

Vemund Rogne

Bluetooth Angle of Departure for Indoor Industrial Environments

Improving Carrier Frequency Offset estimation and exploiting frequency-dependent reflections to reduce measurement error

Master's thesis in Cybernetics and Robotics

Supervisor: Kristoffer Gryte

June 2023

Vemund Rogne

Bluetooth Angle of Departure for Indoor Industrial Environments

Improving Carrier Frequency Offset estimation and
exploiting frequency-dependent reflections to reduce
measurement error

Master's thesis in Cybernetics and Robotics
Supervisor: Kristoffer Gryte
June 2023

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Engineering Cybernetics



This thesis marks the end of my six years as a student at the Norwegian University of Science and Technology (NTNU) in Trondheim.

While the maths and engineering have been valuable lessons; these six years have been, first and foremost, a journey of personal growth.

I would like to thank Sara, Snorre, Maren, Oda, Cecilie, Andreas, Kristian, Synne, Guro, Jonathan, Samson, Jørgen, Petter and many more for making these years the best of my life – you are all amazing people and will always have a place in my heart.

It has been a wonderfully defining period of my life and I am forever grateful to have had this opportunity.

Abstract

Bluetooth 5.1 introduced the feature 'Direction Finding' which allows bluetooth devices to estimate the direction from one device to another by the use of antenna arrays. But the specification does not specify more than the low-level antenna switching and sampling and it is left to the programmer to bridge the gap from raw data to direction estimates. This is not trivial.

Part of the work in this thesis is improving real-time Carrier Frequency Offset estimation (a critical issue in Bluetooth direction finding) by optimizing an existing solution with numerical optimization techniques to increase computational performance by several orders of magnitude.

A conventional beamformer from the literature is used for direction estimation, but a major challenge is reflections which alter the output of the direction finder and there is little information in the literature as to how those can be mitigated. To this end we use both simulations and real-world experiments in an AutoStore warehouse directed toward three main hypotheses:

- H1 Reflections are highly frequency dependent in indoor situations due to the geometry of the reflection and direct path: A slight change in wavelength changes the output of each path differently.
- H2 Combining multiple channels compensates for reflection-caused errors in some cases
- H3 Searching for multiple solutions of the direction finder is important

All three are shown both theoretically and experimentally to be correct. We show that using multiple channels can reduce the RMSE in direction finding from 19.3° to 3.0° in some cases.

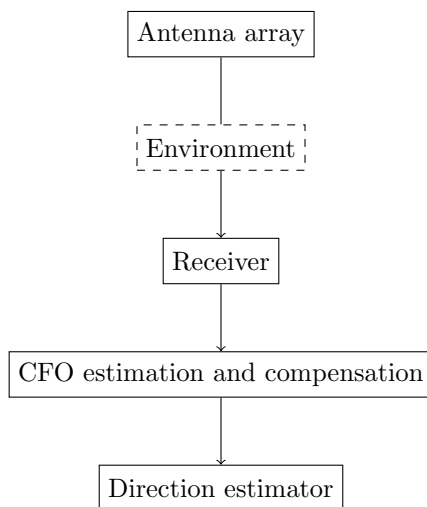
The author is currently employed at AutoStore where he works on a separate project spawned from the authors Bachelors thesis. The results of this masters thesis could be of interest to AutoStore and the company has allowed the use of their testsite and provided $\approx 20k$ NOK in equipment and travel expenses.

Acknowledgements

While writing a masters may be lonely, it is not a solo project.

Thank you to Kristoffer Gryte, my supervisor, for guidance and navigation in the work. Krisitan Brudeli and Jon Rogne for their assistance in writing. And Jonathan Eide, Andreas A. Eide and Sara J. Wolfgang for their relentless patience while listening to my ramblings about directions and frequencies for the past few months.

Outline



Chapter 1: Introduces direction finding by explaining the basic working principle and the details of the Bluetooth specification of direction finding, and introduces the core issues: Carrier Frequency Offset and reflections.

Chapter 2: Details Carrier Frequency Offset estimation and compensation. We optimize a method found in the literature and investigate the importance of good CFO estimation with Monte Carlo simulations.

Chapter 3: Details the implementation of a beamformer for direction finding that utilize frequency diversity for better performance under the influence of reflections. A basic ray-tracing model is implemented to investigate the influence of frequency and position in the disturbance from reflections of the signal.

Chapter 4: Explains the setup for experiments at the AutoStore testsite. Details the specifics of the array, receiver and operation of the AutoStore site.

Chapter 5: Shows the results of the experiments at the AutoStore site.

Chapter 6: Concludes the thesis and gives some recommendations for future work.

Contents

1	Preliminaries	1
1.1	An introduction to Direction Finding	1
1.2	Challenges with direction finding	4
1.2.1	Carrier Frequency Offset	4
1.2.2	Reflections	5
1.3	Summary of project thesis	5
1.4	Bluetooth Low Energy and Direction Finding	5
1.4.1	Channels	5
1.4.2	Direction Finding – Constant Tone Extension	5
1.5	Numerical Optimization	6
2	Estimation and Compensation of Carrier Frequency Offset	7
2.1	Carrier Frequency Offset Model	8
2.1.1	Knowns, unknowns and assumptions	8
2.2	Estimating Carrier Frequency Offset	9
2.2.1	Cost function and optimization problem	9
2.2.2	Developing equations for calculating cost and derivative of cost	10
2.3	Efficient solution of the optimization problem	11
2.3.1	Implementing the estimation strategy	12
2.3.2	Accuracy of the estimation	14
2.4	Impact of reflections on the CFO estimation	15
2.5	Outliers	16
2.5.1	CFO outlier-rejection strategy	16
2.6	Switch-pattern dependence	18
3	Direction estimation	19
3.1	Direction Finding Algorithms	19
3.1.1	Frequency diversity	19
3.2	The classic beamformer for direction finding	20
3.2.1	Cost function	20
3.2.2	Deriving the derivative of the cost function	21
3.2.3	Implementing optimization strategy	21
3.3	Reflection simulation – frequency and position	21
3.3.1	Single reflection and frequency dependence	22
3.3.2	Single reflection and position dependence	24
3.3.3	Multiple reflections	24
3.3.4	Combining frequency and position dependence	25
3.4	Switch-pattern dependence	25
4	Experimental setup	27
4.1	Placement of arrays at the test-site	27
4.2	Movement of the robot on the grid	28
4.3	Antenna-array configuration	28
4.3.1	Hardware – Insight SIP-1907 AoA-DK	28
4.3.2	Antenna-array hardware bugs	28
4.3.3	GPIO changes	29

4.3.4	Switching pattern	30
4.3.5	Firmware	30
4.4	Receiver configuration	30
4.4.1	Hardware – NRF52833DK	30
4.4.2	Firmware	30
4.5	Logging and interfacing to the AutoStore system	30
5	Results	31
5.1	Frequency dependence	31
5.2	Monte carlo RMSE with N frequencies	34
5.3	Grid measurements	36
6	Conclusion	41
6.1	Future work	41

Chapter 1

Preliminaries

Bluetooth 5.1 introduces the feature 'Direction Finding' which allows Bluetooth devices to estimate the direction from one device to another. This is achieved by appending a 'Constant Tone Extension' to a Bluetooth packet, which is either sent from multiple antennae (Angle of Departure (AoD)) or received by multiple antennae (Angle of Arrival (AoA)) – measuring the relative change in the constant tone using multiple antennae enables estimating the signal direction.

