Algorithms for coverage mapping and optimizing beacon placement in a hybrid indoor positioning system

Master's thesis in Algorithms, languages and logic & Software Engineering

Johannes Sjöberg & Anton Johansson

Department of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY AND UNIVERSITY OF GOTHENBURG
Gothenburg, Sweden 2018

Master's thesis 2018

Algorithms for coverage mapping and optimizing beacon placement in a hybrid indoor positioning system

Johannes Sjöberg & Anton Johansson

Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg
Gothenburg, Sweden 2018

Algorithms for coverage mapping and
optimizing beacon placement in a hybrid indoor positioning system

JOHANNES SJÖBERG
ANTON JOHANSSON

Supervisors: Thomas Petig and Elad Schiller, Department of Computer Science and Engineering
Examiner: David Sands, Department of Computer Science and Engineering

Master's Thesis 2018
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Typeset in LaTeX
Gothenburg, Sweden 2018

Algorithms for coverage mapping and optimizing beacon placement in a hybrid indoor positioning system
JOHANNES SJÖBERG
ANTON JOHANSSON
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg

Abstract

This thesis considers a new analytical approach based on modeling indoor positioning as an coverage optimization problem. Over the recent years there has been extensive research in the subject of indoor positioning. Recent research commonly considers promising solutions by deploying new infrastructure, how ever, it does not often make use of already deployed infrastructure. The analytical approach proposed in this thesis considers the indoor environment and make use of meta heuristics to suggest how to extend existing infrastructure to create an reliable indoor positioning system.

A prototype system has been implemented and several experiments have been performed at the expansion of existing Wi-Fi infrastructure with a Bluetooth low energy extension. The results showed that the system produces a reliable and affordable deployment design which uses very few reference nodes. How ever, this approach considers static environmental models and adaptors should be aware that this is best suited for indoor environments which are not subject to major refurbishments or renovations.

Keywords: indoor, positioning, bluetooth, wifi, simulated annealing, deployment, optimization

Acknowledgements

We would like to express gratitude to our supervisor Thomas Petig for the initial idea and for supporting us with his technical and mathematical expertise through this thesis. Furthermore, we would like to thank Elad Schiller for his support and help in writing this thesis. Also, we would like to thank Peter Lindh and Ann-Sofie Rolandsdotter for their support in this thesis. We would like to thank Gustav Ehrenborg and Herman Fransson for the valuable discussions and exchange of knowledge in this subject. Lastly, we would like to thank our parents, Lee Nordevald Sjöberg & Lars Erik Sjöberg and Maria & Åke Johansson who has supported us through this thesis. Thank you all.

Anton Johansson & Johannes Sjöberg, Gothenburg, June 2017

Contents

List of Figures

List of Tables

1

Introduction

Today a large part of the population in the world is equipped with devices (such as smart phones, tablets, laptops etc.) that offer various wireless connection standards. This opens up many new opportunities to provide services that are based on the location of individuals, so-called location based services (LBS).

The Global Positioning System (GPS) [15] have been a widely used service for positioning and tracking by civilians in outdoor environments. However, GPS does not provide good accuracy in indoor environments due to multipath fading.

Over the recent years there has been extensive research in the subject of indoor positioning. The development and deployment of indoor positioning systems (IPS) has increased rapidly, with major companies like Apple Inc. and Google Inc. investing in IPS technologies. An IPS can offer a wide range of services such as indoor navigation, targeted marketing and location related information. These services are useful for existing markets, e.g., emergency services, conferences and shopping malls to name a few.

In order to be able to reach a broad commercial market, one must utilize the opportunities existing in commercial-off-the-shelf devices. It's therefore crucial to use the technologies present in these devices, such as Bluetooth, Wi-Fi and RFID to enable the creation of valuable location based services.

1.1 Problem description

Positioning in an indoor environment is often achieved by using radio frequency reference nodes. By using the signal strength of the reference nodes it's possible to establish the position of a device with positioning methods such as trilateration [27], fingerprinting [22] and extended Kalman filtering [9]. There exist several technologies that can be used for this. Including; Wi-Fi [38], Bluetooth [5], Radio Frequency Identification [30] and Ultra Wideband [17] to name a few.

Indoor environments are usually confined spaces consisting of obstructing obstacles, such as walls, doors, people etc., which distorts and scatters the signal propagation of the reference nodes [33]. As these effects may differ over time, doors open and closes, unexpected noise and other environment changes, the propagation of the signals is subject to unreliability.

Within the indoor environment, we refer to *relevant points* as confined spaces where positioning must be achievable. We refer to an IPS as *reliable* when at all relevant points, sufficient information from the reference nodes is receivable in a given time frame, subject to integration criteria. Example of such an integration criterion: The range calculations of radio frequency signals have a logarithmic relationship to path loss. Thus, small fluctuations in weak signals will result in significant positioning errors. Introducing a signal strength threshold criteria at all relevant points will therefore reduce the positioning errors of an IPS.

To encapsulate the requirements of a reliable IPS, we have constructed the α-unique reference node constraint;

> **Constraint 1** (α*-unique reference node constraint*)**.** *Given an integer α, an indoor environment* $\mathbf{E}$*, a closed set* $\mathbf{S}$ *of relevant points within* $\mathbf{E}$*, at every point in* $\mathbf{S}$ *the integration criteria must be fulfilled for at least α reference nodes.*

Wi-Fi is a communication standard which is widely used and deployed in many indoor facilities such as office buildings, conference halls, airports and train stations [7]. Networks administrators that have deployed Wi-Fi base stations did not have the reliability issues related to indoor positioning in mind. Due to this, an IPS using only existing Wi-Fi infrastructure often performs poorly [8]. Installing additional or moving Wi-Fi infrastructure for the single purpose of IPS is expensive, as deployment includes time-consuming installation, high-cost equipment and an active power supply for each reference node.

An alternative to Wi-Fi is Bluetooth Low Energy (BLE). BLE is a widely available technology that offers low-cost battery-powered equipment. BLE outputs a weaker signal strength than Wi-Fi, thus greater consideration has to be taken to where the BLE reference nodes are to be placed. A poorly positioned reference node can be rendered useless because it does not provide any significant signal propagation.

Wi-Fi technology has been in use the last 15 years, and the investments in infrastructure and deployment has already been made. Therefore, we offer an alternative solution by extending existing Wi-Fi infrastructure with BLE reference nodes. To reduce cost of the BLE extension, we propose that the amount of new reference nodes should be kept minimal.

Our approach. To fulfill a given α-*unique reference node constraint* we extend a existing Wi-Fi infrastructure with a number β of BLE reference nodes. We are looking for the minimum β that satisfy the constraint of the β-BLE infrastructure extension;

> **Constraint 2** (β*-BLE infrastructure extension*)**.** *Given an integer* β*, an indoor environment* $\mathbf{E}$*, and a non-modifiable Wi-Fi infrastructure, a placement of* β *BLE reference nodes must exist such that the Wi-Fi infrastructure together with the BLE reference nodes fulfill the constraint 1.*

1.2 Related Work

Baniukevic et al. [8] propose a hybrid IPS motivated by their observations that using only existing Wi-Fi infrastructure often result in poor positioning performance. Their approach uses Bluetooth classic reference nodes as guard keepers, by placing them at e.g., doors and staircases to detect cross-border movements. Compared to their approach, we combine Bluetooth and Wi-Fi propagation concurrently to utilize as much of the signal propagation as possible. Baniukevic et al. [8] states that it's relevant to further study the setup of Bluetooth reference nodes in addition to existing Wi-Fi infrastructure. Specifically, the optimal number of Bluetooth reference nodes and where they should be deployed to maximize the positioning performance. Aparicio et al. [5] use a fusion of Wi-Fi fingerprinting and Bluetooth cell identification. Their use of Bluetooth classic reference nodes are limited to determine an approximate area to narrow the search space of the Wi-Fi fingerprinting to that area. They present no approach to deploy the reference nodes to ensure sufficient signal propagation throughout the indoor environment. Further, we use BLE reference nodes compared to [8] [5] which use Bluetooth Classic reference nodes.

Ficco et al. [16] approaches deployment of reference nodes as a computational problem. They propose a way to compute the best deployment schema for Wi-Fi reference nodes with stochastic algorithms. Their approach uses a analytical signal propagation model to build a radio map and a multi-objective genetic algorithm to find the best placement of Wi-Fi references nodes. However, their approach does not take any existing Wi-Fi coverage into account. Further, they are approaching this problem using only Wi-Fi, compared to our approach which make use of BLE technology. Aomumpai et al. [4] have a similar approach to the deployment of Wi-Fi reference nodes. They propose two concerting algorithms using Binary Integer Linear Programming to solve this problem. The first algorithm determines the minimum number of reference nodes needed to fulfill their reference nodes in range constraint. The second algorithms maximizes the RSS value at selected test points to achieve the highest possible positioning performance. Their approach, with solving the problem with two independent algorithms contrasts our approach, where the amount of reference nodes and maximum signal coverage is defined as as a single problem, which is solved by one algorithm.

Current literature lacks methods to expand existing Wi-Fi infrastructure to support indoor positioning. To the best of our knowledge, no scientific publications have analyzed a hybrid system that focus on the salvage and utilization of already existing infrastructure using our approach.

1.3 Our contribution

We design and develop a prototype system as well as a new analytical approach for indoor positioning using Wi-Fi and Bluetooth. Our solution allow already deployed Wi-Fi infrastructure to better support indoor positioning by using an affordable BLE extension.

Our prototype system offers a sophisticated way to conduct a survey of the current Wi-Fi infrastructure and create a radio map of the Wi-Fi signal coverage. By using this radio map, the system can suggest how to extend the Wi-Fi infrastructure with a BLE extension to achieve reliable indoor positioning.

Our analytical approach is based on modeling indoor positioning as a coverage optimization problem. We simulate the signal propagation of BLE reference nodes with the Wi-Fi radio map by using the multi-wall-classic signal propagation model. We apply meta heuristics to find a suitably low amount of BLE reference nodes and their respective placement. The meta heuristics we apply has been proven to converge [37], providing approximate minimized solutions fulfilling our integration criteria. The reference node placement is constrained by three specifiable integration criteria; signal strength threshold, least amount of reference nodes in range and non-allowed reference node placements.

We evaluate the deployment design by conducting a second survey of the signal propagation from both Wi-Fi and BLE reference nodes. Our approach has shown to reduce costs and minimize the hardware resulting in an affordable indoor positioning system with viable environmental sustainability. The system is also capable of designing a deployment schema of BLE reference nodes alone.

1.4 Ethics

This thesis involves theory and use of positioning systems, which tracks the movements and positions of electronic devices which may be in constant possession of a single individual. Such systems could thus potentially be used for tracking the whereabouts and movements of individuals, which could be viewed as an invasion of privacy, even without knowing any personal information about the them.

As this thesis is focused on developing deployment techniques of the systems, rather than the systems themselves, it violates no privacy of individuals. During the testing phase of the thesis, we deploy a positioning system with the following limitations;

I Requires an active consent of the user to be enable tracking

II Saves no personal data except for a MAC address

III Only testing equipment are to be tracked

Thus, we see no ethical issues being violated during and/or after this thesis has been conducted.

