_stationary_device_transmitter_direct_number,total_nod
e_E2] = getDirectCoverage(sphere_device_coord_arranged_2_2,nodes_location,M)
    % Find the number of devices that each spheres contains
    [ss2,dd2] = cellfun(@size,sphere_device_coord_arranged_2_2);
    priority_n_2 = max(dd2);

    count_prio = 0;
    for j13 = priority_n_2-count_prio:-1:M+1 %remove all sphere that contains
less than the minimum amount of devices required
        for j14 = 1:size(sphere_device_coord_arranged_2_2,1)
            if size(sphere_device_coord_arranged_2_2{j14,1},2) == j13
                sphere_device_coord_arranged_2_3{j14,j13} =
sphere_device_coord_arranged_2_2{j14,1}; %Place each sphere in different
column based on their priority.
            end
        end
        count_prio = count_prio + 1;
    end

    %Remove devices that aren't find in the sphere
    priority_sphere_B = cell(size(nodes_location,2),2);
    %sphere_device_coord_arranged_2_4 = sphere_device_coord_arranged_2_3;
    for j15 = M+1:priority_n_2 %If a device is found in 2+ priority spheres,
only the highest priority sphere will be kept. The others will be removed.
        for j16 = 1:size(nodes_location,2) %Find the highest priority sphere
in which a device is located
            for j17 = 1:size(sphere_device_coord_arranged_2_3,1)
                for j18 = 2:size(sphere_device_coord_arranged_2_3{j17,j15},2)
                    if (nodes_location{j16}(1) ==
sphere_device_coord_arranged_2_3{j17,j15}{1,j18}(1) && nodes_location{j16}(2)
== sphere_device_coord_arranged_2_3{j17,j15}{1,j18}(2) &&
nodes_location{j16}(3) == sphere_device_coord_arranged_2_3{j17,j15}{1,j18}(3)
&& nodes_location{j16}(4) ==
sphere_device_coord_arranged_2_3{j17,j15}{1,j18}(4))
                        priority_sphere_B{j16,1} = nodes_location{j16};
                        priority_sphere_B{j16,2} =
sphere_device_coord_arranged_2_3{j17,j15};
                        priority_sphere_B{j16,3} = j16;
                    end
                end
            end
        end
    end

    priority_sphere_B_1 =
priority_sphere_B(~cellfun('isempty',priority_sphere_B(:,2)), :); %remove all
empty cells (spheres that have been deleted)
    total_node_E2 = 0;
    for i = 1:size(priority_sphere_B_1,1)
        total_node_E2 = priority_sphere_B_1{i,2}{1,2}(5) + total_node_E2;
    end
    priority_sphere_B_11 = cell(size(priority_sphere_B_1,1),2);
    %Put devices in accending order
    count = priority_n_2;
    count_2 = 1;
```

```
    for j15 = M+1:priority_n_2
        for j16 = 1:size(priority_sphere_B_1,1) %toggle between list1
            if size(priority_sphere_B_1{j16,2},2) == count
                priority_sphere_B_11{count_2,4} = priority_sphere_B_1{j16,1};
                priority_sphere_B_11{count_2,5} = priority_sphere_B_1{j16,2};
                priority_sphere_B_11{count_2,6} = priority_sphere_B_1{j16,3};
            end
            count_2 = count_2 + 1;
        end
        count = count-1;
    end
    priority_sphere_B_11 =
priority_sphere_B_11(~cellfun('isempty',priority_sphere_B_11(:,4)), :);
%remove all empty cells (spheres that have been deleted)
    priority_sphere_B_11(:,1) = priority_sphere_B_11(:,4);
    priority_sphere_B_11(:,2) = priority_sphere_B_11(:,5);
    priority_sphere_B_11(:,3) = priority_sphere_B_11(:,6);

    priority_sphere_B_2 = priority_sphere_B_11;
    %for j15 = priority_n_2:-1:2 %number of priorities
        for j16 = 1:size(priority_sphere_B_11,1) %toggle between list1
            for j17 = j16+1:size(priority_sphere_B_11,1)
                for j18 = M+1:size(priority_sphere_B_11{j16,2},2)
                    for j19 = M+1:size(priority_sphere_B_11{j17,2},2) %toggle
in each trasmitter
                        if (priority_sphere_B_11{j16,2}{1,j18}(1) ==
priority_sphere_B_11{j17,2}{1,j19}(1) &&
priority_sphere_B_11{j16,2}{1,j18}(2) ==
priority_sphere_B_11{j17,2}{1,j19}(2) &&
priority_sphere_B_11{j16,2}{1,j18}(3) ==
priority_sphere_B_11{j17,2}{1,j19}(3) &&
priority_sphere_B_11{j16,2}{1,j18}(4) ==
priority_sphere_B_11{j17,2}{1,j19}(4))
                            priority_sphere_B_2{j17,2}{1,j19} = [];
                        end
                    end
                end
            end
        end
    %end


    for j12 = 1:size(priority_sphere_B_2,1) %rearange cells to only contains
actual values, no empty lists.
        temp_3 = priority_sphere_B_2{j12,2}(1,:);
        priority_sphere_B_2{j12,2} =
temp_3(~cellfun('isempty',priority_sphere_B_2{j12,2}(1,:)));
    end


    for j19 = 1:size(priority_sphere_B_2,1) %Remove all priority spheres that
are equal
        for j20 = j19+1:(size(priority_sphere_B_2,1))
            if size(priority_sphere_B_2{j19,2},2) ==
size(priority_sphere_B_2{j20,2},2)
```

```
                if
cellfun(@isequal,priority_sphere_B_2{j19,2},priority_sphere_B_2{j20,2})
                    priority_sphere_B_2{j20,2} = {};
                end
            end
        end
    end

    %Remove list that don't contains devices
    for j = 1:size(priority_sphere_B_2,1)
        if size(priority_sphere_B_2{j,2},2) == 1
            priority_sphere_B_2{j,2} = [];
        end
    end

    priority_sphere_B_3 =
priority_sphere_B_2(~cellfun('isempty',priority_sphere_B_2(:,2)), :); %Actual
number of sphere that can contains devices.

    A1_stationary_device_transmitter_direct_number(1) =
size(priority_sphere_B_3,1);
    stationary_device_number_2 = 0;
    for j21= 1:size(priority_sphere_B_3,1)
        stationary_device_number_2 = stationary_device_number_2 +
size(priority_sphere_B_3{j21,2},2)-1;
    end
    A1_stationary_device_transmitter_direct_number(2) =
stationary_device_number_2;
end
```

getFriisEquation

```
function [Pt, Pr1, Pr2, d1, d2, Gt1, Gt2, Gr, Freq,
Gt_lin,E_active,E_standby,E_device] = getFriisEquation(duty_cycle)
    %% Designer Input
    %Energy produced by Tx
    % Friis equation
    Pt = 3; %% input %% Power transmit(W)
    Gt1 = 0; %% input %% Type of transmitter antenna(dBi) 0 dBi for
omnidirectional antenna | 8 dBi for directional
    Gt2 = 8; %% 8 dBi for directional
    Gr = 6; %% input %% Type of receiver antenna(dBi) 2.04 for dipole antenna
| 6 for directional antenna
    Freq = 0.915; %% input %% Frequency(GHz)
    incd = 0; % variable holder
    incd2 = 0.01; %% Move distance from transmitter to receiver every "incd2"
meter
    d1 = zeros(1,10/incd2);
    d2 = zeros(1,10/incd2);
    Pr1 = zeros(1,10/incd2);
    Pr2 = zeros(1,10/incd2);

    % Distance and Power received for omnidirectional
    for i1 = 1:(10/incd2) %from 0.5m to 5m with increment of "incd2" meter
```

```
        d1(i1) = 0.5 + incd;
        incd = incd + incd2;
        Pr1(i1) =
(Pt*10^(Gt1/10)*10^(Gr/10)*((3*10^8)/(Freq*10^9))^2)/(4*pi()*d1(i1))^2; %%
Output %%Power received(W)
    end

    incd = 0; % variable holder
    % Distance and Power received for directional
    for j1 = 1:(10/incd2) %from 0.5m to 5m with increment of "incd2" meter
        d2(j1) = 0.5 + incd;
        incd = incd + incd2;
        Pr2(j1) =
(Pt*10^(Gt2/10)*10^(Gr/10)*((3*10^8)/(Freq*10^9))^2)/(4*pi()*d2(j1))^2; %%
Output %%Power received(W)
    end
    %dBi to liner gain
    Gt_lin = 10^(Gt2/10);