This chapter explains the basic working principle of direction finding. Knowledge of how Bluetooth works is not required as the section of interest is the constant tone extension, which is just a pure sine-tone.

1.1 An introduction to Direction Finding

When an oscillating voltage is applied to a suitable piece of metal (an antenna) a field of electric and magnetic energy radiates outward at the speed of light. A simplified description of these fields are wave functions of the electric and magnetic field [1]. In our work we simplify these fields into a single quantity which we will think of as the amplitude of the electromagnetic wave at each point in space and time, equation 1.1 is a description of this wave.

$$A(d, t) = A_{max} \cos(kd - 2\pi ft) \quad (1.1)$$

where A_{max} is the max amplitude, d is the distance from the source, k is the wave number which is described by the wavelength $\frac{2\pi}{\lambda}$, f is the frequency of the signal and t is time. The wave travels at the speed of light along the d direction which results in the wavelength $\lambda = \frac{c}{f}$.

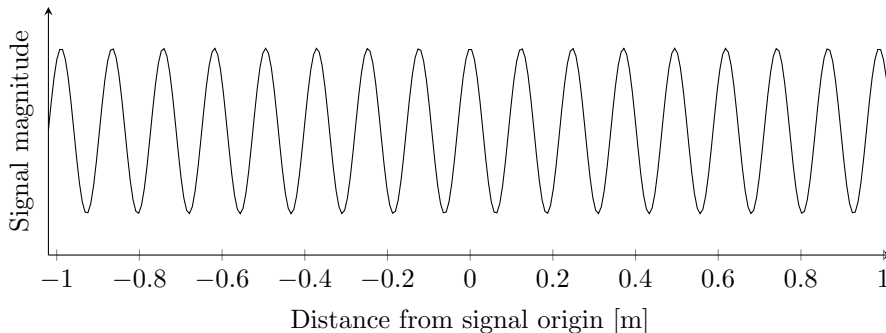


Figure 1.1: Signal power at a distance d from the antenna. This is a visualization of (1.1) when the frequency of the signal is 2.42 GHz and has a wavelength $\frac{c}{f} \approx 0.12$ m

The signal spreads out spherically in every direction from the source of the signal. Along each vector from the source the amplitude follows equation 1.1 and is shown in figure 1.1. A 2D representation of the signal is shown in figure 1.2 which illustrates an important fact: At a distance sufficiently far away from the source, and at an area sufficiently small, the wavefront looks like a plane-wave (a wall) traveling in

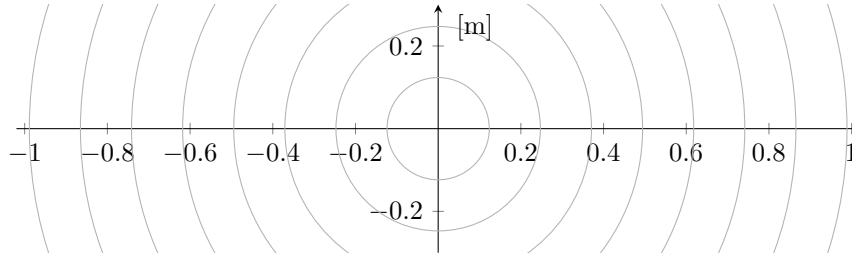


Figure 1.2: Simplified 2D diagram of the wave-front produced by an antenna transmitting a pure sine-tone at 2.42 GHz.

one direction through the area [1][2]. In this thesis the arrays are small enough, and at a distance great enough, that we will always assume that the waves behaves like plane-waves.

Figure 1.3 shows the wave at 10.50 m from the transmitter received by an antenna array which is angled 20.00° relative to the wave¹. The array is a uniform linear array² with spacing $\Delta d = 5.00$ cm. Due to this angular difference between the plane-wave and the array the received signals between two adjacent elements are offset by a phase difference ϕ described by equation 1.2:

$$\phi = 2\pi \frac{\Delta d * \sin(\theta)}{\lambda} \quad (1.2)$$

where Δd is the distance between adjacent elements and λ is the wavelength of the received signal.

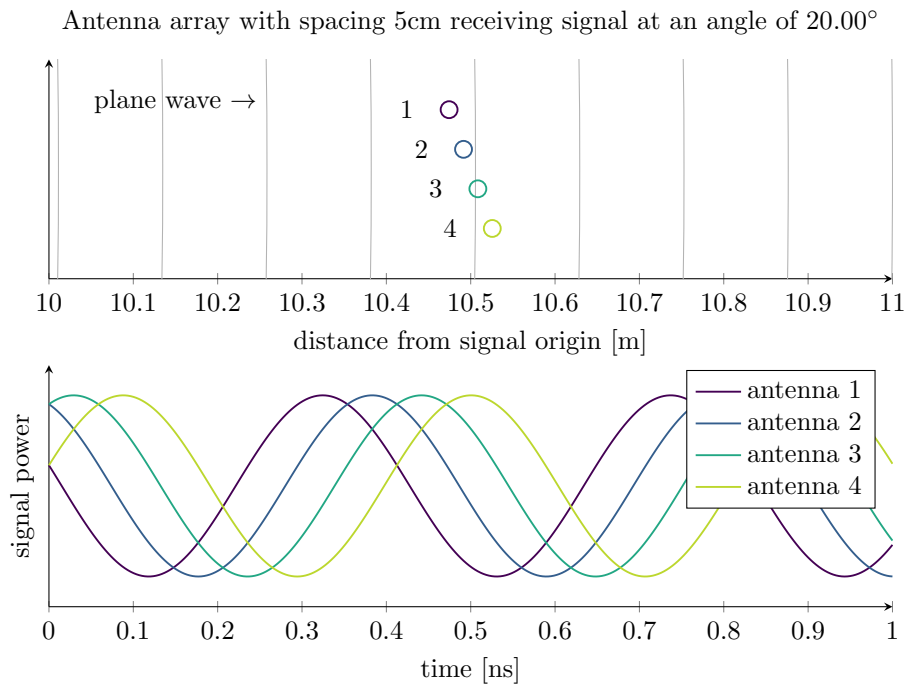


Figure 1.3: Antenna array receiving the signal with four antennas. Each antenna receives the same signal delayed by some amount depending on the angle of the array relative to the plane wave. The phase shift calculated with equation 1.2 is in this case 51.3°.