Part I

Background

2

Reference node technology

This chapter provides background and technical details of the technologies present in the reference nodes used in this thesis. The reference node technology is one of the cornerstones in indoor positioning systems, as it acts as the link from the actuality to the digital representation of the position. While radio-wave based technologies are often used in indoor positioning systems, other solutions has been developed, such as ultrasonic [29] and infrared [3]. Two radio-wave based technologies, Wi-Fi and Bluetooth, are presented in this chapter, and subsequently used in the system we developed.

2.1 Wi-Fi

The IEEE 802.11 standard, commonly denoted Wi-Fi, is a widely adopted wireless communication standard maintained by the Wi-Fi alliance. Wi-Fi originated as a wireless radio wave based alternative to wired ethernet, and operates often in the unlicensed industrial, scientific and medical (ISM) 2.4 GHz frequency band, but also the 3.65 and 5 and 60 GHz frequency band [21]. Generally, connection is made by creating wireless local area networks (WLAN) via 1 or more access points (AP) placed in the indoor environment.

2.1.1 Wi-Fi standards

The first 802.11 standard(802.11-1997), released in 1997, received feedback that products did not meet compatibility needs expected by customers. However, the formidable market success and the perceived drawbacks of 802.11-1997 has provided a basis for many extensions and improvements. There are now a collection of various Wi-Fi versions, designed for different purposes, such as increased throughput and complying with region specific laws and regulations.

802.11 Protocol	Frequency (GHz)	Stream Rate (Mbit/s)	Indoor range(m)
a	5	54	35
b	2.4	11	35
g	2.4	54	38
n	2.4 or 5	600	70
ac	5	> 1000	35

Table 2.1: Common Wi-Fi standards [21]

2.1.2 Distance calculations using RSSI

In 802.11, the RSSI value is broadcast from an AP in an easy to obtain form [28]. In [6], IEEE specifies the RSSI values as an unsigned 8-bit integer value that ranges from 0 to `RSSI_Max`, a specified maximum, where `RSSI_Max` is the strongest signal, and 0 the weakest. As the interpretation of RSSI is vendor specific, the interpretation of received RSSI-values has to be adjusted to the hardware that is being used.

2.2 Bluetooth

2.2.1 Bluetooth classic

Bluetooth classic is a communication technology that operates in the ISM 2400MHz - 2483MHz frequency band, divided into 79 channels, each 1 MHz wide [2] designed for short ranges. It has three standard power classes [1] defining the ranges;

Class 3 which has a range of $\sim$ 1m and a power of 1mW

Class 2 which has a range of $\sim$ 10m and a power of 2.5mW

Class 1 which has a range of $\sim$ 100m and a power of 100mW

where class 2 being the common implementation in many smart devices. Application areas for Bluetooth include streaming media content such as a wireless head-set or sending data from peripherals, such as a stethoscope.

As Bluetooth overlaps the frequency band that other wireless technologies use, such as Wi-Fi, Bluetooth utilizes frequency hopping in order to minimize interference. Such an hopping technique Adaptive Frequency Hopping (AFH), in which the Bluetooth hopping scheme is designed to avoid the channels used by the Wi-Fi connections, the trade-off being that the throughput of Bluetooth is reduced.

When two devices communicates by using Bluetooth classic, a master-slave connection is established in a process called pairing. During pairing the slave device listens for a so called inquiry from a master device. When a inquiry has been received, pairing –and subsequently data transfer– can be established. It is first during the pairing phase that communication settings, including RSSI values, are traded.

2.2.2 Bluetooth low energy

The Bluetooth Low Energy (BLE) is a new communication technology based of the Bluetooth classic protocol. BLE is designed with low energy consumption and small data transactions in mind, making it suitable for areas such as control- and monitoring applications. As BLE and Bluetooth classic has been designed with different applications in mind, BLE has a new protocol stack and is thus not backwards compatible with Bluetooth classic.

BLE operates – like Bluetooth classic – in the ISM 2400 - 2483Mhz frequency band, but has 40 channels, each 2MHz wide. BLE is not designed for streaming information as Bluetooth classic is, but rather providing short bursts of information and several other features which are making it more suitable for an IPS than classic Bluetooth. It has a reduced set-up time at ~ 3 ms and a throughput of 200kbit/s, both lower than the standard bluetooth [1]. In contrast to Bluetooth classic BLE will deliver RSSI values without having established a connection beforehand. The RSSI of BLE set from -127 up to 20dBm.

2.2.2.1 BLE broadcast

A new feature introduced in BLE is the advertising mode, which is well suited for IPSs. The advertising mode is a one-way broadcasting, where the beacon broadcasts data periodically in a set interval from 100ms up to 10.65s in three of the channels specified as advertisement channels (channels 37, 38 and 39). The devices that listens to the advertisement channels is called scanners. During this mode, no connection has to be established between the broadcaster and the scanners, and this type of one-way communication it is possible for a single BLE-device to broadcast a message to a very large amount of scanners at the same time.

2.2.2.2 Estimote beacon

The BLE reference nodes used in this thesis are called Estimote Beacons. They are manufactuered by Estimote Inc and uses a 32-bit ARM® Cortex M0 CPU which is accompanied by a accelerometer and a temperature sensor. At maximum power setting, the Estimote Beacon broadcasts a signal of -60 dBm at the range of 1m, and has a maximum range of 70 m during optimal conditions.

3

Positioning techniques

When designing an IPS, the use of positioning techniques is crucial. A positioning technique is the translation from reference node input to agent real world position, and this could be accomplished with several different approaches. This section presents three positioning techniques that are often used in IPS. The requirements of these positioning techniques relates to the deployment of the reference nodes in a IPS, and therefor to the requirements of our system.

3.1 Proximity based positioning

Proximity based positioning (PBP) is a simple and coarse positioning method which only require 1 reference node in range of an agent to determine a position. The position of an agent is determined by assuming the agent is located at the reference node from which the strongest signal is recieved [19]. This method is commonly used for proximity services, due to the unreliable positioning accuracy [19]. An example of areas where PBP is used is wireless car keys. They allow drivers carrying the keys to unlock and start the car by simply being in the close vicinity of it.

3.2 Trilateration

Trilateration is a technique that use the signal loss with geometric calculations to determine a position of an agent. To achive this, it requires 3 reference nodes in range of the agent their known position in order to determine a position. By first measuring the signal strength of the signals from the reference node to the agent, one proceeds to modelling a corresponding circle for each reference node, and setting the radii of the circles equal to the measured value. By calculating the point of intersection of the circles, and subsequently the position of the agent, a position can be determined. A visualization of trilateration can be seen in figure 3.1. In the figure, the black points represents the reference nodes and the dotted lines the distance to the agent, which is marked with a blue dot. As can be seen, the distances is the radii of circles created from the coverage, and they intersect at the position of the agent.

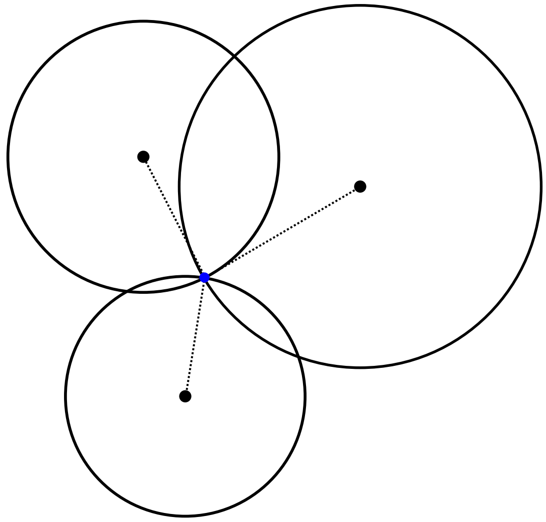

Figure 3.1: A visual example of trilateration.

3.3 Fingerprinting

Fingerprinting is a positioning technique that uses previously gathered data to approximate a position. To achive this, it is split up into two phases; the offline phase and the online phase. The area in which positioning will take place has to be divided into a rectangular grid of cells.

In the offline phase, the RSSI values is gathered and stored in a vector for each cell, creating so called fingerprints. Each fingerprint represents the RSSI's that are to be expected from each reference node in this particular cell, and is used during the positioning.

During the next phase –the online phase–, an agent sample the RSSI readings at the current position. The readings are constructed to a vector, and then compared to the fingerprints stored in the database from the offline phase by e.g cosine similarity. The fingerprint that is most similar to the one given by the agent has the highest probability to be the agents position. [22]

The grid size, or granularity, will have a large impact on the results of the positioning, as it determines the accuracy of the positioning. A large grid size will result less accuracy. A fine grid will increase the accuracy, with the trade off that it will create a large overhead during the offline phase. Also; a very high granularity is not guaranteed to provide better precision, as the signal variances will be low, thus making close cells hard to distinguish from one another resulting in missclassifications [22]. Thus, too high granularity increases the overhead, while yielding no improvement in precision.

3.4 Further enhancements

In addition to reading reference node data, additional methods can be utilized to increase the performance of IPS. Examples include adding other technologies such as magnetometers and gyroscopes.

One problem that reduces positioning accuracy is the human blocking effect. A person holding the positioning device will most often have negative impact on positioning accuracy. This due to a majority of the human body is composed of water, and thus absorb signals traveling through the person holding the device. One method used to counter this effect is by combining fingerprinting with a magnetometer to predict which signal angles have greater positioning reliability [24].

Other approaches to improve indoor positioning is adding inertial sensors to keep track of the users footsteps [39]. Using this method gives an additional layer of information regarding how far the agent has moved. Combined with map-matching, this can provide a good prior to position estimation, for example to avoid guessing that a person has walked through a wall.

4

Indoor signal propagation

In order to be able to simulate the signal coverage, the path-loss of the signal propagation must be accounted for. The underlying characteristics of an indoor environment, such as the existence of obstacles (e.g., walls, doors, windows and furniture), as well as existing noise sources (e.g., human bodies and other transmitters) makes the signal propagation suffer from multipath fading due to to reflection, absorption, scattering, refraction and diffraction [33]. This makes the propagation behavior in indoor environments a harder challenge and a more tedious process than compared to open or outdoor environments.

4.1 Propagation models

Tam et al. [35] states there are two main groups of indoor propagation models to be adopted:

1. **Statistical models** such as One-slope [14] and Multi-Wall-Models [26] are semi-empirical models based on a exponential relationship between the distance and the path loss. The exponential relationship and the attention of obstacles, such as walls and floors, have to be empirically measured on the specific site.

2. **Site specific models** such as Ray-Tracing [35] are based on electromagnetic-wave propagation theory and require great detail of the indoor environment to obtain an accurate prediction of the propagation. Ray tracing is based on the concept that rays and high frequency microwaves behaves in a similar manner.

The statistical models only consider the direct rays of the signal, which provides a lower propagation accuracy compared to the site specific models, which will take reflection and diffraction into hand. Site specific models requires a greater detailed image of the indoor environment and is more vulnerable to changes as the prediction is corrupted at the rearrangement of large objects in the environment. Also, site specific models, such as ray tracing, will have a significantly larger computational complexity. Due to the very large amount of reference node placements and coverage calculations in our simulation, the computational overhead for calculating the propagation has to be kept small, despite resulting in less accuracy. However, our simulation software is not dependent of the propagation model, which can be easily swapped.