    %%% Nodes components choice
    t_active_s = duty_cycle; %% input %% duty cycle
    t_active_h = t_active_s/3600; %% input %% Amount of time the sensor
collect data (s/h or %)
    t_standby = 1-t_active_s; % Amount of time the sensor sleeps
    V_sensor = 5; %% input %%Voltage of sensor (V)
    I_sensor_active = 1; %% input %% Supply current of sensor (mA)
    I_sensor_standby = 0; %% input %% Standby current of sensor (mA)|
ADXL1002 : 0.225mA standby. 0 if sensor is Off
    E_sensor_active = I_sensor_active * t_active_h * V_sensor; %mAh
    E_sensor_standby = I_sensor_standby * t_standby * V_sensor; %mAh

    %%% MCU and radio
    sampling_rate = 76800; %% input %%
    ADC_bit_resolution = 16; %% input %%
    ADC_bit_rate = (sampling_rate*t_active_s)/ADC_bit_resolution;

    wl_flash = 32; %% input %% flash memory word length
    t_flash_prog = 0.0000456; %% input %% amount of time to write in the
flash
    t_flash_w_s = (ADC_bit_rate/wl_flash)*t_flash_prog;
    t_flash_w_h = t_flash_w_s/3600;

    radio_data_rate = 2000000; %% input %% bit/s
    t_radio_active = (ADC_bit_rate/radio_data_rate)/3600;

    t_flash_e = 0.0144/3600; %% input %%
    t_MCU_active = t_active_h+t_flash_w_h+t_radio_active+t_flash_e;

    I_MCU_active = 0.64; %% input %% mA
    I_flash_w = 1.34; %% input %% mA
    V_MCU = 3.3; %% input %% V
    E_MCU_active = (I_MCU_active+I_flash_w)*t_MCU_active*V_MCU;
```

```
    I_radio = 8.5; %% input %% mA
    E_radio = I_radio*t_radio_active*V_MCU;

    t_MCU_standby = 1-t_MCU_active;

    I_MCU_standby = 0.00194; %% input %% mA
    E_MCU_standby = I_MCU_standby*t_MCU_standby*V_MCU;

    %%% PMU
    I_PMU_active = 0.0023; %% input %% mA
    I_PMU_standby = 0.000018; %% input %% mA
    V_PMU = 3; %% input %% V
    E_PMU_active = I_PMU_active*t_MCU_active*V_PMU;
    E_PMU_standby = I_PMU_standby*t_MCU_standby*V_PMU;

    %%% Devive
    E_active = E_MCU_active+E_radio+E_PMU_active+E_sensor_active;
    E_standby = E_MCU_standby+E_PMU_standby+E_sensor_standby;
    E_device = E_active+E_standby;
end
```

getMobileDirect.m

```
function
[A3_mobile_device_transmitter_direct_number,node_life_time_B,zone_B4] =
getMobileDirect(priority_sphere_B_3,E_required_for_xtime,full_charge_time,vit
esse)
    time_fully_charge_2 = zeros(size(priority_sphere_B_3,1),1); %time to
fully charge a device based on energy received from transmitter
    lowest_charge_rate_2 = zeros(size(priority_sphere_B_3,1),1);
    E_received_for_xtime_2 = zeros(size(priority_sphere_B_3,1),1);
    for j21 = 1:size(priority_sphere_B_3,1)
        lowest_charge_rate_2(j21,1) = inf;
        for j22 = 2:size(priority_sphere_B_3{j21,2},2)
            if priority_sphere_B_3{j21,2}{j22}(5) <
lowest_charge_rate_2(j21,1)
                lowest_charge_rate_2(j21,1) =
priority_sphere_B_3{j21,2}{j22}(5); %mAh
            end
        end
        E_received_for_xtime_2(j21,1) = E_required_for_xtime; % Energy
received after fully charged
        time_fully_charge_2(j21,1) =
E_required_for_xtime/lowest_charge_rate_2(j21,1); % hour of charge require to
work for time hour
    end
    num_distances = 0;
    distance_between_zones_2 =
zeros(size(priority_sphere_B_3,1),size(priority_sphere_B_3,1));
    travel_time_2 =
zeros(size(priority_sphere_B_3,1),size(priority_sphere_B_3,1));
    for j23 = 1:size(priority_sphere_B_3,1)
        for j24 = j23:size(priority_sphere_B_3,1)
            num_distances = num_distances + 1;
```

```
            distance_between_zones_2(j23,j24) =
(priority_sphere_B_3{j23,2}{1}(1)-
priority_sphere_B_3{j24,2}{1}(1))^2+(priority_sphere_B_3{j23,2}{1}(2)-
priority_sphere_B_3{j24,2}{1}(2))^2+(priority_sphere_B_3{j23,2}{1}(3)-
priority_sphere_B_3{j24,2}{1}(3))^2;
            distance_between_zones_2(j24,j23) =
(priority_sphere_B_3{j23,2}{1}(1)-
priority_sphere_B_3{j24,2}{1}(1))^2+(priority_sphere_B_3{j23,2}{1}(2)-
priority_sphere_B_3{j24,2}{1}(2))^2+(priority_sphere_B_3{j23,2}{1}(3)-
priority_sphere_B_3{j24,2}{1}(3))^2;
            travel_time_2(j23,j24) =
distance_between_zones_2(j23,j24)/vitesse*(1/3600);
            travel_time_2(j24,j23) =
distance_between_zones_2(j24,j23)/vitesse*(1/3600);
        end
    end

    %% Zone finding
    zone_B = cell(size(priority_sphere_B_3,1),3);
    value_22 = 1;
    while value_22 < size(priority_sphere_B_3,1)
        global_charge_2 = full_charge_time;
        zone_device_number_2 = 0;
        for j26 = value_22:size(priority_sphere_B_3,1)
            if j26 ~= value_22
                value_2 = j26-1;
            else
                value_2 = value_22;
            end
            if (global_charge_2 - 2*travel_time_2(value_2,j26) -
time_fully_charge_2(j26)) > 0
                global_charge_2 = global_charge_2 -
2*travel_time_2(value_2,j26) - time_fully_charge_2(j26);
                zone_device_number_2 = zone_device_number_2 +
size(priority_sphere_B_3{j26,2},2)-1;
                zone_B{value_22, 1}{value_22} =
priority_sphere_B_3{value_22,2}{1};
                zone_B{value_22, 1}{j26} = priority_sphere_B_3{j26,2}{1};
            else
                break;
            end
        end
        zone_B{value_22,2} =
zone_device_number_2+size(priority_sphere_B_3{value_22,2},2)-1;
        zone_B{value_22,3} = value_22;
        value_22 = j26;
    end
    zone_B2 = zone_B(~cellfun('isempty',zone_B(:,1)),:);
    if (~cellfun('isempty',zone_B2{1,1}))
        zone_B2{1,1}(~cellfun('isempty',zone_B2{1,1}));
    end
    zone_B3 = cell(size(zone_B2,1),3);
    for j27 = 1:size(zone_B2,1)
        zone_B3{j27,1} = zone_B2{j27,1}(~cellfun('isempty',zone_B2{j27,1}));
        zone_B3{j27,2} = zone_B2{j27,2};
        zone_B3{j27,3} = zone_B2{j27,3};
    end
```