If we can measure the phase difference between each signal in figure 1.3 we could use equation 1.2 to estimate the angle θ .

It is possible to directly measure these signals and digitally process them to find the phase-difference, but it is not feasible for consumer products; ADC08DJ5200RFAAV is an Analog to Digital Converter (ADC)

¹Notice that spherical nature of the wave is unobservable in this plot even though it is generated with a circle

²Each antenna element is uniformly placed at a line with equal spacing

from Texas Instruments which can directly measure signal at 2.40 GHz, but it is prohibitively expensive³ and subject to shipping restrictions, presumably due to its potential use in military applications.

Instead of directly sampling the raw signal one can use a process called downconversion: a signal is multiplied by another sinusoid which produces two new sinusoids with new frequencies and phase – we show this operation explicitly in equation 1.3 using the well known-Eulers Identity, and its inverse relations, both from [3]. The key idea is that when the high-frequency component is filtered out the remaining low-frequency component is much easier to sample.

$$\begin{aligned}
 \sin(\omega_1 t + \phi) \cdot \sin(\omega_2 t + \theta) &= \sin(\alpha) \cdot \sin(\beta) = \frac{e^{j\alpha} - e^{-j\alpha}}{2j} \cdot \frac{e^{j\beta} - e^{-j\beta}}{2j} \\
 &= -\frac{1}{4} (e^{j\alpha} e^{j\beta} - e^{j\alpha} e^{-j\beta} - e^{-j\alpha} e^{j\beta} + e^{-j\alpha} e^{-j\beta}) \\
 &= -\frac{1}{4} (\cos(\alpha + \beta) + j \sin(\alpha + \beta) \\
 &\quad - \cos(\alpha - \beta) - j \sin(\alpha - \beta) \\
 &\quad - \cos(\beta - \alpha) - j \sin(\beta - \alpha) \\
 &\quad + \cos(-\alpha - \beta) + j \sin(-\alpha - \beta)) \\
 &= \frac{1}{2} \cos(\alpha - \beta) - \frac{1}{2} \cos(\alpha + \beta) \\
 &= \frac{1}{2} \cos((\omega_1 - \omega_2)t + \phi - \theta) - \frac{1}{2} \cos((\omega_1 + \omega_2)t + \phi + \theta)
 \end{aligned} \tag{1.3}$$

This can be used to measure the phase of the signal in relation to a local oscillator. The signal is first multiplied by the local oscillator and then by a shifted variant of the local oscillator, the amplitude of each resulting signal describes a complex vector which has an angle equal to the phase shift between the signal and the local oscillator. This method is used in Bluetooth Direction Finding and is referred to as In-phase and Quadrature measurement and a block diagram describing it is shown in figure 1.4.

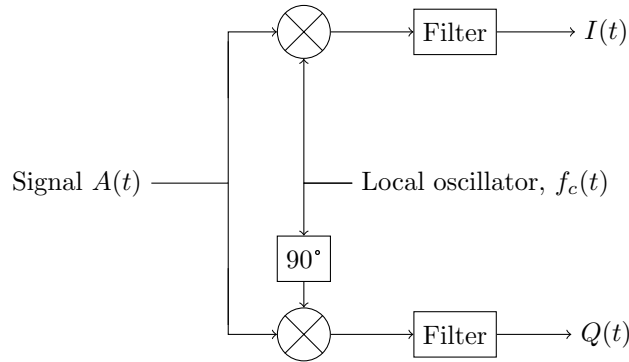


Figure 1.4: In-phase and quadrature block diagram. The signal $A(t)$ is multiplied with the local oscillator and its shifted variant to produce two signals, which when low-pass filtered produce the I and Q signals that measure the phase of the signal in relation to the local oscillator. The signals $I(t)$ and $Q(t)$ are low-frequency and easier to measure than the high-frequency input signal

With a phase measurement of each antenna one can estimate the angle of the signal using beamformer theory which describes how to direct energy from an array of antennas in specific directions in space. Only a small subject of beamformer theory is needed for bluetooth direction finding due to the way it is implemented: only one antenna is sampled/transmitted at any one time, as such only the delay-and-sum beamformer is applicable. Note that Bluetooth *is not capable* of beamforming as antenna elements are not transmitting or receiving at the same time – the theory is however applicable to estimating the direction of the received signal.

A delay-and-sum beamformer takes advantage of the superposition principle (two signals are added together linearly), which applies to radio-frequency waves [2][1]. By delaying transmission from an

³ ≈ 20k NOK – but at least shipping is free

antenna by the amount of time it takes for the plane-wave to reach the next antenna it can constructively add its transmission on top of the other. Figure 1.5 shows the wavefronts of four antennas transmitting with different delays to steer the signal.

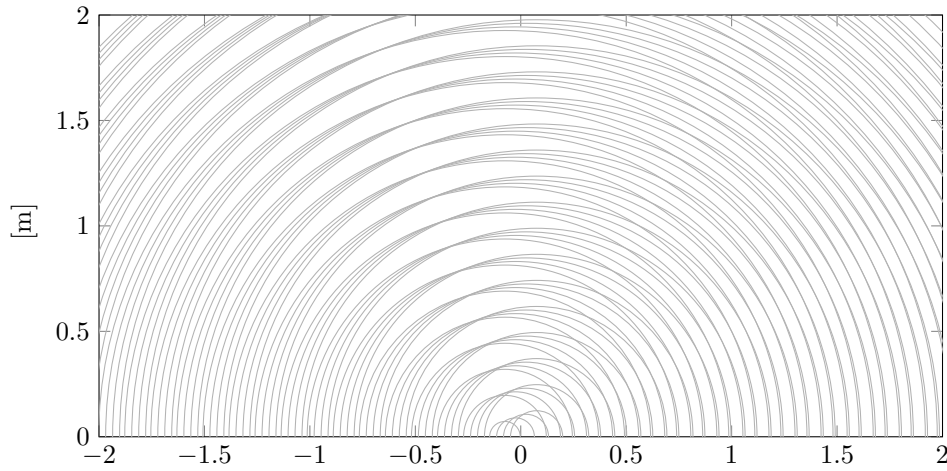


Figure 1.5: Beamforming using delayed transmission from two antenna elements placed 5.00 cm apart transmitting at 2.42 GHz. The phase-shift between each antenna is 51.30° to direct energy toward the angle 20.00° from the array

At each point in space the signal can be found by summing equation 1.1 for each antennas distance d to the receiver. The summation of sinusoids is important here: The sum of two sinusoids of identical frequency but differing phase and magnitude is a new sinusoid with a new phase and magnitude but still the same frequency. We refer to [4] for the equation:

$$A \cos(\omega t + \alpha) + B \cos(\omega t + \beta) = \sqrt{(A \cos(\alpha) + B \cos(\beta))^2 + (A \sin(\alpha) + B \sin(\beta))^2} \cdot \cos\left(\omega t + \tan^{-1}\left(\frac{A \sin(\alpha) + B \sin(\beta)}{A \cos(\alpha) + B \cos(\beta)}\right)\right) \quad (1.4)$$

Equation 1.4 is of great importance and reveals a major challenge in bluetooth direction finding: reflections.