4.2 Free-space path loss

Signal path loss analysis is fundamentally determined by the free-space path loss (FSPL). Assuming the antennas are isotropic, the FSPL can be modeled as in Eq. 4.1.

$$\mathrm{FSPL} = \left(\frac{4\pi d}{\lambda}\right)^2 \tag{4.1}$$

$$\mathrm{FSPL_{dB}} = 10 \cdot \log\left(\frac{4\pi d}{\lambda}\right)^2 \tag{4.2}$$

$$= 10 \cdot \log\left(\frac{4\pi}{\lambda}\right)^2 + 10 \cdot 2\log(d) \tag{4.3}$$

where $d =$ Distance between the receiver and transmitter
$\lambda =$ Wavelength of the signal

Eq. 4.3 gives the FSPL in dBm, a common unit when dealing with signal loss.

A drawback with FSPL is that it does not account for any multi fading, and is therefore performing poorly when predicting propagation in a indoor environment.

4.3 One-slope model

The one-slope model [14] assume a linear dependence between the path loss and the logarithmic distance, where the free space loss term (Eq. 4.3) is modified and set to L_0.

$$L_{\mathrm{one-slope}} = L_0 + 10n\log(d) \tag{4.4}$$

$$L_{\mathrm{one-slope-fspl}} = L_0 + 20\log(d) \tag{4.5}$$

where $L_0 =$ path loss at the distance of 1m
$n =$ power decay index
$d =$ distance between transmitter and receiver

L_0 is determined by measuring the path loss at the distance of 1m. Further, this model is easy to use as the only input parameter is the distance d. The slope factor

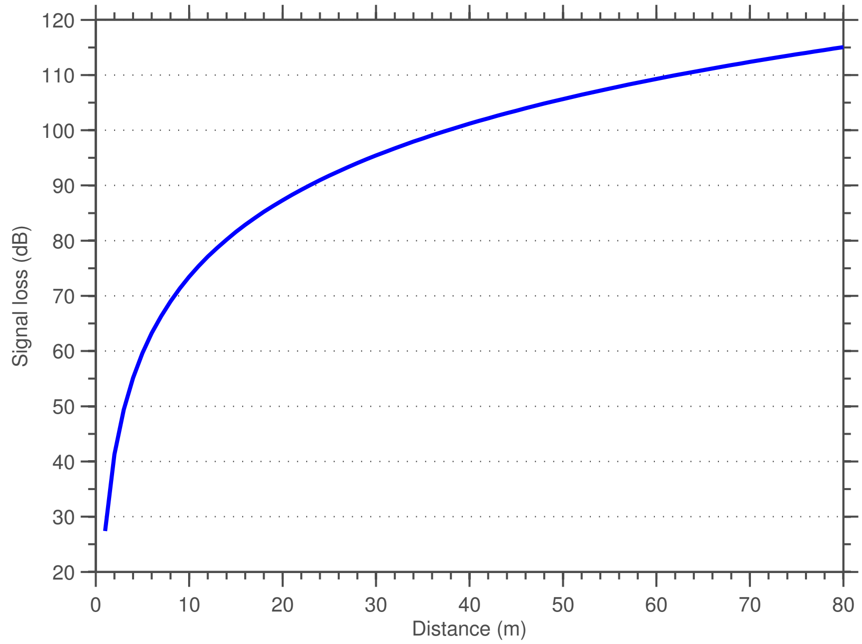

Figure 4.1: A plot of the path loss using standard FSPL calculation

(power decay index) n is set to 2 for free space propagation and otherwise obtained by empirical tests in the set indoor context as in Eq. 4.6.

The variable n is solved empirically by using Eq. 4.6, where the RSSI is measured line-of-sight (LOS) between the transmitter and the receiver at different distances d_i. In the case where the obtained values of n differ, n is set to the average value of all obtained n as in Eq. 4.7.

$$n = \left(\frac{\mathrm{RSSI} - L_0}{10 \log_{10} d} \right) \tag{4.6}$$

$$\bar{n} = \frac{1}{N} \sum_{i=1}^{N} n_i, \;\; N = \text{the number of obtained } n \tag{4.7}$$

The aforementioned way of determining the power decay index n provides an accurate propagation predicition when the context is limited to one room. In order to account for a larger context which contains obstacles such as walls and floors, different models have tried to vary the power decay index with the distance [20]. This is a time consuming process as it require a lot of tests in the set environment and has shown to produce less accurate results as the Multi-Wall models [20].

4.4 Multi-Wall models

Multi-wall (MW) models are more refined versions of the one-slope model (Eq. 4.4) and FSPL (Eq. 4.3) for indoor environments [11, 34, 26]. These models introduce the addition of an attenuation term to represent the signal loss caused by walls and floors in a direct path between the transmitter and the receiver.

$$L_{\mathrm{MWC}}(d) = L_{\mathrm{one-slope-fspl}} + L_c + \sum_{i=1}^{I} k_{wi} L_i + k_f L_f (\mathrm{dB}) \tag{4.8}$$

$$L_{\mathrm{MWE}}(d) = L_{\mathrm{one-slope}} + L_c + \sum_{i=1}^{I} k_{wi} L_i + k_f L_f (\mathrm{dB}) \tag{4.9}$$

where $L_{\mathrm{one-slope-fspl}} = L_0 + 20\log(d)$
$L_{\mathrm{one-slope}} = L_0 + 10n\log(d)$
L_0 = path loss at the distance of 1m
n = power decay index
d = distance between transmitter and receiver
L_c = constant loss
k_{wi} = number of penetrated walls of type i
L_i = loss of wall type i
k_f = number of penetrated floors
L_f = loss between adjacent floors
I = number of wall types

The L_{c} term in Eq. (4.8) and Eq. (4.9) is the result of wall losses determined from measurement using multiple linear regression, and is normally close to 0 [14]. The sole difference between Multi-wall-classic (MWC) and Multi-wall-enhanced (MWE) are the terms L_{fspl} and $L_{\mathrm{one-slope}}$, which in the later case lets one to adjust the power decay index n to the set environment.

5

Coverage optimization

This chapter will cover the algorithmic approach to find a reference node placement given a floor layout of an indoor area to achieve the highest possible positioning accuracy. It will start with describing the problem at hand, and then continue to relate it to a known optimization problem. The final part of the chapter will present several solutions to the reference node placement problem using various methods which are derived directly or related to the known problems.

5.1 Finding a deployment of the reference nodes

In order to develop a system to find the least amount of reference nodes such that constraint 1 and constraint 2 are satisfied, a mathematical model of the problem has to be defined.

5.1.1 Preliminaries

The confining walls of the indoor environment is modeled as a 2 dimensional complex polygon. The complex properties that an indoor environment polygon may posses is "holes" inside the polygon e.g., an elevator shaft. Further, walls inside the confining polygon is modeled as lines.

As stated in constraint 1, it should be satisfied in a *closed* set of points, in this case, a discrete set of sampling points. This set of points is modeled as a grid superimposed over the indoor environment. The grid G consist of $m \times n$ rectangular cells, each cell $c_{i,j} \in G$ representing a point. Reference points are generated on available walls with a fixed granularity.

5.1.2 Set cover problem

The SET COVER PROBLEM is a classic NP-complete problem within the field of combinatorics [23]. Given a universe of elements U, a set S of subsets of U where the union of all subsets in S produces U, find a minimized subset C of S such that $\bigcup C = U$, i.e. all elements in U are covered by C.

The SET COVER PROBLEM formulated as an Integer Linear Program (ILP):
Given an input of universe U of e elements, $S_1, \ldots, S_n$ of subsets of U.

$$\begin{aligned} & \min \sum_{i=1}^{n} x_i \\ \text{subject to} \quad & \sum_{i:e\in S_i} x_i \geq 1, \ \forall e \in U \\ & \quad x_i \in \{0,1\}, \ \forall i \in \{1,\ldots,n\} \end{aligned} \tag{5.1}$$

The variable x_i is assigned to 1 if set S_i is selected, otherwise assigned to 0.

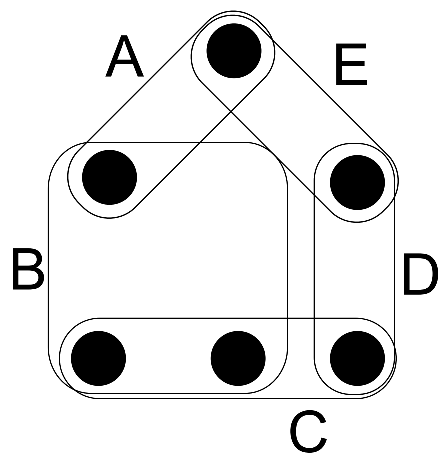

Figure 5.1: A visual example of the set cover problem. An optimal solution is $\{A, C, E\}$.

5.1.3 The set cover problem in relation to our problem

We use the definition of 5.1 to translate our problem into the SET COVER PROBLEM; model the indoor environment as universe U, each sample point within the environment as an element e and each reference node placement as a set S_i. Let the set of each reference node consist of the sample points in the indoor environment that the reference node cover. Then minimize the amount of sets needed, given that each element in the universe is covered by at least 3 sets (instead of 1, as in 5.1). This differs from the definition of the set cover problem as the union of the selected subsets of $\bigcup C = U$ might not satisfy our problem. We must in our implementation keep track of each covered element e to ensure it is covered by at least three sets in C.

The set cover problem ILP transformed to our problem definition:
Given an input of universe U of e elements, $S_1, \ldots, S_n$ of subsets of U.

$$\begin{aligned} & \min \sum_{i=1}^{n} x_i \\ \text{subject to} \quad & \sum_{i: e \in S_i} x_i \geq 3, \ \forall e \in U \\ & x_i \in \{0, 1\}, \ \forall i \in \{1, \ldots, n\} \end{aligned} \tag{5.2}$$

The variable x_i is assigned to 1 if set S_i is selected, otherwise assigned to 0.

5.1.4 Solving the set cover problem

As the set cover problem lies in NP-Complete it is unfeasable to find a solution for larger search spaces [23]. The search space for our problem – with magnitude in the hundreds – makes it impractical to do exhaustive searches. Thus approximation approaches such as heuristics has to be implemented to reach feasable solutions.

5.2 Heuristics

Heuristics are stochastic search algorithms that converge to the global optima. They are commonly used to find solutions for problems in NP in a reasonable time frame, with the trade-off that it is not guaranteed to be optimal but rather *reasonably good*, depending how early the search is terminated. With *reasonably good* meaning that the result is sufficiently close to global optima for it to be useful in the application of the result [37]. Two known heuristics will be described, and explained how they are implemented to solve for the in both expected results and performance.

5.2.1 Simulated Annealing

Simulated annealing (SA) is a probabilistic local search algorithm for discrete optimization problems, which is able to escape local optima [18] [25]. It is a popular heuristic due to the effectiveness and the relative ease of implementation. The name derives from the real-world annealing process; a slow, controlled cooling of a heated solid lowers the lattice energy state and thus increases the structural integrity [18].

Description The implementation of simulated annealing is a iterative meta heuristic, which runs until a predefined stopping condition is met (e.g. a fixed set of iterations).