```
    zone_B4 = zone_B3;
    for j28 = 1:size(priority_sphere_B_3,1)
        zone_count = 0;
        for j29 = 1:size(zone_B2,1)
            for j30 = 1:size(zone_B3{j29,1},2)
                if ((priority_sphere_B_3{j28,2}{1}(1) ==
zone_B3{j29,1}{1,j30}(1)) && (priority_sphere_B_3{j28,2}{1}(2) ==
zone_B3{j29,1}{1,j30}(2)) && (priority_sphere_B_3{j28,2}{1}(3) ==
zone_B3{j29,1}{1,j30}(3)))
                    if zone_count >= 1
                       zone_B4{j29,1}{1,j30} = [];
                    end
                    if zone_count == 0
                        zone_count = zone_count + 1;
                    end
                end
            end
        end
    end
    count = 0;
    zone_4_device_2 = zeros(1,size(zone_B,1));
    for j34 = 1:size(zone_B4,1)
        count_2 = 0;
        for j35 = 1:size(zone_B4{j34,1},2)
            if ~isempty(zone_B4{j34,1}{1,j35})
                count = count + 1;
                zone_4_device_2(count) = zone_B4{j34,3}+count_2;
                count_2 = count_2 + 1;
            else
                count_2 = count_2 + 1;
            end
        end
    end
    zone_4_device_22 = zone_4_device_2;
    for j = size(zone_4_device_22,2):-1:1
        if zone_4_device_22(j) == 0
            zone_4_device_2(j) = [];
        end
    end

    zone_device_number_2_2 = 0;
    A3_mobile_device_transmitter_direct_number(1) = size(zone_B4,1);
    %count = 1;
    for j36 = 1:size(zone_4_device_2,2) %Find the number of devices charged
        zone_device_number_2_2 =
zone_device_number_2_2+size(priority_sphere_B_3{zone_4_device_2(j36),2},2)-1;
    %     for j = 2:size(priority_sphere_B_3{zone_4_device_2(j36),2},2)
    %         MD_covered_x(count) =
priority_sphere_B_3{zone_4_device_2(j36),2}{1,j}(1);
    %         MD_covered_y(count) =
priority_sphere_B_3{zone_4_device_2(j36),2}{1,j}(2);
    %         MD_covered_z(count) =
priority_sphere_B_3{zone_4_device_2(j36),2}{1,j}(3);
    %         count = count + 1;
    %     end
    end
    node_life_time_B = (time_fully_charge_2*10000)/8760;
```

```
    A3_mobile_device_transmitter_direct_number(2) = zone_device_number_2_2;
end
```

getMobileOmni.m

```
function [A2_mobile_device_transmitter_omni_number,node_life_time_A,zone_A4]
=
getMobileOmni(priority_sphere_A_3,E_required_for_xtime,full_charge_time,vites
se)
    time_fully_charge_1 = zeros(size(priority_sphere_A_3,1),1);%time to fully
charge a device based on energy received from transmitter
    lowest_charge_rate_1 = zeros(size(priority_sphere_A_3,1),1); %longest
time a device in a zone can be charged
    E_received_for_xtime_1= zeros(size(priority_sphere_A_3,1),1);  % Energy
received after fully charged
    %charge_time_1= zeros(size(priority_sphere_A_3,1),1);
    for i21 = 1:size(priority_sphere_A_3,1) %Find the device that will take
the longest to charge within a zone
        lowest_charge_rate_1(i21,1) = inf;
        for i22 = 2:size(priority_sphere_A_3{i21,2},2)
            if priority_sphere_A_3{i21,2}{i22}(5) <
lowest_charge_rate_1(i21,1)
                lowest_charge_rate_1(i21,1) =
priority_sphere_A_3{i21,2}{i22}(5); %mAh
            end
        end
        time_fully_charge_1(i21,1) =
E_required_for_xtime/lowest_charge_rate_1(i21,1); %time to fully charge a
device based on energy received from transmitter
        %time_fully_charge_1(i21,2) = i21;
        E_received_for_xtime_1(i21,1) = E_required_for_xtime; % Energy
received after fully charged
    end
    %time_fully_charge_1 = sortrows(time_fully_charge_1);
    %travel_time
    num_distances = 0;
    distance_between_zones_1 =
zeros(size(priority_sphere_A_3,1),size(priority_sphere_A_3,1)); % distance
between 2 zones
    travel_time_1 =
zeros(size(priority_sphere_A_3,1),size(priority_sphere_A_3,1)); % travel time
between 2 zones
    for i22 = 1:size(priority_sphere_A_3,1) % Determine if a transmitter can
charge multiple zones
        for i23 = i22:size(priority_sphere_A_3,1)
            num_distances = num_distances + 1;
            distance_between_zones_1(i22,i23) =
(priority_sphere_A_3{i22,2}{1}(1)-
priority_sphere_A_3{i23,2}{1}(1))^2+(priority_sphere_A_3{i22,2}{1}(2)-
priority_sphere_A_3{i23,2}{1}(2))^2+(priority_sphere_A_3{i22,2}{1}(3)-
priority_sphere_A_3{i23,2}{1}(3))^2;
            distance_between_zones_1(i23,i22) =
(priority_sphere_A_3{i22,2}{1}(1)-
priority_sphere_A_3{i23,2}{1}(1))^2+(priority_sphere_A_3{i22,2}{1}(2)-
```

```
priority_sphere_A_3{i23,2}{1}(2))^2+(priority_sphere_A_3{i22,2}{1}(3)-
priority_sphere_A_3{i23,2}{1}(3))^2;
            travel_time_1(i22,i23) =
distance_between_zones_1(i22,i23)/vitesse*(1/3600); % travel time between 2
zones
            travel_time_1(i23,i22) =
distance_between_zones_1(i23,i22)/vitesse*(1/3600); % travel time between 2
zones
        end
    end

    %% Zone finding

    zone_A = cell(size(priority_sphere_A_3,1),3);
    value_11 = 1;
    while value_11 < size(priority_sphere_A_3,1)
        global_charge_1 = full_charge_time;
        zone_device_number_1 = 0;
        for i25 = value_11:size(priority_sphere_A_3,1)
            if i25 ~= value_11
                value_1 = i25-1;
            else
                value_1 = value_11;
            end
            if (global_charge_1 - 2*travel_time_1(value_1,i25) -
time_fully_charge_1(i25,1)) > 0
                global_charge_1 = global_charge_1 -
2*travel_time_1(value_1,i25) - time_fully_charge_1(i25,1);
                zone_device_number_1 = zone_device_number_1 +
size(priority_sphere_A_3{i25,2},2)-1;
                zone_A{value_11, 1}{value_11} =
priority_sphere_A_3{value_11,2}{1};
                zone_A{value_11, 1}{i25} = priority_sphere_A_3{i25,2}{1};
            else
                break;
            end
        end
        zone_A{value_11,2} =
zone_device_number_1+size(priority_sphere_A_3{value_11,2},2)-2;
        zone_A{value_11,3} = value_11;
        value_11 = i25;
    end
    zone_A2 = zone_A(~cellfun('isempty',zone_A(:,1)),:);
    if (~cellfun('isempty',zone_A2{1,1}))
        zone_A2{1,1}(~cellfun('isempty',zone_A2{1,1}));
    end
    zone_A3 = cell(size(zone_A2,1),3);
    for i26 = 1:size(zone_A2,1)
        zone_A3{i26,1} = zone_A2{i26,1}(~cellfun('isempty',zone_A2{i26,1}));
        zone_A3{i26,2} = zone_A2{i26,2};
        zone_A3{i26,3} = zone_A2{i26,3};
    end
    zone_A4 = zone_A3;
    for i27 = 1:size(priority_sphere_A_3,1)
        zone_count = 0;
        for i28 = 1:size(zone_A2,1)
```

```
            for i29 = 1:size(zone_A3{i28,1},2)
                if ((priority_sphere_A_3{i27,2}{1}(1) ==
zone_A3{i28,1}{1,i29}(1)) && (priority_sphere_A_3{i27,2}{1}(2) ==
zone_A3{i28,1}{1,i29}(2)) && (priority_sphere_A_3{i27,2}{1}(3) ==
zone_A3{i28,1}{1,i29}(3)))
                    if zone_count >= 1
                       zone_A4{i28,1}{1,i29} = [];
                        %zone_4{i27,1} =
zone_4{i27,1}(~cellfun('isempty',zone_4{i27,1}));
                    end
                    if zone_count == 0
                        zone_count = zone_count + 1;
                    end
                end
            end
        end
    end
    % for i = 1:size(zone_A4,1) % remove empty cells after sorting
    %     if (cellfun('isempty',zone_A4{i,1}))
    %         zone_A4{i,1} = {};
    %     end
    % end
    zone_A4 = zone_A4(~cellfun('isempty',zone_A4(:,1)),:);
    count = 0;
    zone_4_device_1 = zeros(1,size(zone_A,1));
    for i30 = 1:size(zone_A4,1)
        count_2 = 0;
        for i31 = 1:size(zone_A4{i30,1},2)
            if ~isempty(zone_A4{i30,1}{1,i31})
                count = count + 1;
                zone_4_device_1(count) = zone_A4{i30,3}+count_2;
                count_2 = count_2 + 1;
            else
                count_2 = count_2 + 1;
            end
        end
    end
    zone_device_number_1_1 = 0;
    A2_mobile_device_transmitter_omni_number(1) = size(zone_A4,1);
    zone_4_device_11 = zone_4_device_1;