Estimating the direction in Bluetooth Direction Finding can be done by calculating how much 'power' is present in each direction based on how a transmitting array would have directed its power. There is no mathematical difference if the array is transmitting or receiving power, but nomenclature is often that the array is transmitting [2].

1.2 Challenges with direction finding

There are two primary challenges with implementing direction finding for an indoor industrial environment: Carrier Frequency Offset which affects how phases are measured when the signal is received, and reflections which affects how phases arrive at the receiver.

Carrier Frequency Offset can be estimated and compensated for and therefore be eliminated, but reflections are harder to deal with and only mitigated in this thesis by exploiting channel-diversity.

1.2.1 Carrier Frequency Offset

The In-phase and quadrature technique that was previously discussed and shown in block-diagram form in figure 1.4 requires that the frequency of the local oscillator is equal to the frequency of the incoming signal. Equation 1.3 can be used to predict the output if this is not the case: the output will be $\propto \cos(\omega_1 - \omega_2)t + \phi - \theta$ and the measured phase drifts constantly. It is still the case that each antenna has a predictably different phase $\phi - \theta$, but the problem is that it takes time to measure each antenna: by the time the next antenna is measured, the reference has changed and the output is wrong.

With bluetooth direction finding, where receiver and transmitter both have low-cost oscillators, the Carrier Frequency Offset $\omega_1 - \omega_2$ is significant and the phase measurements are heavily affected. Chapter 2 is devoted to developing a method to estimate and correct for the Carrier Frequency Offset that could run real-time on an embedded system.

1.2.2 Reflections

A radio wave can be reflected of a surface [1] and then, as with beamforming, constructively or destructively combine. When two waves combine both amplitude and phase are changed as shown in 1.4 – since direction finding relies completely on phase-measurements it will be negatively affected by reflections that change the phase of the signal.

Reflections have been shown to negatively affect Bluetooth Direction Finding for outdoor situations in [5], and indoor situations in [6]: In contrast to the outdoor situation where reflections are not very frequency dependent, in indoor situations the output of the direction finder is significantly affected by the frequency of the signal. This might be due to the different geometry in reflections for indoor situations making them more frequency dependent. A method of simulating the effects of reflections is ray-tracing where radio waves are modelled as particles [2][7] and their phase is found by the distance travelled and equation 1.1. We make a simple version of such a ray-tracing model in chapter 3 to investigate the effect of frequency and position on reflections to form a strategy to mitigate the effects of reflections for indoor situations.

1.3 Summary of project thesis

The project thesis is a smaller preproject to this masters, in that work four main points were established:

- Estimating Carrier Frequency Offset is challenging and critical
- The errors in each individual measurement is significantly affected by the channel in use
- The errors are significantly affected by the position
- Summing up the directional-power-spectra for multiple channels improves the total estimate, often to within $\pm 5^\circ$

1.4 Bluetooth Low Energy and Direction Finding

Bluetooth Low Energy is a complex system to wirelessly connect multiple devices while consuming minimal amounts of power. In this work the only part of interest is the CTE and a full description of Bluetooth Low Energy is left to others. We will however outline the critical parts for direction finding: Physical RF Channels and Constant Tone Extension configuration.

1.4.1 Channels

Bluetooth Low Energy (BLE) operates in the unlicensed 2.40 GHz ISM band. BLE is divided into 40 channels that are separated by 2.00 MHz: three are advertising channels spread evenly across the spectrum and 37 are used as data channels. The center frequency of each channel is given by $f = 2402 + k \cdot 2.00$ MHz, $k = 0, \dots, 39$. The advertising channels are at $k = 0, 12, 39$. A bluetooth device advertises its existence on an advertising channel and when two devices connect to each other they move to the data channels and employ a synchronized pseudo-random frequency hopping between the 37 data channels to reduce collisions and multipath-fading [8].

1.4.2 Direction Finding – Constant Tone Extension

A Constant Tone Extension (CTE) is a constantly modulated series of unwhitened $1s^4$ [8] – this just means that there is a constant oscillation of the antenna at the channel frequency plus a frequency shift of 250.00 kHz or 500.00 kHz depending on PHY which is due to frequency modulation.

⁴The terminology used here comes from wireless communications. Digital signals are sent over analog domains. Sending a constant series of 1s results in a pure oscillation: a constant tone

There are two primary modes of operation: Angle of Departure (AoD) and Angle of Arrival (AoA). In AoD mode the array is on the transmitter and the CTE is sent from multiple antennas. In AoA mode the array is on the receiver and the CTE is sent from a single antenna and received at multiple antennas. The order that each antenna is switched and sampled is controlled by a switching pattern [8]. The receiver always samples and collects data in the form of IQ-samples, which are a method of measuring phase and explained in section 1.

For direction finding it is mathematically irrelevant whether AoD or AoA is used due to the Reciprocity Theorem – if a propagation medium is linear and time-invariant then the channel from unit 1 to unit 2 is identical to the channel from unit 2 to unit 1 [9]. The difference between AoA and AoD will then only be apparent in the difference in how radio-waves are transmitted by either the Array or single antenna. [10] suggest that microstrip antennas have little mutual coupling, such that even if one antenna in the array is transmitting there is not much power transferred to neighbouring elements which can transmit further. This seems to be verified by [11] who found minor difference with or without compensating for mutual coupling.

The first 4.00 μs is a guard period, the next 8.00 μs is the reference period. After the reference period there is a sequence of switch and sample slots that are 1.00 μs or 2.00 μs depending on the configuration [8]. The Bluetooth Specification [8] notes that 'In order to obtain good quality data for angle estimation [...] starts 0.125 μs after the beginning and ends 0.125 μs before the end of each microsecond period' [8]. Therefore it is not exactly 2 or 4 microseconds from the last reference period sample and the first sample period sample, which must be taken into account in the software later.

1.5 Numerical Optimization

Numerical Optimization is a collection of methods to numerically find the best, or optimal, value for some parameter based on a cost function. The cost function quantifies 'how good' the current value is.

It is convention to formulate numerical optimization as minimization problems rather than maximization problems, this is of course equivalent by simply changing the sign of the cost function [12]. In this thesis all problems will be formulated as minimization problems.