First, an initial solution s and temperature T is set during initialization, where the temperature is a predefined value and the solution is most commonly a random solution. At the start of each iteration, the temperature is lowered according to a set cooling schedule. Next, a random change is performed on the current solution s to produce a new solution $\hat{s}$, and if $\hat{s} < s$, $\hat{s}$ is accepted with a probability equal to 1 i.e., a better solution is always accepted. If $\hat{s} \geq s$, i.e. $\hat{s}$ is a worse solution, the Boltzmann factor

$$F = \exp \frac{s - \hat{s}}{T} \tag{5.3}$$

is calculated, where T is the current temperature. Then the possible hill climbing (selecting a worse solution) is decided by

$$s = \begin{cases} \hat{s}, & \text{if } F < r \\ s, & \text{otherwise} \end{cases} \tag{5.4}$$
$$r = \text{random variable} \in (0, 1)$$

The Eq. 5.4 of allowing a worse solution to be accepted enables simulated annealing to escape from a local optima. In early iterations the risk of reaching a local optima is greater, which is countered by the high temperatures which enables the algorithm to "climb" out of the local optima. As the iterations continue and the solutions begin to converge towards the optima, the risk of getting stuck in a poor local optima is lower. Likewise, the temperature lowers with the iterations, and so does the chance of accepting a worse solution.

Initial temperature The cooling schedule and initial temperature are both problem specific, and there exist some research in regard how to set them. Methods include; i) make an educated guess, ii) by trial and error [13] or iii) calculating an optimal initial temperature [10].

Cooling schedule Regarding cooling schedule, several approaches exist. A common schedule (which is used in this thesis) is the *proportional cooling schedule* [31];

$$T_{\text{new}} = \alpha \cdot T, \quad 0 < \alpha < 1 \tag{5.5}$$

and α is chosen according to

$$\alpha = \left(\frac{T_M}{T_{\text{init}}}\right)^{\frac{1}{M-1}} \tag{5.6}$$

$$\tag{5.7}$$

where T_M is the final temperature,
T_{init} is the initial temperature,
M is the total number of temperature changes

Stopping criteria There exist several ways to define the stopping critera. One

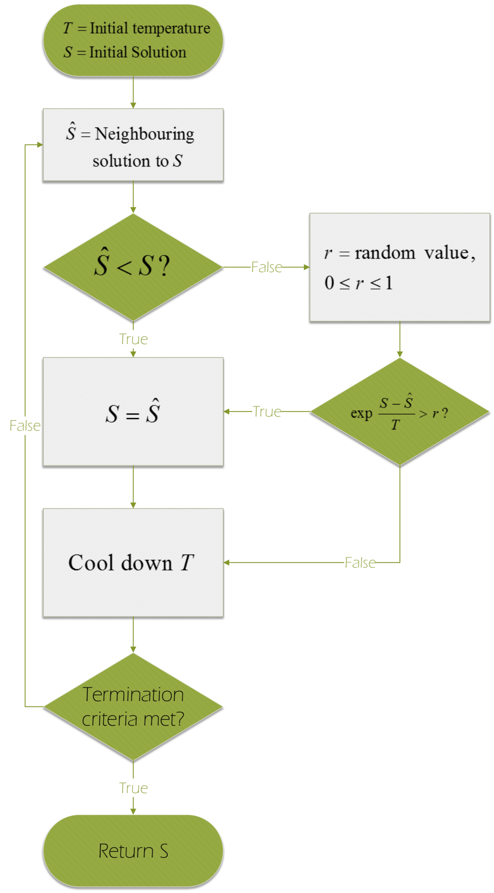

Figure 5.2: Flow chart of simulated annealing

often used principle is stopping the execution after a fixed set of iterations, which could also be defined as reaching a defined temperature or setting a maximum of neighbouring solutions to be generated, such as in [12]. Another common approach to the stopping criteria is to terminate the execution when no better result has been observed after a set number of iterations [32, 25].

5.2.2 Genetic algorithms

The aim of genetic algorithms (GA) is to mimic the natural selection process in nature. In natural selection, individuals compete with each other for the chance of reproduction, where the fittest individual has the highest probability to reproduce. When reproducing, the genetic code of the individuals are merged and produces an offspring that possesses traits of both parents. During evolution, individuals may have a mutation in their genetic code, which alters their fitness in regard to the other individuals. This altercation can either increase or decrease the fitness of the indivdual. GAs work in a similar fashion, with a population of *individuals* that mutates, breed and spawns new generations as the iterations carries on [36];

The population The population is a set of individuals; each constructed of a genetic code, called *chromosome*. The chromosomes are most often implemented as an array of binary values, or *genes*. Via encoding and decoding schemes, a solution to a function can be written as a chromosome and vice versa.

Initialization When creating the population, the population size has to be set. A larger population increases the chances for good results, as there will be more genetic material. The trade-off is running time. A common initialization of the individuals is a random assignment of the genes in each chromosome [36].

Mutation of the individuals The first step of each generation is to mutate the individuals. This is done by iterating through the chromosomes of all individuals, and given a certain probability the chromosome may mutate, i.e. a 0 becomes a 1 and vice versa. This probability, commonly refered to as *mutation rate*, should be set to a value that the expected outcome is 1 mutation per chromosome for each iteration [36].

Selection Each individual is evaluated based on their fitness and "pitted" against each other via a selection scheme. A common selection scheme is tournament selection [36], in which first a random subset of the population are selected and decoded, then sorted in non-increasing order of fitness. A random value is generated, and if it is less than a set *selection rate*, the best individual is selected and copied into the new generation. If not, a new random value is generated, and the same procedure is carried out on the second to best individual and so on. These steps are repeated until a new generation as large as the *selection size* is created.

Crossover In the crossover phase, the individuals in the new generation have their genetic code crossed with each other, i.e. they mate and produce an offspring. This is usually done by randomly selecting two individuals, cutting their chromosomes in a random place, and switching the parts with each other. This is done until a new generation, consisting entirely of offsprings, has been created.

These steps –Mutation, selection, crossover– are constantly repeated until a stopping condition is met, usually a fixed amount of generations.

Part II

Results

6

Implementation Design

We propose a model of a prototype system for designing the deployment of a BLE infrastructure extension. To reduce the amount of hardware deployed, we designed the prototype system to make use of existing signal coverage from Wi-Fi nodes in the indoor environment. Existing signal coverage is gathered using the signal strength survey application (section 6.2). The gathered signal data is then sent to the deployment planning application (section 6.1), which use it together with signal simulations and heuristics to produce a deployment plan.

6.1 The Deployment Planning Application

The deployment planning application (DPA) is a part of the prototype system we developed for the deployment planning of reference nodes. The application proposes how many reference nodes are needed and where they shall be placed, along with a graphical representation of the deployment. The DPA consists of four main parts: (1) The optimizing component, (2) Map logic component, (3) Signal propagation logic component and (4) User interface. An architectural overview of the DPA can be seen in figure 6.1.

The DPA is written in the JavaSE programming language, and developed following an object oriented paradigm with a Model-View-Controller (MVC) design pattern.

6.1.1 The optimizing component

This component handles the placing and minimization of reference nodes. The objective of the component is to produce a deployment design with as few reference node as possible, while satisfying the requirements of a reliable indoor positioning system (constraint 1).

In our application, we implemented the simulated annealing heuristic (SA) (sec-

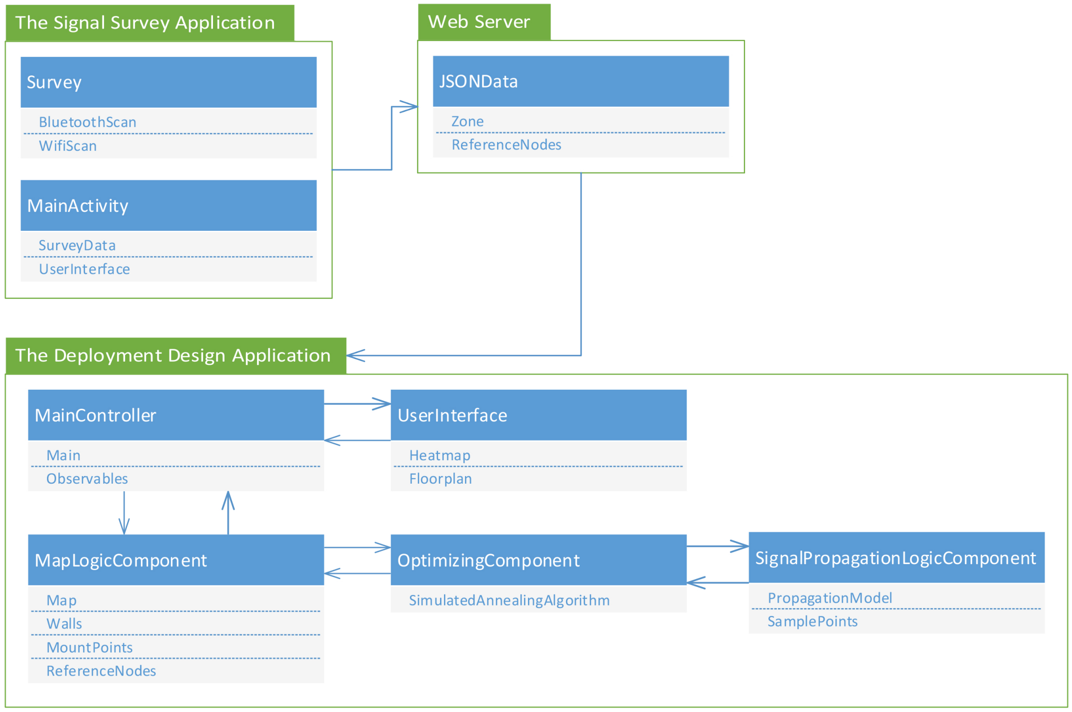

Figure 6.1: The system architecture of the DPA together with the SSSA

tion 5.2.1) for solving the minimization of reference nodes in the deployment design. Our implementation of the simulated annealing heuristic in pseudo code can be seen in algorithm 1.

The algorithm begins with initializing the temperature (line 2) and generates a random deployment design (line 3) with the initial number of reference nodes and assumes that this design it the best found so far (line 4). While the temperature is hot (line 5), the algorithm produce a neighbouring solution (line 7). If the neighbouring solution is better than the global best (line 8), it sets this neighbouring solution as the new global best (line 9) and continues searching with the neighbouring solution (line 10). Otherwise, it compare an acceptance probability with a randomly generated number between zero and one (line 12), which will decide if the search shall continue with the current solution or the neighbouring solution (line 13). This acceptance probability decreases as the temperature drops, thus lowering the probability to continue with a neighbouring solution that is worse than the current solution. We added internal iterations (line 6) to reduce the risk of getting stuck in local optima. Basically, the internal iterations restart the annealing phase m times at each temperature. If a better solution than the current best is found in the new iteration, this solution is assigned as the current best, then the complete simulated annealing process is repeated with a decremented number of reference nodes (line 20). If no better solution is found, or no valid reference node configuration can be produced, the SA returns the best valid solution found so far (line 22).