    for j = size(zone_4_device_11,2):-1:1
        if zone_4_device_11(j) == 0
            zone_4_device_1(j) = [];
        end
    end
    % count = 1;
    for i32 = 1:size(zone_4_device_1,2)
        zone_device_number_1_1 =
zone_device_number_1_1+size(priority_sphere_A_3{zone_4_device_1(i32),2},2)-1;
    %     for j = 2:size(priority_sphere_A_3{zone_4_device_1(i32),2},2)
    %         MO_covered_x(count) =
priority_sphere_A_3{zone_4_device_1(i32),2}{1,j}(1);
    %         MO_covered_y(count) =
priority_sphere_A_3{zone_4_device_1(i32),2}{1,j}(2);
    %         MO_covered_z(count) =
priority_sphere_A_3{zone_4_device_1(i32),2}{1,j}(3);
```

```
    %     end
    %     count = count + 1;
    end
    node_life_time_A = (time_fully_charge_1);
    A2_mobile_device_transmitter_omni_number(2) = zone_device_number_1_1;
end
```

getNodeLocation.m

```
function [xl,yl,zl] =
getNodeLocation(r1,r2,r3,plant_height,number_nodes,node_location_type)
    num_node_rec1 = number_nodes*3/10;
    num_node_rec2 = number_nodes*2/10;
    num_node_rec3 = number_nodes*5/10;

    if strcmp(node_location_type,'random')
        %rectangle1
        x1l = r1(1,1) + (r1(1,2)-r1(1,1)).*rand(1,round(num_node_rec1)); % x
coordinates of equipment to monitor
        y1l = r1(2,1) + (r1(2,2)-r1(2,1)).*rand(1,round(num_node_rec1)); % y
coordinates of equipment to monitor
        z1l = (plant_height-2)*rand(1,round(num_node_rec1)); % z coordinates
of equipment to monitor
        %rectangle2
        x2l = r2(1,1) + (r2(1,2)-r2(1,1)).*rand(1,round(num_node_rec2)); % x
coordinates of equipment to monitor
        y2l = r2(2,1) + (r2(2,2)-r2(2,1)).*rand(1,round(num_node_rec2)); % y
coordinates of equipment to monitor
        z2l = (plant_height-2)*rand(1,round(num_node_rec2)); % z coordinates
of equipment to monitor
        %rectangle3
        x3l = r3(1,1) + (r3(1,2)-r3(1,1)).*rand(1,round(num_node_rec3)); % x
coordinates of equipment to monitor
        y3l = r3(2,1) + (r3(2,2)-r3(2,1)).*rand(1,round(num_node_rec3)); % y
coordinates of equipment to monitor
        z3l = (plant_height-2)*rand(1,round(num_node_rec3)); % z coordinates
of equipment to monitor
    elseif strcmp(node_location_type,'uniform spacing')
        %rectangle1
        count = 0;
        inc1 = (r1(1,2)-r1(1,1))/(round(num_node_rec1^(1/3))-1);
        inc2 = (r1(2,2)-r1(2,1))/(round(num_node_rec1^(1/3))-1);
        inc3 = (plant_height-1)/(round(num_node_rec1^(1/3))-1);
        for i1 = r1(1,1):inc1:r1(1,2)
            for i2 = r1(2,1):inc2:r1(2,2)
                for i3 = 1:inc3:plant_height
                    count = count + 1;
                x1l(count) = i1;
                y1l(count) = i2;
                z1l(count) = i3;
                end
            end
        end
        %rectangle2
        count = 0;
```

```
    inc1 = (r2(1,2)-r2(1,1))/(round(num_node_rec2^(1/3))-1);
    inc2 = (r2(2,2)-r2(2,1))/(round(num_node_rec2^(1/3))-1);
    inc3 = (plant_height-1)/(round(num_node_rec2^(1/3))-1);
    for i1 = r2(1,1):inc1:r2(1,2)
        for i2 = r2(2,1):inc2:r2(2,2)
            for i3 = 1:inc3:plant_height
                count = count + 1;
            x2l(count) = i1;
            y2l(count) = i2;
            z2l(count) = i3;
            end
        end
    end
    %rectangle3
    count = 0;
    inc1 = (r3(1,2)-r3(1,1))/(round(num_node_rec3^(1/3))-1);
    inc2 = (r3(2,2)-r3(2,1))/(round(num_node_rec3^(1/3))-1);
    inc3 = (plant_height-1)/(round(num_node_rec3^(1/3))-1);
    %inc3 = ceil(num_node_rec3^(1/3))^3-num_node_rec3;
    for i1 = r3(1,1):inc1:r3(1,2)
        for i2 = r3(2,1):inc2:r3(2,2)
            for i3 = 1:inc3:plant_height
                count = count + 1;
            x3l(count) = i1;
            y3l(count) = i2;
            z3l(count) = i3;
            end
        end
    end
elseif strcmp(node_location_type,'clumped')
    %rectangle1
    rng(0,'twister')
    Rayon = num_node_rec1/number_nodes;     % radius of the sphere
    for i = 1:num_node_rec1
        rval = 2*rand(1)-1;
        elevation = asin(rval);
        azimuth = 2*pi*rand(1);
        radius = Rayon*(rand(1))^(1/3);
        [x,y,z] = sph2cart(azimuth,elevation,radius);
        x1l(i)=x+Rayon+r1(1,1);
        y1l(i)=y+Rayon+r1(1,2);
        z1l(i)=z+Rayon+2;
    end
    %rectangle2
    rng(0,'twister')
    Rayon = num_node_rec2/number_nodes;     % radius of the sphere
    for i = 1:num_node_rec2
        rval = 2*rand(1)-1;
        elevation = asin(rval);
        azimuth = 2*pi*rand(1);
        radius = Rayon*(rand(1))^(1/3);
        [x,y,z] = sph2cart(azimuth,elevation,radius);
        x2l(i)=x+Rayon+r2(1,1)-2;
        y2l(i)=y+Rayon+r2(1,2)-2;
        z2l(i)=z+Rayon+2;
    end
    %rectangle3
```

```
        rng(0,'twister')
        Rayon = num_node_rec3/number_nodes;     % radius of the sphere
        for i = 1:num_node_rec3
            rval = 2*rand(1)-1;
            elevation = asin(rval);
            azimuth = 2*pi*rand(1);
            radius = Rayon*(rand(1))^(1/3);
            [x,y,z] = sph2cart(azimuth,elevation,radius);
            x3l(i)=x+Rayon+r3(1,1);
            y3l(i)=y+Rayon+r3(1,2);
            z3l(i)=z+Rayon+2;
        end
    end
    if strcmp(node_location_type,'manual')
        xl =
[2.4664,2.3697,3.5970,4.4028,4.5050,2.5898,3.2871,3.0027,3.3967,2.7063,2.1823
,3.1635,2.4636,2.0696 ...

,9.4436,10.125,9.4801,10.0617,10.4167,11.0438,8.87787,11.0618,9.85751,10.3770
,10.3864,8.33661 ...

,10.2243,9.10088,3.35541,7.54838,5.81987,3.01362,3.62634,4.17987,9.39497,6.74
662,9.63844,4.82463 ...

,2.10735,7.85511,5.07331,12.4085,11.9709,6.31953,2.27341,9.38580];
        yl =
[12.7960,10.4377,12.8270,11.9309,8.5331,11.5602,11.6768,11.4031,10.9854,9.481
5,11.9756,9.5261 ...

,10.1689,12.5092,9.5868,9.5941,9.1937,12.8039,12.7105,12.1842,11.7772,9.2911,
10.1217,5.7673,6.3718 ...