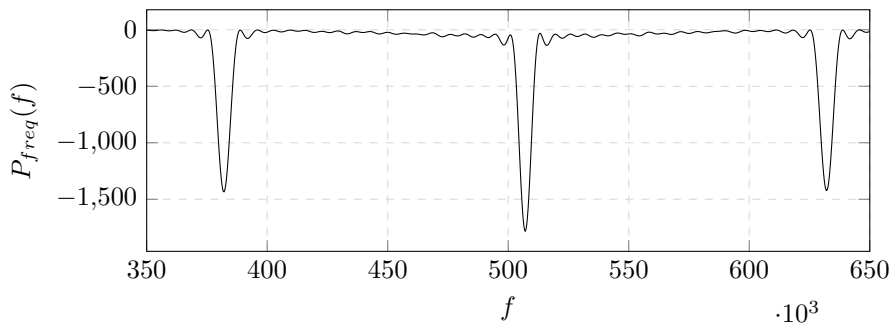


Figure 1.6: Example of a cost function. This cost function is taken from chapter 2

A value is a local optimizer of some region if it is the smallest value in that region and the gradient of the cost function is zero. In example 1.6 there are multiple local optima depending on the size of the region you are looking at.

When programming a computer to optimize a function it is important that the optimization strategy uses as few evaluations of the cost function as possible to reduce the computation time. A common method to quickly find an optimal value is to perform a strategy like gradient descent: Evaluate the gradient at the current best guess and move in the direction of the gradient until the gradient reaches zero.

A key factor in the speed of the optimization strategy is the step size – how far should the method go with the gradient: too far and it may overshoot the optimal value and never converge, too short and the convergence rate will be slow.

Several different optimization routines are available in the literature. In this work the BFGS2 optimization routine from [12] and an implementation of the algorithm from [13]

Chapter 2

Estimation and Compensation of Carrier Frequency Offset

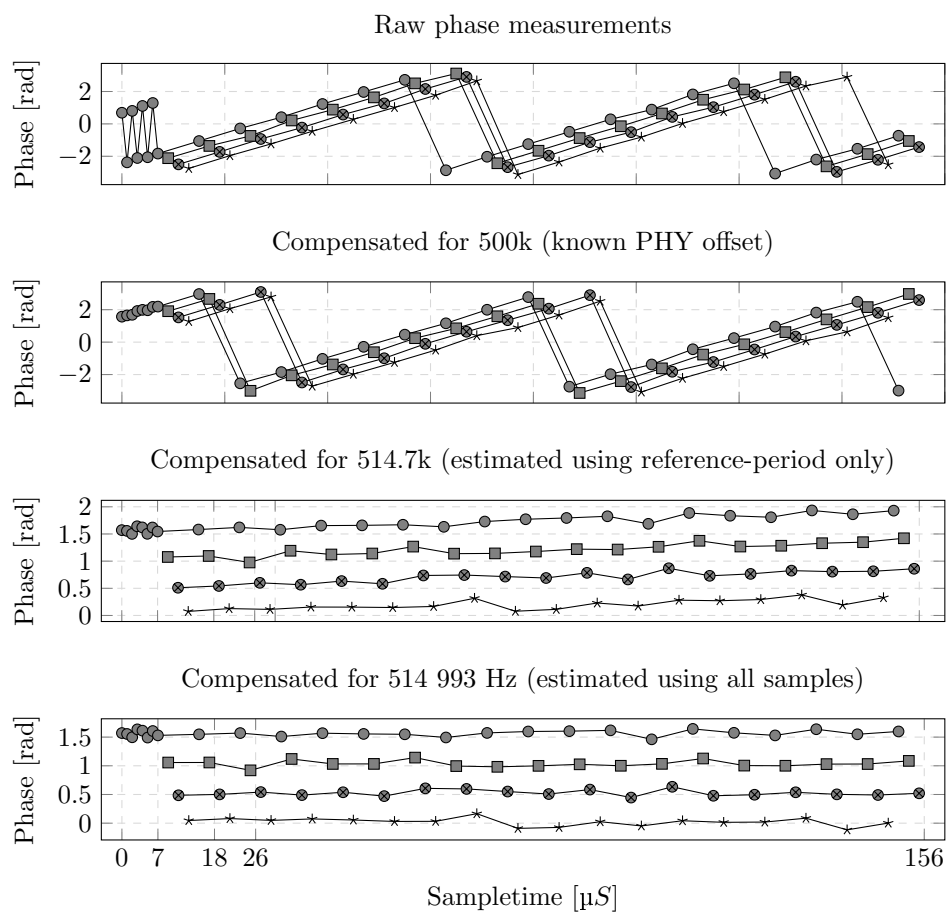


Figure 2.1: Phase measurements of antennas with different estimations of CFO for a simulated CTE. The true CFO value is 515.00 kHz. It may seem that the difference between reference period only and all samples is small, but on average it is critical to use all samples due to the limited number of samples in the reference period as will be seen in figure 2.7

Figure 2.1 shows the stages of a CFO estimation procedure.

A normal frequency error¹ between transmitter and receiver could be 7.00 kHz; in the duration of a

¹Based on our data

CTE of length $160.00\ \mu\text{s}$ that will change the phase around 400.00° – if this is not compensated for the measured phase will drift significantly and the average value will approach 0 due to the cyclical nature of phase measurements. This effect shows up by the tendency of a poorly compensated CTE to output angles close to zero.

A Monte-Carlo simulation with varying standard deviation in CFO error, as shown in 2.2, reveals how critical it is to estimate and compensate for CFO – even $1.00\ \text{kHz}$ or more will cause RMS-Error in angle estimation to be at least 1.00°

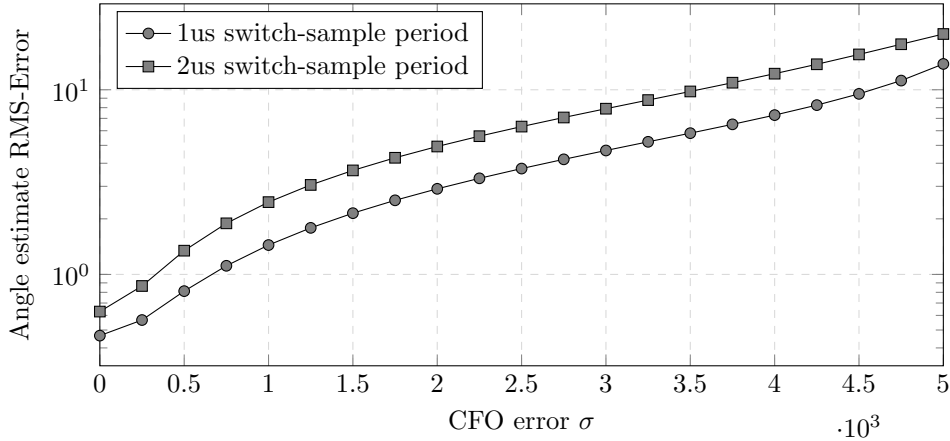


Figure 2.2: Monte Carlo simulation of angle-estimation Root Mean Square Error (RMSE) with increasing standard deviation in CFO

2.1 Carrier Frequency Offset Model

The frequency of the CTE is governed by four components: Channel center frequency, modulation shift, doppler shift and transmitter frequency error.

$$f_t = f_c + f_m + f_d + \Delta f_t \quad (2.1)$$

where f_t is the frequency of the transmitted signal, f_c is the channel center frequency, f_m is the modulation shift, f_d is the doppler shift and Δf_t is the frequency error of the transmitter.