Algorithm 1: Our implementation of Simulated Annealing to minimize the number of reference nodes to be deployed.

input : Map M, initial number of reference nodes n, initial temperature T_i, number of iterations m
output: Proposed reference node placement

while *There exists a valid solution of n reference nodes* **do**
  $T \leftarrow T_i$; `// initialize temperature`
  $S \leftarrow$ Random solution of M ; `// generate a starting solution`
  $S_{\text{best}} \leftarrow S$; `// assume starting solution is the global best`
  while T *is hot* **do**
    for $i \leftarrow 1 \ldots m$ **do**
      $\hat{S} \leftarrow$ Neighbouring solution to S ; `// generate candidate`
      if $\hat{S}$ *is better than* S_{best} **then**
        $S_{\text{best}} \leftarrow \hat{S}$; `// assign candidate as best so far`
        $S \leftarrow \hat{S}$; `// assign candidate as current best`
      else
        r $\leftarrow$ uniformRand(0,1) ; `// assign random value [0 - 1]`
        if $\exp\left(\hat{S} - S\right)/T > r$ **then**
          $S \leftarrow \hat{S}$; `// accept worse solution`
        end
      end
    end
    Lower temperature T ;
  end
  Decrement n ;
end
return S_{best} ;

6.1.2 Map logic component

This component handles the digital representation of the indoor environment (indoor geometry). We designed it to handle a variety of indoor environments, from simple rooms with parallel walls, to complex facilities with angled walls, non-closed rooms etc.

The indoor geometry in this component is represented from a 2 dimensional top down perspective with walls and structures being modeled by lines. This component also stores the positions where reference nodes can be placed, so called *mount points*. The mount points are automatically generated by using algorithm 2 with a given indoor geometry as input. This algorithm moves along each wall and generates mount points on both sides, with a pre-set granularity. The exception is if one side of a wall is facing the outside of the area being tracked, where in this case, the mount points are only generated on the inside facing side.

Algorithm 2: Generate reference node mount points.

```
input  : Map geometry M, Mount point granularity g_mp
output: A set of mount points
for ∀Wall w_i ∈ M do
    p_i ← start point of w_i ;          // Initialize placement iterator
    p_e ← end point of w_i ;            // Set placement stopping point
    while p_i has not reached p_e do
        if w_i is an outside wall then
            place one mount point at point p_i at the side of w_i facing inside ;
        else
            place two mount points at point p_i, one on each side of w_i ;
        end
        p_i ← move along w_i with step size g_mp ;    /* move to next mount
        point candidate */
    end
end
return placed mount points ;
```

6.1.3 Signal propagation logic component

The purpose of this component is to simulate how signals from reference nodes propagate through a indoor environment. When signals propagate through a medium, the strength the signals will decrease as the travel distance increases, as well as scatter and diffract. Different mediums have different impact on signals. Walls may often be of brick, concrete and/or plaster have very different impact on the signals than air. In an indoor environment, it is likely that the signals travels through several walls, which may have a large impact on the signal strength. For this reason, it is critical to have a good signal simulation which consider wall properties (chapter 4 provides detailed background on propagation models).

This component uses the multi-wall-classic (MWC) propagation model (described in section 4.4) to simulate realistic behavior of signals propagation through open areas and solid objects. It uses the one-slope model (described in section 4.3) when propagating in line of sight, and a wall-specific loss factor when propagating through a wall. The wall specific losses is dependent on properties such as materials and thickness of the walls.

In our prototype, the simulated signal strengths are calculated as a grid over the indoor environment, where each zone represent the signal strength of the area it covers. The signal strength for each zone is calculated by measuring the euclidean distance from the center of the zone to each reference node. This distance, together with the wall collisions in the path, is used in the MWC propagation model to calculate the signal strength in the zone.

Further, to get a sound result from the DPA, it is important that the resolution settings is tuned to match the sampled resolution from the signal strength survey application (section 6.2). If the grid resolution is significantly higher than the sampled resolution, the execution time will see a sharp increase without any rise in accuracy. On the other hand, if the resolution is significantly lower than the sampled values, it will result in a drop of accuracy and possibly misleading results in the later reference node placement.

6.1.4 User interface

The user interface (UI) graphically presents information to ease the interpretation of the proposed deployment design. A screen shot of the user interface can be seen in figure 6.2. The interface is written in Java with the Swing library providing the interactivity and look-and-feel, and consist of a single window with three views.

The first view presents a graphical representation of the coverage, with colored threshold levels for each step in the coverage. The colors representing the coverage levels are stored as an array of hexadecimal ARGB-values and can easily be exchanged for any color palette. White to blue 10-step palette is set as default.

The second view presents the layout of the floor plan and where the reference nodes should be located for optimal coverage, and is also able to display the positions of all generated mount points. In addition to this, the view is also able to display the colored coverage representation below the floor plan, presenting the user with an intuitive display of how the signal coverage propagates in the indoor environment.

The third and last view is a text display that is context sensitive. When clicking on a grid tile in the first view, this view displays the coverage values of the current selected zone. When the clicking on the second view, the real world coordinates in meters are displayed, with origin $\{x, y\} = (0.0, 0.0)$ located in the top left corner of the floor layout and the coordinates increments top $\rightarrow$ bottom, left $\rightarrow$ right. This creates a useful tool for placing the new proposed reference nodes.

6.2 The Signal Strength Survey Application

The purpose of the signal strength survey application (SSSA) is to collect and store the signal strength of Wi-Fi and Bluetooth reference nodes. The SSSA collects the received signal strength indicator (RSSI) by conducting a survey which continuously scans for Wi-Fi and Bluetooth reference nodes for a set time period. The SSSA has a sampling point ID as input for each survey, in order to match the sampling points created in the indoor geometry in the DPA.

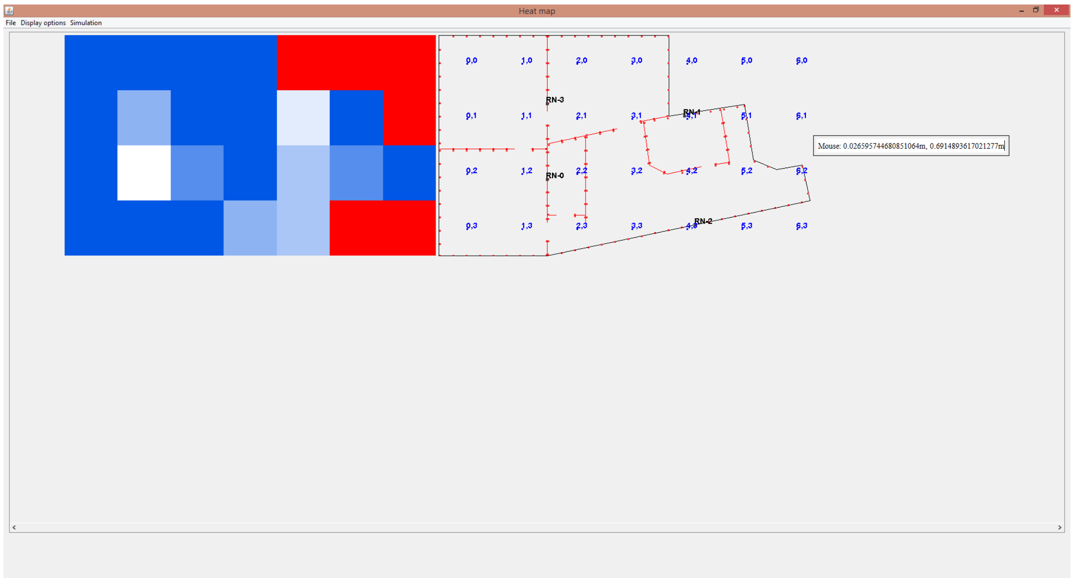

Figure 6.2: An example view of the user interface.

By scanning continuously during a survey, the SSSA obtains multiple RSSI values in order to produce a accurate result. Every time a scan is complete, the RSSI value and accompanied meta data of detected reference nodes is stored. The SSSA then immediately starts a new scan. The survey implementation can be seen in algorithm 3. When a survey has reach its time limit, the stored data is parsed to JavaScript Object Notation (JSON), packaged, and forwarded to a web server. The SSSA is runnable on devices with: Android 4.3+, with support for Wi-Fi and Bluetooth 4.0+.

Algorithm 3: Signal Strength Survey

input : Survey time length t
output: A set of reference nodes with RSSI values and meta data

```
L ← Empty List
on new thread do
    S ← New reference node scanner ;
    while t time has not passed do
        if S.scan is not running then
            start S.scan ;
        end
        when S.scan is finished do
            Append scan results from S to L ;
        end
    end
end
return L ;
```

6.3 Development methodology

The software modules in this project has been developed in a agile fashion. By adapting the development methodology of SCRUM the applications have been developed iterative and incremental. We have distributed the time over three equally long iterations, which in SCRUM are defined as Sprints.

Each Sprint began with defining what works shall be accomplished during its set time period, which for us meant breaking down the requirements in manageable sub tasks and estimating the implementation time. At the end of each Sprint a retrospective analysis was held to review the completed work to determine what went well and what could be improved in the next sprint.

From the start of this project the overall vision of the final solution was quite clear, but not how to implement the functionality. Adapting SCRUM has contributed with an organized way of planning, prioritizing and estimating the development. Implementation issues that could not have been foreseen has been easily dealt with since the implementation have been iterative and divided into small tasks.

7

Experiments

This chapter will present the experiment settings, the environment and the experiments with their subsequent results. The evaluation begins with an survey of simulation accuracy, then three experiments are presented. Each experiment is designed to validate the key design features of the system developed in our thesis.

7.1 Experiments settings

All of the experiments is executed in a office environment provided by Squeed. The office environment offers common indoor properties, such as windows, brick walls, doors and furniture. Some special cases is also included, such as a vault with reinforced walls and a 2500kg heavy iron door, as well as a glass wall. The office measures 11m x 18m in size, and is slightly L-shaped in layout.

An area covering roughly half of the office is used to perform the experiments, with sampling points generated with equal spacing in x- and y-axis, axes being perpendicular to the floor. At each sample point data is gathered during a preset amount of time, with the testing equipment mounted on a tripod always facing the same direction. The changes in environment is kept at a minimum during the gathering, as to get consistent sampling results.

7.1.1 Sampling-, simulation- and heuristics settings

Sampling settings

Paramenter	Value
Sampled points	22
Sample spacing	2.6m
Sample time	20s
Sample frequency	2400Hz
Sampling direction	N. East

Simulation settings

Paramenter	Value
Propagation model	MWE
BLE base strength	-59dBm
Wall attenuation	3.1dBm
LOS decay Rate	2.05dBm
Min.RNs in range	3
Cutoff threshold	-80dBm

Heuristic settings

Paramenter	Value
SA initial T	100
SA α	0.01
SA iterations	10

7.1.2 Indoor environment

The indoor environment used in the experiments can be seen in fig 7.1.

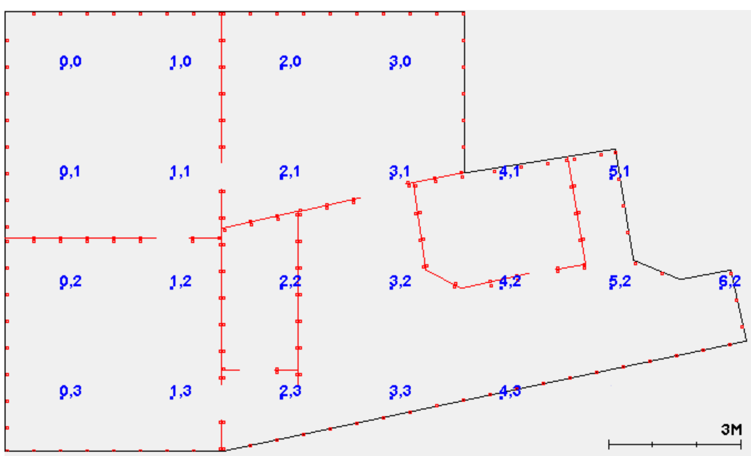

Figure 7.1: Indoor environment with sample points

7.2 Simulation accuracy

This evaluation measures how well our prototype system simulates the signal propagation. It is critical that the system has low error in the simulations, otherwise it could result in irrational and poor performing reference node placements.