,2.2562,4.0264,4.6211,5.0899,5.2374,4.9252,5.2711,3.6823,4.6171,2.5225,2.2594
,6.4089,3.2817,4.6774 ...
            ,6.3297,2.8360,2.8687,3.5374,6.1980,3.7580,3.2295];
        zl =
[7.7073,3.6219,2.8622,5.4777,8.1918,8.1819,5.3243,2.9931,7.6776,3.9816,8.1392
,0.2986,4.7918,6.4485 ...

,1.6991,0.0108,2.8478,6.2966,5.6273,4.8876,3.9513,2.5868,4.5149,1.3675,3.5740
,3.3725,1.1800,3.9154 ...

,0.8236,5.5316,0.0988,5.1593,7.1076,2.1183,4.0321,5.1242,0.5526,4.4666,5.7808
,1.9914,7.5335,8.7397 ...
            ,7.6173,4.5540,2.5099,6.7196];
    else
        xl = [x1l, x2l, x3l];
        yl = [y1l, y2l, y3l];
        zl = [z1l, z2l, z3l];
    end
end
```

getOmniCoverage.m

```
function
[priority_sphere_A_3,A0_stationary_device_transmitter_omni_number,total_node_
E] = getOmniCoverage(sphere_device_coord_arranged_1_2,nodes_location,M)
```

```
    % Find the number of devices that each spheres contains
    [ss1,dd1] = cellfun(@size,sphere_device_coord_arranged_1_2);
    priority_n_1 = max(dd1);
    count_prio = 0;
    for i12 = priority_n_1-count_prio:-1:M+1 %remove all sphere that contains
less than the minimum amount of devices required
        for i13 = 1:size(sphere_device_coord_arranged_1_2,1)
            if size(sphere_device_coord_arranged_1_2{i13,1},2) == i12
                sphere_device_coord_arranged_1_3{i13,i12} =
sphere_device_coord_arranged_1_2{i13,1}; %Place each sphere in different
column based on their priority.
            end
        end
        count_prio = count_prio + 1;
    end

    priority_sphere_A = cell(size(nodes_location,2),2);

    %sphere_device_coord_arranged_1_4 = sphere_device_coord_arranged_1_3;
    for i14 = M+1:priority_n_1 %If a device is found in 2+ priority spheres,
only the highest priority sphere will be kept. The others will be removed.
        for i15 = 1:size(nodes_location,2) %toggle between list1
            for i16 = 1:size(sphere_device_coord_arranged_1_3,1)
                for i17 = 2:size(sphere_device_coord_arranged_1_3{i16,i14},2)
%toggle in each trasmitter
                    if (nodes_location{i15}(1) ==
sphere_device_coord_arranged_1_3{i16,i14}{1,i17}(1) && nodes_location{i15}(2)
== sphere_device_coord_arranged_1_3{i16,i14}{1,i17}(2) &&
nodes_location{i15}(3) == sphere_device_coord_arranged_1_3{i16,i14}{1,i17}(3)
&& nodes_location{i15}(4) ==
sphere_device_coord_arranged_1_3{i16,i14}{1,i17}(4))
                        priority_sphere_A{i15,1} = nodes_location{i15};
                        priority_sphere_A{i15,2} =
sphere_device_coord_arranged_1_3{i16,i14};
                        priority_sphere_A{i15,3} = i15;
                    end
                end
            end
        end
    end

    priority_sphere_A_1 =
priority_sphere_A(~cellfun('isempty',priority_sphere_A(:,2)), :); %remove all
empty cells (spheres that have been deleted)
    total_node_E = 0;
    for i = 1:size(priority_sphere_A_1,1)
        total_node_E = priority_sphere_A_1{i,2}{1,2}(5) + total_node_E;
    end
    priority_sphere_A_11 = cell(size(priority_sphere_A_1,1),6);
    %Put devices in accending order
    count = priority_n_1;
    count_1 = 1;
    for i15 = M+1:priority_n_1
        for i16 = 1:size(priority_sphere_A_1,1) %toggle between list1
            if size(priority_sphere_A_1{i16,2},2) == count
                priority_sphere_A_11{count_1,4} = priority_sphere_A_1{i16,1};
```

```
                priority_sphere_A_11{count_1,5} = priority_sphere_A_1{i16,2};
                priority_sphere_A_11{count_1,6} = priority_sphere_A_1{i16,3};
            end
            count_1 = count_1 + 1;
        end
        count = count-1;
    end
    priority_sphere_A_11 =
priority_sphere_A_11(~cellfun('isempty',priority_sphere_A_11(:,4)), :);
%remove all empty cells (spheres that have been deleted)
    priority_sphere_A_11(:,1) = priority_sphere_A_11(:,4);
    priority_sphere_A_11(:,2) = priority_sphere_A_11(:,5);
    priority_sphere_A_11(:,3) = priority_sphere_A_11(:,6);

    priority_sphere_A_2 = priority_sphere_A_11;
        for i16 = 1:size(priority_sphere_A_11,1) %toggle between list1
            for i17 = i16+1:size(priority_sphere_A_11,1)
                for i18 = M+1:size(priority_sphere_A_11{i16,2},2)
                    for i19 = M+1:size(priority_sphere_A_11{i17,2},2) %toggle
in each trasmitter
                        if (priority_sphere_A_11{i16,2}{1,i18}(1) ==
priority_sphere_A_11{i17,2}{1,i19}(1) &&
priority_sphere_A_11{i16,2}{1,i18}(2) ==
priority_sphere_A_11{i17,2}{1,i19}(2) &&
priority_sphere_A_11{i16,2}{1,i18}(3) ==
priority_sphere_A_11{i17,2}{1,i19}(3) &&
priority_sphere_A_11{i16,2}{1,i18}(4) ==
priority_sphere_A_11{i17,2}{1,i19}(4))
                            priority_sphere_A_2{i17,2}{1,i19} = [];
                        end
                    end
                end
            end
        end

    for i12 = 1:size(priority_sphere_A_2,1) %rearange cells to only contains
actual values, no empty lists.
        temp_4 = priority_sphere_A_2{i12,2}(1,:);
        priority_sphere_A_2{i12,2} =
temp_4(~cellfun('isempty',priority_sphere_A_2{i12,2}(1,:)));
    end


    for i18 = 1:size(priority_sphere_A_2,1) %Remove all priority spheres that
are equal
        for i19 = i18+1:(size(priority_sphere_A_2,1))
            if size(priority_sphere_A_2{i18,2},2) ==
size(priority_sphere_A_2{i19,2},2)
                if
cellfun(@isequal,priority_sphere_A_2{i18,2},priority_sphere_A_2{i19,2})
                    priority_sphere_A_2{i19,2} = {};
                end
            end
        end
    end
```

```
    for j = 1:size(priority_sphere_A_2,1)
        if size(priority_sphere_A_2{j,2},2) == 1
            priority_sphere_A_2{j,2} = [];
        end
    end

    priority_sphere_A_3 =
priority_sphere_A_2(~cellfun('isempty',priority_sphere_A_2(:,2)), :); %Actual
number of sphere that can contains devices.
    A0_stationary_device_transmitter_omni_number(1) =
size(priority_sphere_A_3,1);
    stationary_device_number = 0;
    for i20= 1:size(priority_sphere_A_3,1)
        stationary_device_number = stationary_device_number +
size(priority_sphere_A_3{i20,2},2)-1;
    end
    A0_stationary_device_transmitter_omni_number(2) =
stationary_device_number;
end
```

getZoneMobileDirect.m

```
function zone_mobile_B = getZoneMobileDirect(nodes_location,
sphere_device_coord_NLOS_2_2)
%% New mobile algorithm
    nodes_location{1,1}(1);
    sphere_device_coord_NLOS_2_2{1,1}(1);
    larger_power = zeros(1,5);
    sphere_larger_power = zeros(1,5);
    zone_mobile_B = cell(size(nodes_location,2),2);
    for i1 = 1:size(nodes_location,2)
        larger_power = sphere_device_coord_NLOS_2_2{1,2};
        for i2 = 1:size(sphere_device_coord_NLOS_2_2,1)
            if (nodes_location{1,i1}(1) ==
sphere_device_coord_NLOS_2_2{i2,2}(1) &&...
                nodes_location{1,i1}(2) ==
sphere_device_coord_NLOS_2_2{i2,2}(2) &&...
                nodes_location{1,i1}(3) ==
sphere_device_coord_NLOS_2_2{i2,2}(3))
                if (sphere_device_coord_NLOS_2_2{i2,2}(5) > larger_power(5))
                    larger_power = sphere_device_coord_NLOS_2_2{i2,2};
                    sphere_larger_power = sphere_device_coord_NLOS_2_2{i2,1};
                end
            end
        end
        zone_mobile_B{i1,1} = larger_power;
        zone_mobile_B{i1,2}{1,1} = sphere_larger_power;
        zone_mobile_B{i1,2}{1,2} = larger_power;
        zone_mobile_B{i1,3} = i1;
        zone_mobile_B{i1,4} = zone_mobile_B{i1,1};
        zone_mobile_B{i1,5} = zone_mobile_B{i1,2};
        zone_mobile_B{i1,6} = zone_mobile_B{i1,3};
    end
end
```

getZoneMobileOmni.m