The receiver measures the phase of the CTE in relation to its onboard oscillator, which is governed by two components: Channel center frequency and receiver frequency error.

$$f_r = f_c + \Delta f_r \quad (2.2)$$

where f_r is the frequency of the receivers oscillator, f_c is the channel center frequency and Δf_r is the frequency error of the receiver.

2.1.1 Knowns, unknowns and assumptions

The channel center frequency f_c is known and shared between transmitter and receiver. The modulation shift f_m is also known – its value is dependent on the physical layer when sending all 1's: 1M has a shift of 250kHz and 2M has a shift of 500kHz .

Three parameters are left as unknown: doppler shift f_d , transmitter frequency error Δf_t and receiver frequency error Δf_r as unknown and time-varying. None of these are directly observable from a CTE. Only the difference between f_t and f_r is observable and is known as Carrier Frequency Offset (CFO).

To estimate CFO some model of how the phase at each antenna should behave during a CTE must be adopted. In our work the assumption is that the phase at each antenna should be constant – this assumes that there is no change in frequency in transmitter or receiver during the CTE and that doppler-shift is negligible.

The bluetooth core specification requires that the frequency can not drift beyond $\pm 75kHz$ [8], which results in a maximum of $150kHz$ with two devices. It is reasonable to assume that the frequency drifts slowly as a random-walk process. The frequency is affected by production variations, age of components and temperature.

2.2 Estimating Carrier Frequency Offset

In [11] a Maximum Likelihood estimator due to [14] was used to estimate CFO using just the reference period, or by using a Return-to-First switching pattern where the reference antenna is sampled every other sample period. A whitepaper by Nordic Semiconductor [15] seems to indicate a similar strategy.

Using just the reference period, which has eight samples of a single antenna, is not a viable strategy and estimations are too noisy. We show with a monte-carlo simulation in figure 2.6 that using just the reference period results in several kilohertz error in CFO-estimate which results in several degrees error as shown in figure 2.2.

In [16] eight different CFO estimators were analysed and compared to each other. The best estimator there was MUSIC, which is a computationally costly method, while the ML estimator due to [14] was the worst. They concluded that a method like the ML estimator with an extension is better than MUSIC due to the complexity of MUSIC

A simple method with good accuracy was presented in [5]: it is a simple cost function, similar to a least squares method, that can take any switch-pattern. They present it in a periodogram form as a beamformer, with a 'steering vector' and a spectrum which when maximised gives an estimate for the CFO. This can be used with just the reference period or with all samples and they demonstrate that using all samples gives the best results.

In our work we will start with the method of [5] as it is the best method due to its flexibility in switch pattern design and the ability to use all available data for the estimation. Taking the method further we will develop a way of quickly finding the optimal value with a cost function and minimization framework instead of the brute force method used in their implementation.

2.2.1 Cost function and optimization problem

The cost function used here is due to [5] with only a sign-change in eq 2.4 since we are in a optimization framework where one looks for minima².

$$\mathbf{a}_k(f) = e^{j2\pi f \mathbf{t}_k} \quad (2.3)$$

$$P_{freq}(f) = - \sum_{k=1}^m |\mathbf{a}_k(f)^H \mathbf{x}_k|^2 \quad (2.4)$$

where \mathbf{t}_k is a column vector of the relative measurement times for antenna k , \mathbf{x}_k is the subset of the measurement vector \mathbf{x} with measurements from antenna k and $(\cdot)^H$ is the Hermitian transpose.

The optimization problem to be solved is formulated as:

$$\min_f P_{freq}(f), \quad s.t. \quad 350kHz < freq < 650kHz \quad (2.5)$$

the solution f is the CFO estimate, where we only want values inside the expected range of $\pm 150kHz$ of the nominal PHY-offset; in this thesis the 2M phy is used which has a nominal offset of 500 kHz.

The geometry of this cost-function is dependent on the switching pattern and sampling configuration. The Nyquist-Shannon sampling theorem [17] gives some insight into the issue. We are trying to estimate a value which is expected to be $500.00kHz \pm 150.00kHz$, but the Bluetooth standard allows sampling at $1.00\mu s$ during the reference period and then only at $2.00\mu s$ during the switch-sample period. Because of this undersampling where much of the signal is not uniquely determined there will be aliasing and multiple identical solutions of the cost function at different frequencies.

²Searching for minima is the convention is optimization theory [12]. If one needs to find the maximum of a function one simply negates it before optimizing

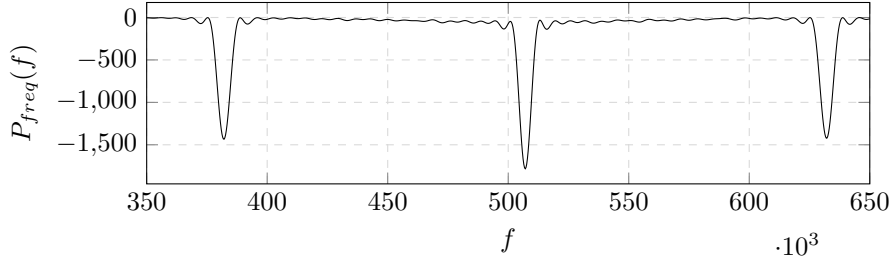


Figure 2.3: Example of a CFO-cost function. This is a linear repeating scan of a four-element array with $1.00 \mu\text{s}$ switch-sample slots. The data here is simulated with no noise and the true CFO value is 508.00 kHz

The radio on the nrf52833, which is used in this thesis, is capable of sampling at $0.25 \mu\text{s}$ and could therefore simplify this optimization routine by simply reducing the length of the CTE. Unfortunately the operating system is designed to follow the bluetooth standard and it would require very low-level modifications to change the samplerate, as such we must (at least in this thesis) make do with the slow samplerate.

2.2.2 Developing equations for calculating cost and derivative of cost

For convenience the cost functions (eq. 2.3 and eq. 2.4) are restated here:

$$P_{freq}(f) = - \sum_{k=1}^m |\mathbf{a}_k(f)^H \mathbf{x}_k|^2, \quad \mathbf{a}_k(f) = e^{j2\pi f t_k}$$

where \mathbf{t}_k is a column vector of the relative measurement times for antenna k , \mathbf{x}_k is the subset of the measurement vector \mathbf{x} with measurements from antenna k and $(\cdot)^H$ is the hermitian transpose.