This simulation accuracy evaluation is carried out in three steps; First, we let the application generate a RN placement scheme, and simulate the signal values at given sampling points. Next, we place RNs according to the placement scheme, and gather signal data at each sampling point. Last, we calculate the mean error of the simulated data for each sampling point. This procedure will give us an estimate of the reliability of the placement scheme.

7.2.1 Results

We calculated the average simulation error for each zone with our experimental setup. The result can be seen in fig. 7.2; for each zone, the average, together with maximum and minimum error is shown. The data used for the error calculations is the one used in experiment A. The calculations shows that the simulation has an mean average error of 4.3dBm of the RNs, and a maximum of 8.6dBm average error.

We believe that both external and internal factors contribute to the errors in the model. The external factors (or noise) include variations in the environment, as well as environmental factors that is not accounted for in the simulation. This has been expected from the start, as an indoor environment is seldom completely static. Examples of variations is unknown properties of walls (piping, electrical wiring, material variations etc.), changes in humidity and signal interference to name a few.

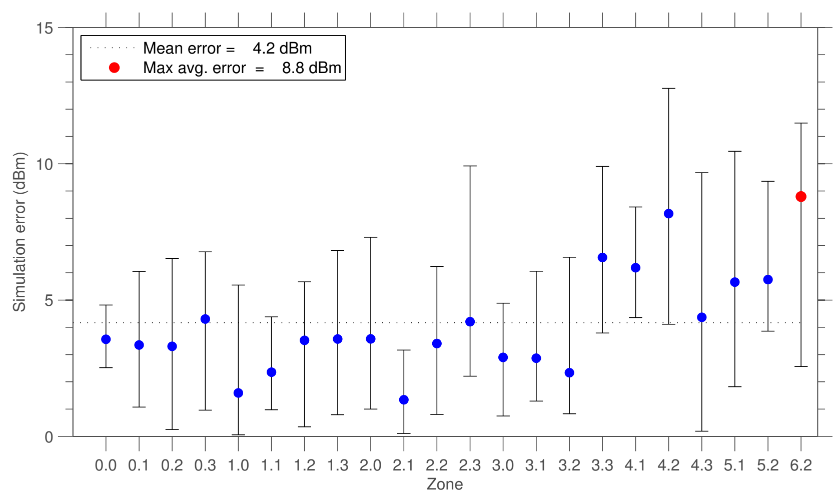

Figure 7.2: Simulation error for each respective sample point.

The errors from internal factors occurs due to simplifications in the signal propagation calculations. We believe the single greatest contributing factor to the internal errors is that the signal falloff is only calculated using the euclidean path from the sampling point to the RN. This creates poor signal strengths when many walls intersect this path, as can be seen in fig 7.3. In reality, many more paths are viable for the signals, thus our model has a tendency to set too large of an signal loss in certain cases. One may argue that this is not too bad, as it at the very least provide some robustness to the signal strength, but it may at the same time add redundant

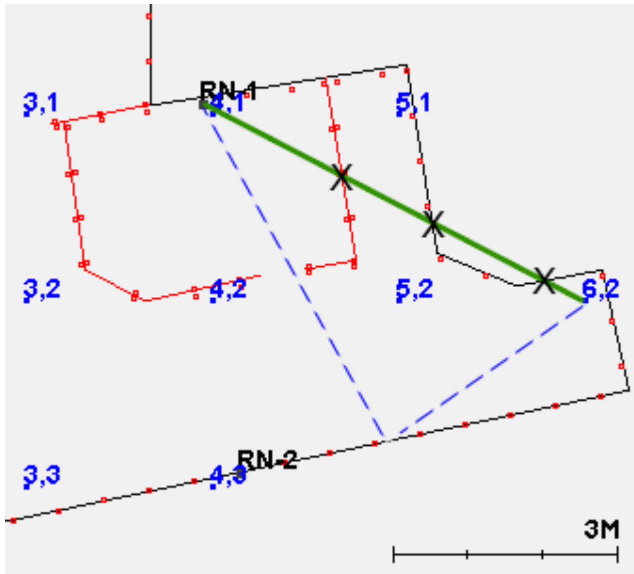

Figure 7.3: Example of unforunate wall penetration. In simulations, the solid line is used to calculate signal fall-off, and due to the high amount of wall passes, the signal fall-off becomes large. In reality, a reflected signal –like the dashed line– has lower fall-off as it passes through fewer walls.

reference nodes. The sample points in the lower right corner of the map can be seen affected by this error. This error can be largely eradicated by using a more detailed signal propagation simulation model, such as ray-tracing (presented in ch. 4).

Overall the simulation software performs well with good accuracy and reliability, with relatively low errors. Furthermore, it has high performance in regard to computation speed; with our benchmark settings, a result is produced with average of ~ 69s execution time.

7.3 BLE extension experiment

The goal of this experiment is to validate that our prototype program produces a practical result. The experiment is designed in 3 parts. Firstly, we collect the Wi-Fi coverage to check if the existing Wi-Fi alone satisfy constraint 1. If it does, the system should return that no more action is needed. Secondly, we extend the Wi-Fi infrastructre with the BLE extension that our program suggests subject to constraint 2. Lastly, we validate our solution by collecting the total coverage to check if the BLE extension together with the Wi-Fi infrastructure satisfies constraint 1.

Evaluation criteria:

- Constraint 1.
- Constraint 2.
- Propagation comparison.

7.3.1 Results

The collected data of the Wi-Fi coverage shows that the signal strength of the Wi-Fi RN is above the threshold of -80dBm at all sample points in the environment. As a result of this, the Wi-Fi RN is included as a reliable RN at all sample points.

The suggested deployment design for the BLE extension from our program can be viewed in Figure 7.4, where the RN placement is marked with a solid black square accompanied with a label **RN-x**. Further, as seen in Fig. 7.5 the gathered

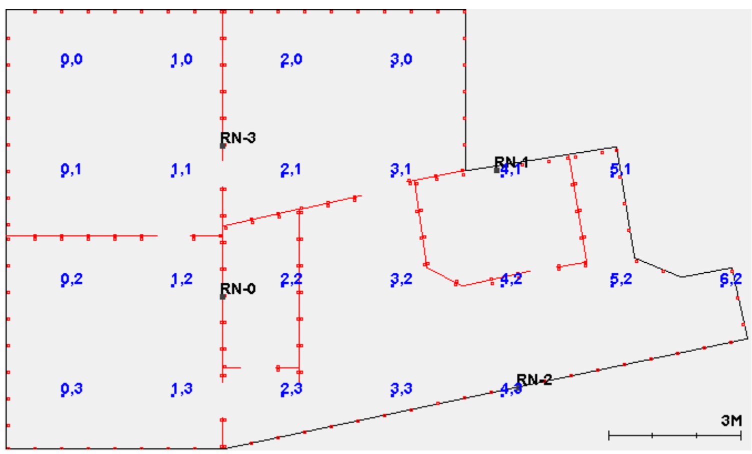

Figure 7.4: Deployment design suggestion for BLE extension.

data reveals that the required signal threshold is held by three RNs at 20 of the 22 sample points. This results remained the same during both executions of the experiment. At sample point 2.0 **RN-0** measure -82.82dBm and at sample point 3.0 **RN-1** measures -81.06dBm, compared to the simulated values of -79.3dBm and -78.65dBm respectivly.

As all sample points fulfills constraint 1 in the program simulation, it becomes apparent that our simulation model is not completely accurate as it cannot predict the noise which exist within a real environment. Thus, they become prone to failure with both miscalculations in the simulation, as well as changes in the environment. Because of this, our program cannot guarantee the set dBm threshold. But the results show, it can produce a deployment which satisfies the set threshold with a few dBm as margin of error.

Concerning constraint 2, and minimizing the number of RNs used, we are using the popular simulated annealing search method for finding minima. Observing the number of redundant overlaps, i.e. sample points which has more than 3 RNs above the set threshold, gives a good indication if our program has placed an excessive amount of RNs. Fig. 7.5 shows that only 3 out of the 22 sample points have redundant overlaps. A valid solution without redundant overlaps has not been found by our simulation in any run, and we believe it is unlikely to exist. Only 3 redundant overlaps is a good indication that our program aggressively minimizes the number of RNs used in the BLE extension.

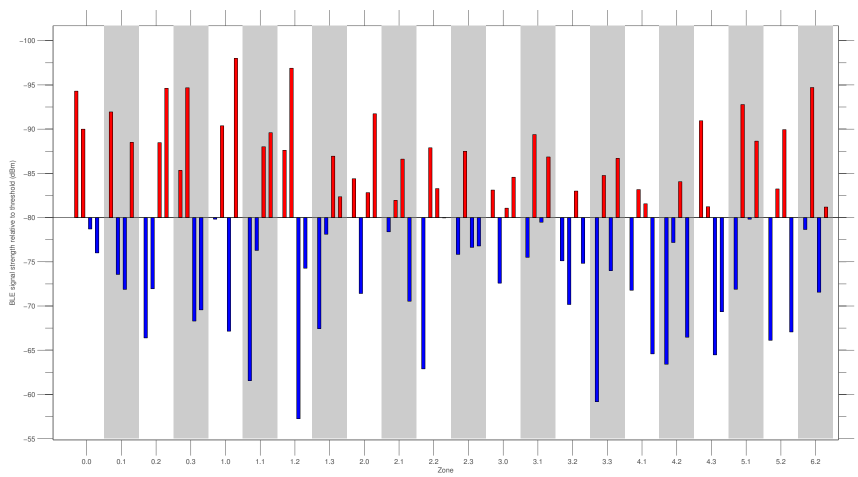

Figure 7.5: The recieved signal strength from each RN relative to the threshold (Wi-Fi not included), grouped by respective sampling points. Red bars indicate the value in dBm above the threshold and blue the value in dBm below the threshold. Note the two zones (2.0, 3.0) failing constraint 1.

7.4 Deployment by individual experiment

The purpose of this experiment is to determine how our program performs compared to an individual in the deployment of indoor position systems. To evaluate the usefulness of our solution, we compare how a placement done by an individual performs in regard to the placement of our program. The individual in question does not have any particular knowledge about our work, but is introduced to the basic concepts of trilateration. The individual is given the same amount of reference nodes as our system suggests and is asked to place these reference nodes to achieve the best possible conditions for trilateration.

Evaluation criteria:

- Constraint 1.
- Deployment comparison.

7.4.1 Results

In this experiment we asked an employee at Squeed to deploy 4 RNs (the same amount our program suggested) to cover the same indoor environment as seen in fig. 7.1. The employee was briefed on the basics of indoor positioning, and what criteria constitute a good positioning system. The employee was further informed that there exist a Wi-Fi access point that provided sufficient signal strength at all sample points. The deployment design can be seen in fig. 7.6.

This deployment did not fulfill constraint 1, and did not perform better than the deployment suggestion from our program. Out of the 22 sample points, 11 failed to fulfill constraint 1. A comparison between the deployment results can be viewed in table 7.1.