```
function zone_mobile_A = getZoneMobileOmni(nodes_location,
sphere_device_coord_NLOS_1_1)
    %% New mobile algorithm
    zone_mobile_A = cell(size(nodes_location,2),2);
    larger_power = zeros(1,5);
    sphere_larger_power = zeros(1,5);
    for i1 = 1:size(nodes_location,2)
        larger_power = sphere_device_coord_NLOS_1_1{1,2};
            for i2 = 1:size(sphere_device_coord_NLOS_1_1,1)
                if (nodes_location{1,i1}(1) ==
sphere_device_coord_NLOS_1_1{i2,2}(1) &&...
                    nodes_location{1,i1}(2) ==
sphere_device_coord_NLOS_1_1{i2,2}(2) &&...
                    nodes_location{1,i1}(3) ==
sphere_device_coord_NLOS_1_1{i2,2}(3))
                    if (sphere_device_coord_NLOS_1_1{i2,2}(5) >
larger_power(5))
                        larger_power = sphere_device_coord_NLOS_1_1{i2,2};
                        sphere_larger_power =
sphere_device_coord_NLOS_1_1{i2,1};
                    end
                end
            end
        zone_mobile_A{i1,1} = larger_power;
        zone_mobile_A{i1,2}{1,1} = sphere_larger_power;
        zone_mobile_A{i1,2}{1,2} = larger_power;
        zone_mobile_A{i1,3} = i1;
        zone_mobile_A{i1,4} = zone_mobile_A{i1,1};
        zone_mobile_A{i1,5} = zone_mobile_A{i1,2};
        zone_mobile_A{i1,6} = zone_mobile_A{i1,3};
    end
end
```

Appendix C – Visualization of the Architecture Implementation Process

This appendix provides a visualization of the simulator implementation steps for different parameters.

Table C-1 and Table C-2 provide the final PTXs locations and power received by each node from their associated PTXs, respectively, for SO and SD. Table C-3 and Table C-4 provide the final PTXs locations and power received by each node from their associated PTXs, respectively, for MO and MD.

Table C-1. Final PTX locations and power received by each node for omni-directional antennas

PTX	Node #	Location (m)	Power received (mW)
[2.56,10.06,4.06]	1	[2.5,12.8,7.7]	0.0187
	10	[2.7,9.5,4]	0.0196
	13	[2.5,10.2,4.8]	0.0104
[2.56,12.46,8.26]	2	[2.4,10.4,3.6]	0.0177
	11	[2.2,12,8.1]	0.0265
[4.06,13.06,3.16]	3	[3.6,12.8,2.9]	0.0127
[3.76,12.16,5.56]	4	[4.4,11.9,5.5]	0.0183
	7	[3.3,11.7,5.3]	0.0123
[4.96,8.86,7.96]	5	[4.5,8.5,8.2]	0.0120
[3.16,11.56,7.96]	6	[2.6,11.6,8.2]	0.0146
	9	[3.4,11,7.7]	0.0118
[2.86,10.66,3.16]	8	[3.3,11.4,3]	0.0113
[3.76,9.76,0.46]	12	[3.2,9.5,0.3]	0.0135
[2.26,12.76,7.06]	14	[2.1,12.5,6.4]	0.0145
[9.76,9.46,2.26]	15	[9.4,9.6,1.7]	0.0139
	17	[9.5,9.2,2.8]	0.0132
[10.66,9.76,0.46]	16	[10.1,9.6,0]	0.0122
[10.66,13.066.16]	18	[10.1,12.8,6.3]	0.0104
	19	[10.4,12.7,5.6]	0.0150
[11.56,12.46,4.96]	20	[11,12.2,4.9]	0.0134
[9.46,11.86,4.06]	21	[8.9,11.8,4]	0.0127
[11.56,9.46,2.86]	22	[11.1,9.3,2.6]	0.0165
[10.36,10.66,4.66]	23	[9.9,10.1,4.5]	0.0102
[10.06,6.16,1.06]	24	[10.4,5.8,1.4]	0.0211
	37	[9.6,6.4,0.6]	0.0165
[10.96,6.76,3.76]	25	[10.4,6.4,3.6]	0.0154