To aid simplification and derivation of the gradient the vector-based formulas are changed to sums and combined into one equation:

$$P_{freq}(f) = - \sum_{k=1}^m \left(\left| \sum_i (e^{j2\pi f t_i})^* x_i \right|^2 \right) \quad (2.6)$$

where t_i is the relative measurement time for sample i of antenna k , x_i is sample i of antenna k and $(\cdot)^*$ is the complex conjugate operator.

Developing equations for the gradient

The well-known Euler's formula from [3] states that $e^{jv} = \cos(v) + j \sin(v)$ which we will use to remove the exponential in eq. 2.6 – note that in eq. 2.6 the exponential is complex-conjugated so that the \Im term in eq. 2.7 is negated.

We will use $x_i = I_i + jQ_i$ to represent IQ-sample i of antenna k .

$$\begin{aligned} P_{freq}(f) &= - \sum_{k=1}^m \left(\left| \sum_i (\cos(2\pi f t_i) - j \sin(2\pi f t_i))(I_i + jQ_i) \right|^2 \right) \\ &= - \sum_{k=1}^m \left(\left| \sum_i (I_i \cos(2\pi f t_i) + Q_i \sin(2\pi f t_i)) + j(Q_i \cos(2\pi f t_i) - I_i \sin(2\pi f t_i)) \right|^2 \right) \\ &= - \sum_{k=1}^m \left(|g(f)|^2 \right) \end{aligned} \quad (2.7)$$

where $g(f) : \mathbb{R} \rightarrow \mathbb{C}$

For computations it is beneficial to note that

$$|u(x) \pm jv(x)|^2 = \left(\sqrt{u(x)^2 + v(x)^2} \right)^2 = u(x)^2 + v(x)^2 \quad (2.8)$$

The starting point for the gradient is eq. 2.7

The product rule in differentiation yields:

$$\frac{d}{dx}|h(x)|^2 = \frac{d}{dx}|h(x)||h(x)| + |h(x)|\frac{d}{dx}|h(x)| = 2|h(x)|\frac{d}{dx}|h(x)| \quad (2.9)$$

And the chain rule yields:

$$\begin{aligned} \frac{d}{dx}|h(x)| &= \frac{d}{dx}|u(x) + jv(x)| = \frac{d}{dx}\sqrt{u(x)^2 + v(x)^2} \\ &= \frac{1}{2} \frac{2(u(x)u'(x) + v(x)v'(x))}{\sqrt{u(x)^2 + v(x)^2}} \\ &= \frac{\Re(h(x)(h'(x))^*)}{|h(x)|} \end{aligned} \quad (2.10)$$

Eq. 2.9 and 2.10 yields:

$$\frac{d}{dx}|h(x)|^2 = 2\Re(h(x)(h'(x))^*) \quad (2.11)$$

Such that

$$\frac{d}{df}P_{freq}(f) = -\sum_{k=1}^m 2\Re(g(f)(g'(f))^*) \quad (2.12)$$

where

$$\begin{aligned} g(f) &= \sum_i \left[(I_i \cos(2\pi ft_i) + Q_i \sin(2\pi ft_i)) \right. \\ &\quad \left. + j(Q_i \cos(2\pi ft_i) - I_i \sin(2\pi ft_i)) \right] \end{aligned} \quad (2.13)$$

and

$$\begin{aligned} (g'(f))^* &= \sum_i \left[(Q_i 2\pi t_i \cos(2\pi ft_i) - I_i 2\pi t_i \sin(2\pi ft_i)) \right. \\ &\quad \left. + j(I_i 2\pi t_i \sin(2\pi ft_i) + Q_i 2\pi t_i \cos(2\pi ft_i)) \right] \end{aligned} \quad (2.14)$$

2.3 Efficient solution of the optimization problem

This is a one-dimensional problem and it is trivial to find the global optimal by simply evaluating all frequencies within the expected range. In [5] a resolution of 25Hz is used to search for the CFO estimate – which results in twelve thousand queries of the cost function if the whole range of $\pm 150kHz$ is searched.

Searching twelve thousand points is not feasible for real-time operation on a microcontroller³. The cost function 2.4 is fairly expensive to calculate, requiring many cos-, sin-, addition- and multiplication operations. The speed of calculating 2.4 is linearly increased by adding more samples of antennas – potentially being a limiting factor for more advanced sampling-configurations. In a navigation system one would need to process data from multiple arrays at the same time which again places demands on solving problem 2.5 efficiently. Utilizing information about the previous result to narrow the search area is feasible and could give real-time performance, but at the cost of instability. If the initial search is inaccurate, or if there is a span of bad measurements, such that the estimate falls into a wrong local minima instead of the correct minima then the iterative solution will stay in that local minima. Workarounds to widen the search-space to include neighbouring local minima could work at the cost of slowing the method down. It will likely be difficult to prove stable and consistent optimization with such a strategy.

Instead of brute-force one can use the field of numerical optimization. Numerical optimization strategies typically work by searching in a 'downhill' direction to find the lowest value of the cost function –

³Even on my laptop it is frustratingly slow to analyze logs with such a brute-force search

imagine a ball rolling down a hill. Because of this downhill search strategy they are not guaranteed to find a globally optimal value, just the lowest value in the current basin. Rapid inspection of 2.3 shows that one must be within just a few kilohertz of the true value to be able to use an optimizer and get the correct result⁴.

We can't find the optimal value of the full cost function without having an approximate solution, but if we can find an approximate solution we can use that result to prime the full cost function. If this approximation is a simple function that is easier to solve, and as long as the result from the simpler cost function give estimates good enough to fall within the global minima then the problem can be solved efficiently. There is an obvious choice for such a function: the reference period contains samples of one antenna sampled at $1\ \mu\text{s}$ and is easier to optimize due to its simple structure as shown in figure 2.4.

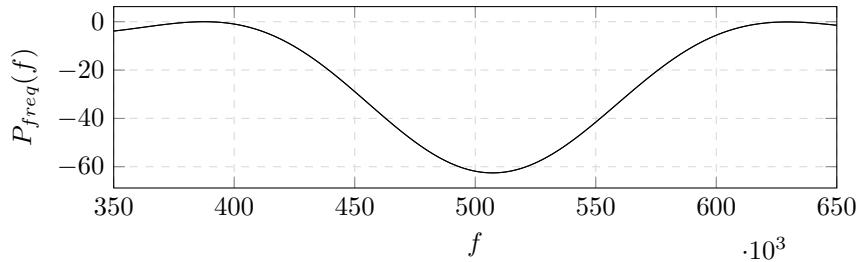


Figure 2.4: CFO cost function of just the reference period

Simulations show, see figure 2.6, that the results from the reference period alone is too noisy to start the optimizer at the cost function with all samples, therefore a second function was designed: ref-ext (reference period extended). This function takes samples of the reference antenna from the sample-period of the CTE: up to $24\ \mu\text{s}$ ⁵ – such that the total length of the period used in this cost function is $33\ \mu\text{s}$. This function has multiple minima, but the optimal value is more stable than the reference period and the basin is wide enough such that the reference period is accurate enough to approximate the ref-ext solution.