A major weakness of the individual deployment design are the RNs which are placed in corners or at the edge of the map. The visual representation of the corner placement is reminiscent of motion detectors and surveillance cameras, which might feel natural and like a good choice. However, when placing a RN near a corner of the map, up to 75% of the possible coverage of this RN becomes located outside of the indoor environment and subsequently unusable. Likewise, when placing an RN at a edge wall of the map, 50% of the coverage becomes unusable. Such preconceived thought patterns could of people could impair deployment designs. Thus, letting software design the deployment by computational means frees it from such patterns, and could potentially find new, unexpectedly efficient designs.

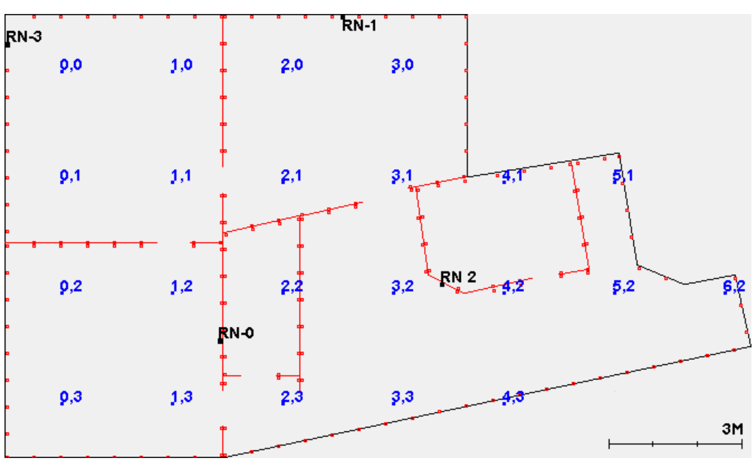

Figure 7.6: Individual deployment design.

	Individual	**Our program**
Number of RNs	4	4
Sample points fulfilling Req 1	11 (50%)	20 (91%)
Number of redundant overlaps	5	3

Table 7.1: Deployment design comparison

7.5 Deployment repeatability experiment

The goal of this experiment is to evaluate the fault tolerance of our system. In a real world scenario it's likely to see loss of reference nodes. The loss of reference nodes may occur by different reasons, for example; batteries may run out, reference nodes can be stolen or destroyed. We will simulate these causes by removing reference nodes from the suggested deployment of our program. We will then collect the coverage and run our program again, to see how it adapts to these changes. This experiment will test the output consistency and show if the program suggests new designs, new amounts of reference nodes or suggests the same original design.

Evaluation criteria:

- Constraint 1.
- Number of new reference nodes suggested.
- New reference node placement.

7.5.1 Results

In each scenario, reference nodes are removed from the original deployment (fig. 7.4), to simulate node failures. Our program is then executed again, with the collected coverage from the faulty deployment, and suggests a new deployment design. Several scenarios is presented, varying both which nodes, as well as the amount of nodes to fail.

Each result is presented as an image of the new suggested deployment. The non-failed nodes remain, and the areas where new nodes is suggested is marked within black circles. None of the new suggested deployments changed the amount of nodes to use, and each area corresponds to one reference node, i.e, two circles represents two nodes, with one node deployed in each circle.

7.5.1.1 One reference node failure scenario

In this scenario, each RN suffers failure, while the other remain active. Each time, the program suggests one new RN placement, which can be seen in 7.7.

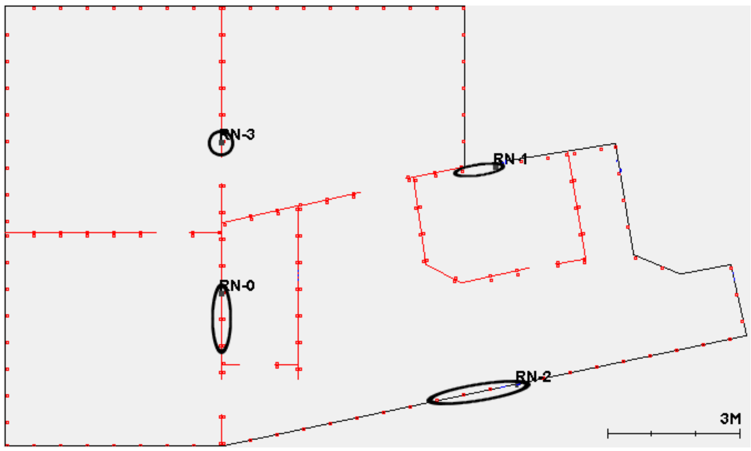

Figure 7.7: One reference node failure scenario. Each circle indicate the area where the program suggest one RN placement.

7.5.1.2 Two reference nodes failure scenario A

In this scenario, **RN-1** and **RN-3** suffers from failure, which causes failure to hold constraint 1 in 15 of the 22 sample points. Our program suggests different deployment designs when executed several times. However, always with the same amount of RNs and minor variations in the placement. The results can be seen in fig. 7.8.

7.5.1.3 Two reference nodes failure scenario B

In this scenario, **RN-0** and **RN-2** suffers from failure, which causes the constraint 1 to fail in 18 of the 22 sample points. Our program suggests different deployment designs when executed several times. However, always with the same amount of RNs and minor variations in the placement. The results can be seen in fig. 7.9.

In every scenario, the new deployment suggestion deviates little to none from the original deployment. The largest deviation occurs in scenario 2b, where the one of the RNs was proposed on two different walls between simulations, and the total deviating distance was roughly 3m for a single RN. As the random search algorithm finds the same or a close optima at each search, the software is clearly capable to repeatability.

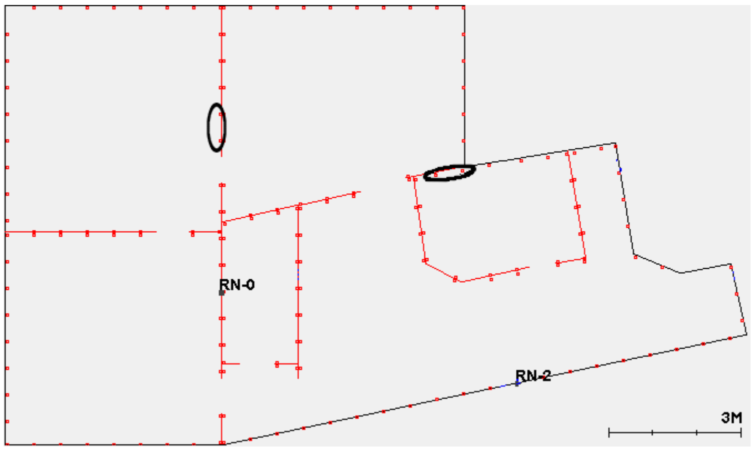

Figure 7.8: Two reference nodes failure scenario A: **RN-3** and **RN-1** failure. Each circle indicate the area where the program suggests one RN placement.

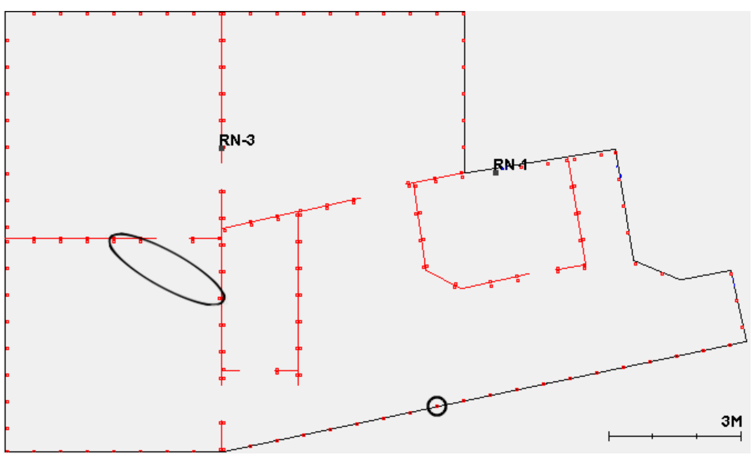

Figure 7.9: Two reference nodes failure scenario B: **RN-0** and **RN-2** failure. Each circle indicate the area where the program suggests one RN placement.

8

Discussion

We present a prototype system and a new analytical approach for deployment design of reference nodes for indoor positioning. Our solution allow already deployed Wi-Fi infrastructure to better support indoor positioning by using an affordable BLE extension. The results shows that our prototype system reduces costs and minimizes the amount of reference nodes while still satisfying the constraints of a reliable indoor positioning system.

8.1 Approach

Our analytical approach is based on modeling indoor positioning as a coverage optimization problem. We simulate the signal propagation of BLE reference nodes with the Wi-Fi radio map by using the multi-wall-classic signal propagation model. Multi-wall-classic was selected for its low computational cost and good accuracy. Higher accuracy is possible to achieve with different models, but will impact the running time performance of the signal propagation logic component. We do not see a need for a signal propagation model with higher accuracy when analyzing the results and considering the performance trade-off it would bring.

To create a deployment design of BLE reference nodes, we apply meta heuristics to find a suitably low amount of reference nodes and their respective placement. The use of meta heuristics allows the prototype system to efficiently discard inefficient subsets in the solution space, and quickly find high performing solutions. However, the drawback of the meta heuristics is that no guarantee of optimiality can be given, due to the stochastic nature of such methods. In our case we do not believe that this can cause any major impacts, as a deployed system is prone to external disruptive elements, such as signal interference, changes in the environment etc. To compute an optimal solution is infeasible, and would provide minimal advantage over an lesser solution, given that the lesser still lies close to optimality in the solution space.

8.2 Experiments

We conducted several experiments to test if our prototype system satisfies the constraints of a reliable indoor positioning system. The experiments were conducted with a Nexus 4 smart phone which has access to both the Wi-Fi and Bluetooth stack, but has some limitations in the ability control how often it retrieves sensor readings. Due to this limitation we decided to collect as many sensor readings as possible during a set time period instead of fixed number of readings. To assure a sound result we collected all readings facing the same direction and same height over ground.

To ensure soundness of the results, all experiments were conducted in the same static indoor environment during two different occasions. The difference between these two occasions were negligible and it seems correct to assume there would be no difference in another similar setting. However, to solidify this assumption, the experiments needs to be conducted in another static indoor environment.

8.3 Literature

One of the related works presented in the introduction of this thesis is Aomumpai et al. [4], where they present a RN placement optimization with binary linear programming. When comparing their results to the ones that we present, one can see clear similarities in the placement patterns of RNs. Both approaches tend to place the RNs in a saw-tooth patter in the indoor environment.

But in Aomumpai et al. [4] several parameters are set with a different approach than what we present in this thesis. The minimum RSSI value that they suggest lies att -100 dBm, which is a very low value in a practical application, and severly lowers the amount of RNs that needs to be placed in an indoor environment. Further, they have suggested a low granularity in their proposed RN placements (3 m in their approach, whereas we suggest 0.75 m). A low granularity limits both the search and the solution space in this kind of optimization problem, especially when taking disturbance factors such as walls in consideration.

9

Conclusion

By viewing indoor positioning reference nodes placement as a coverage optimization problem, we have been able to create a prototype system which effectively produces a reliable and affordable deployment design. By considering the indoor environment and the signal propagation of reference nodes our system can create a deployment design which is reliable and uses very few nodes. However, our approach is best suited for small to medium indoor environment as the manual survey of Wi-Fi infrastructure might be a time consuming process that scales badly for large environments. Limitations also occur if the environment is exposed to major refurbishments as this present changes to signal propagation of reference nodes.