[8.86,2.56,3.46]	26	[8.3,2.3,3.4]	0.0116
	35	[9.4,2.5,4]	0.0108
[10.66,4.36,1.36]	27	[10.2,4,1.2]	0.0146
[9.46,4.66,4.36]	28	[9.2,4.6,3.9]	0.0151
[4.06,5.26,0.76]	29	[3.4,5.1,0.8]	0.0115
[8.26,5.26,5.56]	30	[7.5,5.2,5.5]	0.0100
[6.46,4.96,0.46]	31	[5.8,4.9,0.1]	0.0139
[2.56,4.96,5.56]	32	[3,5.3,5.2]	0.0120
	39	[2.1,4.7,5.8]	0.0231
[4.36,3.76,7.06]	33	[3.6,3.7,7.1]	0.0108
[4.96,4.66,2.26]	34	[4.2,4.6,2.1]	0.0102
[7.36,2.56,5.26]	36	[6.7,2.3,5.1]	0.0109
[5.26,3.76,4.66]	38	[4.8,3.3,4.5]	0.0124
[8.26,6.76,2.26]	40	[7.9,6.3,2]	0.0115
[5.56,3.16,7.66]	41	[5.1,2.8,7.5]	0.0159
[12.16,3.16,8.26]	42	[12.4,2.9,8.7]	0.0144
	43	[12,3.5,7.6]	0.0117
[7.06,6.46,4.66]	44	[6.3,6.2,4.6]	0.0103
[2.86,4.06,2.86]	45	[2.3,3.8,2.5]	0.0104
[10.06,3.46,6.76]	46	[9.4,3.2,6.7]	0.0099

Table C-2. Final PTX locations and power received by each node for directional antennas

PTX	Node #	Location (m)	Power received (mW)
[2.7,12.1,7.4]	1	[2.5,12.8,7.7]	0.0570
	6	[2.6,11.6,8.2]	0.0158
	11	[2.2,12,8.1]	0.0418
	14	[2.1,12.5,6.4]	0.0098
[2.9,10.6,3.8]	2	[2.4,10.4,3.6]	0.1174
	8	[3,11.4,3]	0.0141
	10	[2.7,9.5,4]	0.0098
	13	[2.5,10.2,4.8]	0.0112
[3.6,12.4,3.4]	3	[3.6,12.8,2.9]	0.0353
[3.5,12.3,6.1]	4	[4.4,11.9,5.5]	0.0129
[4.7,9.5,8.4]	5	[4.5,8.5,8.2]	0.0104
[2.9,11,4.4]	7	[3.3,11.7,5.3]	0.0103
[2.7,11.7,7.6]	9	[3.4,11,7.7]	0.0153
[3.3,10.6,0.4]	12	[3.2,9.5,0.3]	0.0104
[10.3,9.6,2.2]	15	[9.4,9.6,1.7]	0.0130
	17	[9.5,9.2,2.8]	0.0119
	22	[11.1,9.3,2.6]	0.0230
[10.3,9.8,1]	16	[10.1,9.6,0]	0.0207
[9.4,12.8,6.1]	18	[10.1,12.8,6.3]	0.0646
	19	[10.4,12.7,5.6]	0.0146
[10.8,12.8,4.6]	20	[11,12.2,4.9]	0.0447
[9.7,11.2,4.1]	21	[8.9,11.8,4]	0.0101
	23	[9.9,10.1,4.5]	0.0106
[10.6,6.6,1]	24	[10.4,5.8,1.4]	0.0185
	37	[9.6,6.4,0.6]	0.0186
[10.3,6.2,2.4]	25	[10.4,6.4,3.6]	0.0112
[9.3,2.1,3.3]	26	[8.3,2.3,3.4]	0.0101
	35	[9.4,2.5,4]	0.0545
[10.7,4.9,1.5]	27	[10.2,4,1.2]	0.0099
[9.5,3.7,3.8]	28	[9.2,4.6,3.9]	0.0098
[4.4,5.1,1.4]	29	[3.4,5.1,0.8]	0.0100
	34	[4.2,4.6,2.1]	0.0461
[7.1,6.1,5.3]	30	[7.5,5.2,5.5]	0.0184
	44	[6.3,6.2,4.6]	0.0156
[6.5,5.9,0.1]	31	[5.8,4.9,0.1]	0.0114
[2.8,5.4,6]	32	[3,5.3,5.2]	0.0294
	39	[2.1,4.7,5.8]	0.0144
[4.6,3.6,7.6]	33	[3.6,3.7,7.1]	0.0136
	41	[5.1,2.8,7.5]	0.0180
[7.7,2.5,5.4]	36	[6.7,2.3,5.1]	0.0132
[5.6,3.9,5]	38	[4.8,3.3,4.5]	0.0102
[8.7,6.3,1.4]	40	[7.9,6.3,2]	0.0114
[12.4,3.8,8.6]	42	[12.4,2.9,8.7]	0.0148
	43	[12,3.5,7.6]	0.0141
[3.2,4.3,2.9]	45	[2.3,3.8,2.5]	0.0108
[10.3,3.8,7.1]	46	[9.4,3.2,6.7]	0.0100

Table C-3. Charging zones and time each nodes takes to be fully charged by a PTX for omni-directional antennas

PTX #	Node #	Location (m)	Charge time (h)	PTX #	Node #	Location (m)	Charge time (h)
1	1	[2.5,12.8,7.7]	1099	14	24	[10.4,5.7,1.4]	1388
	2	[2.4,10.4,3.6]	1382		25	[10.4,6.3,3.5]	991
2	3	[3.6,12.8,2.9]	2269	15	26	[8.3,2.2,3.3]	2582
	4	[4.4,11.9,5.5]	878	16	27	[10.4,4,1.18]	2193
3	5	[4.5,8.5,8.2]	2288	17	28	[9.2,4.6,3.9]	2191
4	6	[2.6,11.6,8.2]	1773		29	[3.3,5.1,0.8]	1097
	7	[3.3,11.6,5.3]	1102	18	30	[7.5,5.2,5.5]	1481
5	8	[3.3,11.4,3]	1322		31	[5.8,4.9,0.1]	1039
	9	[3.4,11,7.7]	1436	19	32	[3,5.2,5.1]	1389
6	10	[2.7,9.5,4]	1419		33	[3.6,3.7,7.1]	1104
	11	[2.2,11.9,8.1]	882	20	34	[4.2,4.6,2.1]	947
7	12	[3.1,9.5,0.3]	1545		35	[9.4,2.5,4]	898
	13	[2.4,10.1,4.8]	1190	21	36	[6.7,2.2,5.1]	1679
8	14	[2.1,12.5,6.4]	1177		37	[9.6,6.4,0.5]	903
	15	[9.4,9.6,1.7]	1403	22	38	[4.8,3.3,4.4]	1677
9	16	[10.1,9.5,0]	1270		39	[2.1,4.7,5.8]	1387
	17	[9.5,9.1,2.8]	1162	23	40	[7.8,6.3,2]	1769
10	18	[10.1,12.8,6.3]	1737	24	41	[5.0,2.8,7.5]	1750
	19	[12.4,12.7,5.6]	1216	25	42	[12.4,2.8,8.7]	1739
11	20	[11,12.2,4.9]	2405		43	[12,3.5,7.6]	876
12	21	[8.9,11.8,3.9]	2482	26	44	[6.3,6.2,4.5]	895
13	22	[11,9.3,2.6]	1780		45	[2.3,3.7,2.5]	1225
	23	[9.8,10.1,4.5]	1113	27	46	[9.4,3.2,6.7]	1408

Table C-4. Charging zones and time each nodes takes to be fully charged by a PTX for directional antennas

PTX #	Node #	Location (m)	Charge time (h)	PTX #	Node #	Location (m)	Charge time (h)
1	1	[2.5,12.8,7.7]	200	3	28	[9.2,4.6,3.9]	398
	2	[2.4,10.4,3.6]	253		29	[3.3,5.1,0.8]	198
	3	[3.6,12.8,2.9]	403		30	[7.5,5.2,5.5]	252
	4	[4.4,11.9,5.5]	160		31	[5.8,4.9,0.1]	159
	5	[4.5,8.5,8.2]	397		32	[3,5.2,5.1]	251
	6	[2.6,11.6,8.2]	316		33	[3.6,3.7,7.1]	198
	7	[3.3,11.6,5.3]	200		34	[4.2,4.6,2.1]	158
	8	[3.3,11.4,3]	158		35	[9.4,2.5,4]	163
	9	[3.4,11,7.7]	252		36	[6.7,2.2,5.1]	253
	10	[2.7,9.5,4]	251		37	[9.6,6.4,0.5]	158
	11	[2.2,11.9,8.1]	161		38	[4.8,3.3,4.4]	250
	12	[3.1,9.5,0.3]	250		39	[2.1,4.7,5.8]	250
	13	[2.4,10.1,4.8]	199		40	[7.8,6.3,2]	316
	14	[2.1,12.5,6.4]	200		41	[5.0,2.8,7.5]	315
	15	[9.4,9.6,1.7]	259	4	42	[12.4,2.8,8.7]	315
	16	[10.1,9.5,0]	199		43	[12,3.5,7.6]	159
	17	[9.5,9.1,2.8]	198		44	[6.3,6.2,4.5]	158
2	18	[10.1,12.8,6.3]	316		45	[2.3,3.7,2.5]	198
	19	[12.4,12.7,5.6]	237		46	[9.4,3.2,6.7]	251
	20	[11,12.2,4.9]	409				
	21	[8.9,11.8,3.9]	404				
	22	[11,9.3,2.6]	317				
	23	[9.8,10.1,4.5]	200				
	24	[10.4,5.7,1.4]	253				
	25	[10.4,6.3,3.5]	159				
	26	[8.3,2.2,3.3]	402				
	27	[10.4,4,1.18]	401				

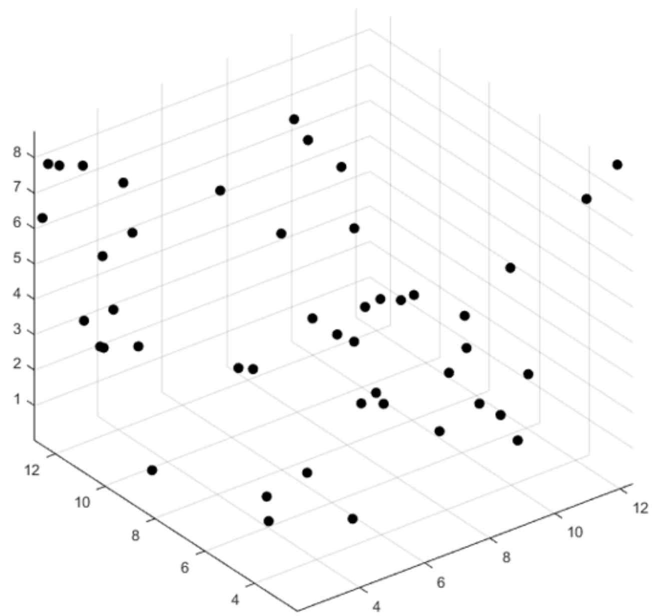


Figure C-1. 3D visualisation of node locations

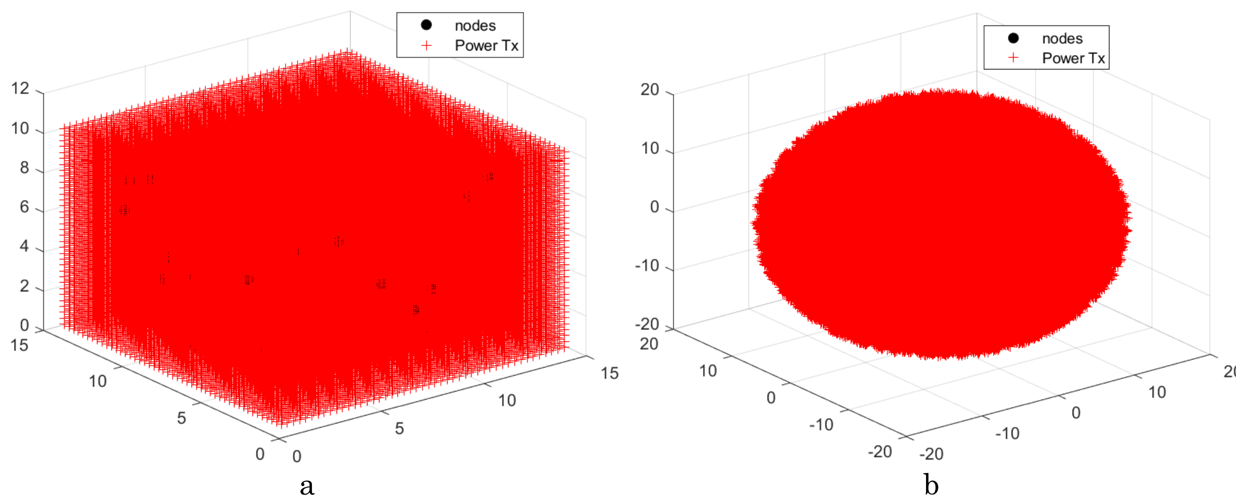


Figure C-2. 3D visualisation of multiple PTX coverage regions generated over the entire plant space for a) omni-directional antennas and b) directional antennas

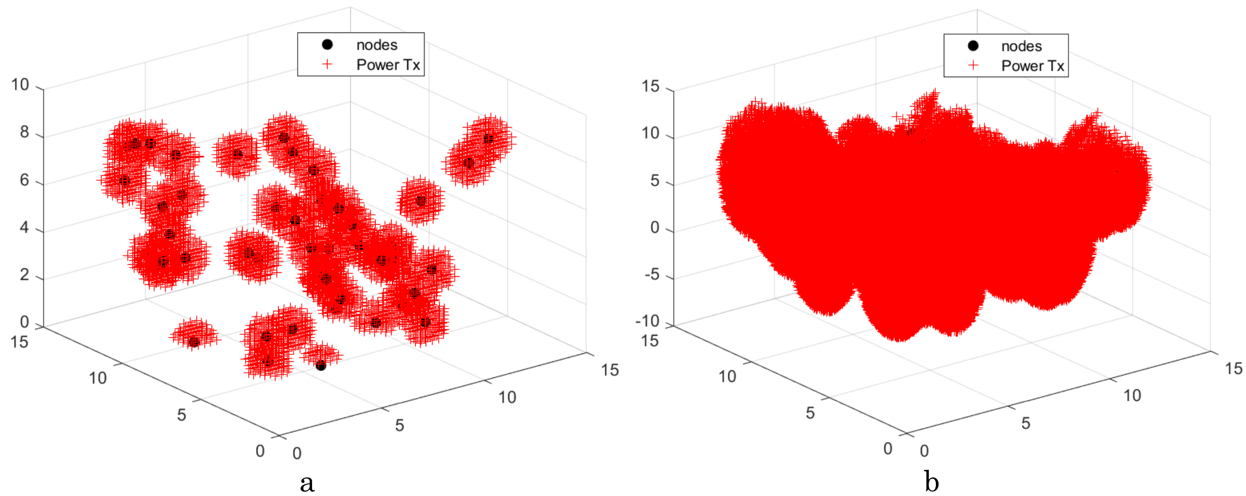


Figure C-3. 3D visualisation of multiple PTX that cover nodes for a) omni-directional antennas and b) directional antennas

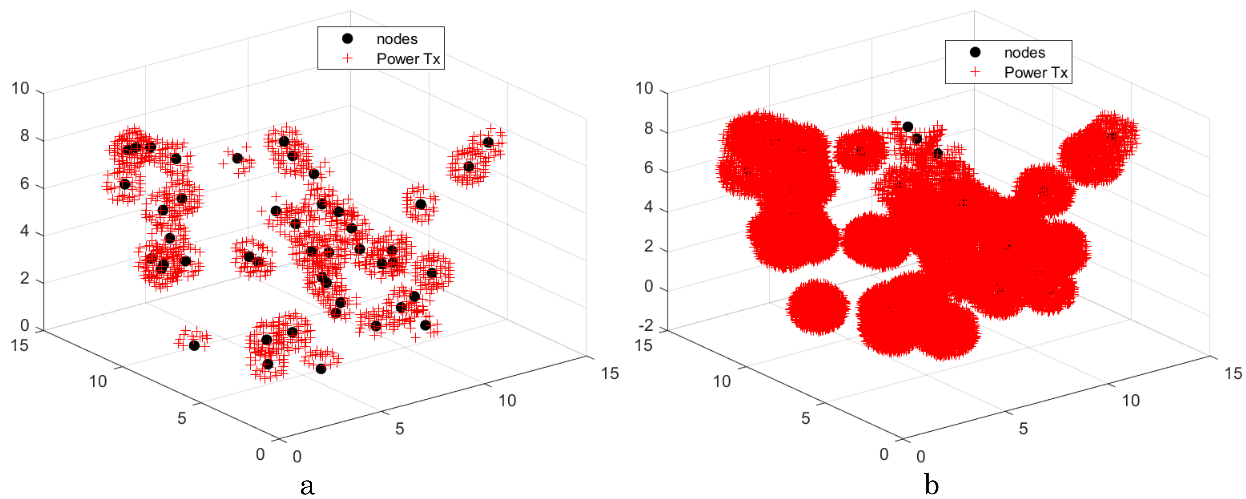


Figure C-4. 3D visualisation of multiple PTX that cover nodes for a) omni-directional antennas and b) directional antennas after applying attenuation

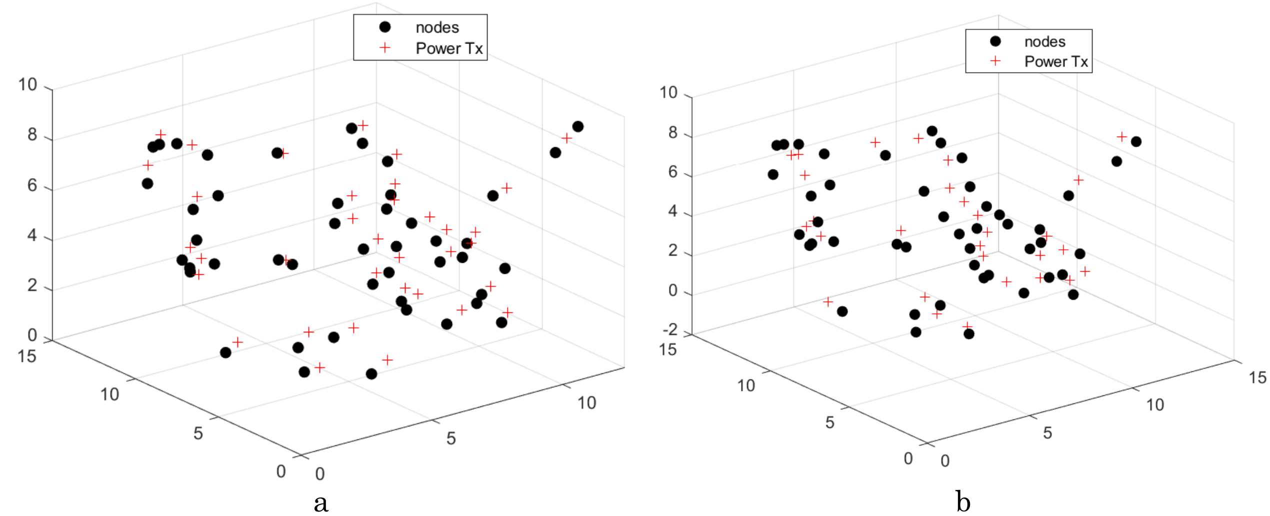


Figure C-5. 3D visualisation of PTX locations relative to the nodes for stationary architecture and a) omni-directional antennas and b) directional antennas