9.1 Future work

Our prototype system's main purpose is to create a deployment design for Wi-Fi and Bluetooth reference nodes, but our model is adaptable for solving problems in other areas. We propose two areas where our model can be adapted to provide solutions to known problems.

9.1.1 Network planning

In network planning and design, the topological design of a network involves how to place links and nodes. Designing the network requires good models of the traffic that is going to be transmitted through the network, as to dimension the network.

Our prototype can be used to find a network provisioning that is fulfills requirements for redundancy and load balancing in routing. By remodeling the signal wave propagation as network traffic and the placed reference nodes as network nodes the prototype is able to find a topological placement for network infrastructure. New requirements and considerations may have to be added, such as varying network traffic at peak hours in order to get a more accurate result.

9.1.2 Placing smoke detectors

When placing smoke detectors it's of high importance to place them in a way that prevents a large area from being covered by a single detector. In the case where a smoke detector fails, another detector must be able to register a fire in a sufficiently short time frame to prevent the fire from spreading. Thus, the placement of the detectors must be chosen with great care.

Taking our prototype, and remodeling the signal wave propagation from each beacon to a smoke detection probability model, the prototype can produce placement suggestions that are redundant and secure. In this example, our prototype may have to be expanded with new considerations, such as not placing smoke detectors near windows or ventilation system.

9.2 Extensions

For future reference, based on this project, we suggest the following improvements and extensions:

- In our model, we only take in mind single floors i.e., a two dimensional approach. If our approch were to be used in multiple floors in the same building, each floor has to be treated individually. By further expand our system to a multi-level approach, where signals from multiple floors could be combined, a more refined solution could be obtained. This could lower the accumulated amount of RNs needed for a multi floor deployment, as deployed RNs is then capable of handling multiple floors at the same time, reducing installation, hardware and maintenance costs.
- The base of this project considers an extension using BLE technology, as it is low cost equipment supported by a majority of modern smart devices. Using up-and-coming technologies which are more suitable for indoor positioning, such as UWB, would provide interesting, possibly better results. This could give an indication of which technology is better suited for IPS, and provide information regarding cost differences.
- The single largest error contributions in our prototype system is the naïve signal propagation model. That is, the system assumes that the lowest expected signal loss occurs at the shortest path from the RN. As can be seen in the previous chapter, that is not always the case. Implementing a more detailed and refined signal propagation model, such as ray tracing, would possible achieve better simulation accuracy, and thus potentially yield even better solutions.
- The positioning equipment in this thesis is only microwave based. The environment could further be extended with supporting technologies such as light,

sound and magnetism to name a few. This type of extensions could provide the system with better resistance against signal interference, more fault tolerance and better precision.

Bibliography

[1] Bluetooth low energy finally earns some respect. `http://search.proquest.com/docview/923610018?accountid=10041`. Accessed: 2015-04-10.

[2] A look at the basics of bluetooth technology. `http://www.bluetooth.com/Pages/Basics.aspx`. Accessed: 2015-04-10.

[3] E. Aitenbichler and M. Muhlhauser. An ir local positioning system for smart items and devices. In *Distributed Computing Systems Workshops, 2003. Proceedings. 23rd International Conference on*, pages 334–339, May 2003.

[4] Supattra Aomumpai, Kittipob Kondee, Chutima Prommak, and Kamol Kaemarungsi. Optimal placement of reference nodes for wireless indoor positioning systems. In *Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), 2014 11th International Conference on*, pages 1–6. IEEE, 2014.

[5] Sofıa Aparicio, Javier Pérez, Ana M Bernardos, and José R Casar. A fusion method based on bluetooth and wlan technologies for indoor location. In *Multisensor Fusion and Integration for Intelligent Systems, 2008. MFI 2008. IEEE International Conference on*, pages 487–491. IEEE, 2008.

[6] IEEE Standards Association et al. 802.11-2012-ieee standard for information technology–telecommunications and information exchange between systems local and metropolitan area networks–specific requirements part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications, 2012.

[7] Al-Gizawi T. Peppas K. Protonotarios E. N. Lazarakis F. I. Papadias C. Philippopoulos P. I Axiotis, D. I. Services in interworking 3g and wlan environments. *Wireless Communications, IEEE*, 11(5):14–20, 2004.

[8] Artur Baniukevic, Dovydas Sabonis, Christian S Jensen, and Hua Lu. Improving wi-fi based indoor positioning using bluetooth add-ons. In *Mobile Data Management (MDM), 2011 12th IEEE International Conference on*, volume 1, pages 246–255. IEEE, 2011.

[9] Abdelmoula Bekkali, Horacio Sanson, and Mitsuji Matsumoto. Rfid indoor positioning based on probabilistic rfid map and kalman filtering. In *Wireless*

and Mobile Computing, Networking and Communications, 2007. WiMOB 2007. Third IEEE International Conference on, pages 21–21. IEEE, 2007.

[10] Walid Ben-Ameur. Computing the initial temperature of simulated annealing. *Computational Optimization and Applications*, 29(3):369–385, 2004.

[11] Monti C.-Vari M. Mazzenga F Borrelli, A. Channel models for ieee 802.11 b. *Communications, IEEE International Conference*, 6:3701–3705, 2004.

[12] KLEIN Bouleimen and HOUSNI Lecocq. A new efficient simulated annealing algorithm for the resource-constrained project scheduling problem and its multiple mode version. *European Journal of Operational Research*, 149(2):268–281, 2003.

[13] Vladimír Černỳ. Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm. *Journal of optimization theory and applications*, 45(1):41–51, 1985.

[14] Eraldo Damosso and Luis M Correia. *COST Action 231: Digital Mobile Radio Towards Future Generation Systems: Final Report*. European Commission, 1999.

[15] Per K Enge. The global positioning system: Signals, measurements, and performance. *International Journal of Wireless Information Networks*, 1(2):83–105, 1994.

[16] Massimo Ficco, Christian Esposito, and Aniello Napolitano. Calibrating indoor positioning systems with low efforts. *Mobile Computing, IEEE Transactions on*, 13(4):737–751, 2014.

[17] Gabriel E Garcia, L Srikar Muppirisetty, Elad M Schiller, and Henk Wymeersch. On the trade-off between accuracy and delay in cooperative uwb localization: Performance bounds and scaling laws. *Wireless Communications, IEEE Transactions on*, 13(8):4574–4585, 2014.

[18] Fred Glover and Gary A Kochenberger. *Handbook of metaheuristics*. Springer Science & Business Media, 2003.

[19] Yanying Gu, Anthony Lo, and Ignas Niemegeers. A survey of indoor positioning systems for wireless personal networks. *Communications Surveys & Tutorials, IEEE*, 11(1):13–32, 2009.

[20] H Hashemi. The indoor radio propagation channel. *Proceedings of the IEEE*, 81(7):943–968, 1993.

[21] Guido R Hiertz, Dee Denteneer, Sebastian Max, Rakesh Taori, Javier Cardona, Lars Berlemann, and Bernhard Walke. Ieee 802.11 s: the wlan mesh standard. *Wireless Communications, IEEE*, 17(1):104–111, 2010.

[22] Kamol Kaemarungsi and Prashant Krishnamurthy. Modeling of indoor positioning systems based on location fingerprinting. In *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, volume 2, pages 1012–1022. IEEE, 2004.

[23] Richard M Karp. *Reducibility among combinatorial problems.* Springer, 1972.

[24] Thomas King, Stephan Kopf, Thomas Haenselmann, Christian Lubberger, and Wolfgang Effelsberg. Compass: A probabilistic indoor positioning system based on 802.11 and digital compasses. In *Proceedings of the 1st international workshop on Wireless network testbeds, experimental evaluation & characterization*, pages 34–40. ACM, 2006.

[25] Scott Kirkpatrick, MP Vecchi, et al. Optimization by simmulated annealing. *science*, 220(4598):671–680, 1983.

[26] Forkel-I. Lott, M. A multi-wall-and-floor model for indoor radio propagation. *Vehicular Technology Conference. VTC 2001 Spring. IEEE VTS 53rd*, 1:464–468, 2001.

[27] Nor Aida Mahiddin, Noaizan Safie, Elissa Nadia, Suhailan Safei, and Engku Fadzli. Indoor position detection using wifi and trilateration technique. In *The International Conference on Informatics and Applications (ICIA2012)*, pages 362–366. The Society of Digital Information and Wireless Communication, 2012.

[28] S. Mazuelas, A. Bahillo, R.M. Lorenzo, P. Fernandez, F.A. Lago, E. Garcia, Juan Blas, and E.J. Abril. Robust indoor positioning provided by real-time rssi values in unmodified wlan networks. *Selected Topics in Signal Processing, IEEE Journal of*, 3(5):821–831, Oct 2009.

[29] Masateru Minami, Yasuhiro Fukuju, Kazuki Hirasawa, Shigeaki Yokoyama, Moriyuki Mizumachi, Hiroyuki Morikawa, and Tomonori Aoyama. Dolphin: a practical approach for implementing a fully distributed indoor ultrasonic positioning system. In *UbiComp 2004: Ubiquitous Computing*, pages 347–365. Springer, 2004.

[30] Lionel M Ni, Yunhao Liu, Yiu Cho Lau, and Abhishek P Patil. Landmarc: indoor location sensing using active rfid. *Wireless networks*, 10(6):701–710, 2004.

[31] Moon-Won Park and Yeong-Dae Kim. A systematic procedure for setting parameters in simulated annealing algorithms. *Computers & Operations Research*, 25(3):207–217, 1998.

[32] Jin Qin, Ling-lin Ni, and Feng Shi. Combined simulated annealing algorithm for the discrete facility location problem. *The Scientific World Journal*, 2012, 2012.

[33] Tapan K Sarkar, Zhong Ji, Kyungjung Kim, Abdellatif Medouri, and Magdalena Salazar-Palma. A survey of various propagation models for mobile communication. *Antennas and Propagation Magazine, IEEE*, 45(3):51–82, 2003.

[34] Kumar Ghodgaonkar-D. Ali B. M. Khatun S. Sujak, B. Indoor propagation channel models for wlan 802.11 b at 2.4 ghz ism band. *Applied Electromagnetics, APACE, Asia-Pacific Conference*, 2005.

[35] WK Tam and VN Tran. Propagation modelling for indoor wireless communication. *Electronics & Communication Engineering Journal*, 7(5):221–228, 1995.

[36] Mattias Wahde. *Biologically inspired optimization methods: an introduction*. WIT press, 2008.

[37] Laurence A Wolsey. *Integer programming*, volume 42. Wiley New York, 1998.

[38] Zhe Xiang, Song Song, Jin Chen, Hao Wang, Jian Huang, and Xingxin Gao. A wireless lan-based indoor positioning technology. *IBM Journal of research and development*, 48(5.6):617–626, 2004.

[39] Wendong Xiao, Wei Ni, and Yue Khing Toh. Integrated wi-fi fingerprinting and inertial sensing for indoor positioning. In *Indoor Positioning and Indoor Navigation (IPIN), 2011 International Conference on*, pages 1–6. IEEE, 2011.