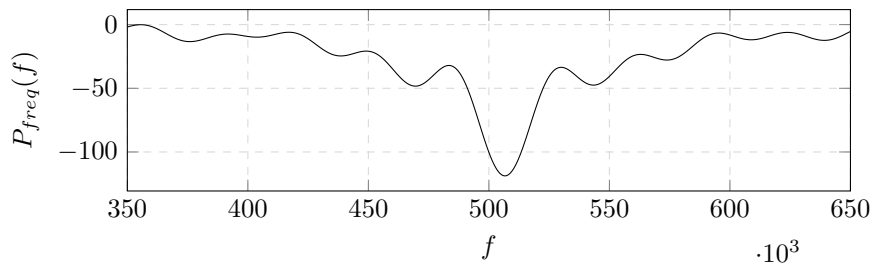


Figure 2.5: CFO cost function for the extended period with a linear scan of a four-element array

2.3.1 Implementing the estimation strategy

To solve each optimization problem we need some algorithm. This is a one-dimensional search space, and line-search methods are applicable. One could develop a suitable algorithm from [12] or other sources to run well on a microcontroller, but this is not pursued here due to time constraints: for the purposes of this masters the optimization tools in the GNU Scientific Library (GSL) [13] were used.

GSL implements several optimizers but in this work the BFGS2 implementation with default parameters is best suited. BFGS2 is optimized to solve large search spaces and several techniques are employed to make that efficient [12][13], but we are really only using the line-search part of the optimizer and much simpler algorithms could be used for this problem.

⁴There are optimization strategies that are designed to solve problems that have multiple minima, some have cool names like Simulated Annealing, but they are likely to be overkill in this situation

⁵There is some dependence on the switching pattern here, if the antenna does not come up during this period then ref-ext is identical to ref

BFGS2 is an optimizer which uses the gradient of the cost function to solve the minimization problem faster. One could use a numerical library to approximate the gradient but it is faster to calculate the gradient directly – to enable this equations for the gradient of the cost function 2.4 are developed in section 2.2.2.

Additionally BFGS2 takes advantage of the fact that the cost function and gradient of the cost function often can be calculated together faster than individually.

Trigonometric calculation performance

Calculating the cost and derivative of the cost using equations outlined in section 2.2.2 require many calculations of sine and cosine terms. They are unique to each f and unique to each sampletime t_i within each CFO. Trigonometric functions are costly and is a significant contributor to the total running time of the CFO-estimation procedure. Can we do better? It turns out, yes – we can!

In [5] a method of efficiently generating the steering vector⁶ for an array was presented. It works by sequentially adding the phase-shift for each successive antenna rather than calculating the phase-shift anew, this method reduces the number of trigonometric calculations by 75% for a 4x1 linear array. We apply this method for CFO-cost function as follows:

Each CTE is divided up into n slots at predetermined time $t_i \forall i \in n$. The time between two successive samples is $1.00 \mu\text{s}$ during the reference period and $2.00 \mu\text{s}$ or $4.00 \mu\text{s}$ during the sampling period depending on the PHY⁷. One can initially calculate two complex numbers: $e^{2\pi f 1 * 10^{-6}}$ for the time difference in the reference period and $e^{2\pi f 2 * 10^{-6}}$ for the time difference in the sample-period.

The complex exponential terms for the reference and sample period are then:

$$e_{ref_0} = 1, \quad e_{ref_i} = e_{ref_{i-1}} \cdot e^{2\pi f 1 * 10^{-6}} \quad (2.15)$$

$$e_{smp_0} = e_{ref_7} \cdot e^{2\pi f 2 * 10^{-6}}, \quad e_{smp_i} = e_{smp_{i-1}} \cdot e^{2\pi f 2 * 10^{-6}} \quad (2.16)$$

The required sine and cosine terms are now, thanks to Euler's formula, found in the real and imaginary parts of the exponentials⁸.

With a complete CFO with 82 samples (both counting reference period and sample period) this method can reduce the number of trigonometric evaluations by $\approx 97\%$. One should take care to evaluate if the numerical precision is good enough, particularly if single-precision is used – the Floating Point Unit on the Cortex-M4 processors⁹ run on single-precision floats. Drift can be reset by calculating the exponential for t_i and then proceeding with this method.

Queries of the cost function

The point of using optimization strategies is to limit the number of queries of the cost function, but how many queries does each method use?

We evaluate this by counting each call to each cost function and their derivatives. We compare the brute force method used in [5] with a step-size of 25, the same method but with a step size of 500 and finally the multimethod based on three iterations of cost function-optimization using BFGS2.

The data used to evaluate is a dataset collected on-site, and contains 17023 samples, and should therefore give a nice average evaluation-count for each method. This is important because optimizers do not use a fixed number of iterations since they sometimes have faster or slower convergence based on the problem at hand.

The range of frequencies tested by the brute force methods is $\pm 100 \text{ kHz}$, which is similar to the range that the multimethod can handle due to the width of the cost function with the reference period as shown in 2.4.

⁶The steering vector is a term in beamforming and is used for direction estimation, see chapter 3

⁷We will proceed the derivation with 2M PHY

⁸C implements complex numbers by two numbers that represent the real and imaginary part, so accessing real and imaginary components are instant

⁹nr52 is based on a Cortex-M4 processor

¹⁰Without iterative exponentials

¹¹On my laptop

	Brute 25	Brute 500	multimin
all	1.36×10^8	6.83×10^6	1.60×10^5
df all	-	-	1.26×10^5
ext	-	-	8.74×10^4
df ext	-	-	8.59×10^4
ref	-	-	1.18×10^5
df ref	-	-	1.15×10^5
Total evaluations:	1.36×10^8	6.83×10^6	7.43×10^5
Trig calls ¹⁰	1.12×10^{10}	5.60×10^8	3.07×10^7
Execution time: ¹¹	3.75×10^2 s	1.90×10^1 s	1.22 s
per second:	4.53×10^1 s	8.95×10^2	1.40×10^4

Table 2.1: Queries of the cost function per method. The brute methods find the optimal of the cost function using all data with stepsize 25 or 500. The execution time is how long it took my laptop to solve 17023 samples

The multimin method is significantly faster than the brute force methods: around one order of magnitude faster than brute with 500 hertz stepsize and around three orders of magnitude faster than brute with 25 hertz stepsize.

This performance increase will be important for real time operation on embedded hardware.

2.3.2 Accuracy of the estimation

To evaluate the accuracy of different estimators a monte-carlo simulation was developed. It generates IQ-samples following code shown in listing 2.1, estimates CFO using a variety of estimators and calculates CFO-estimation-error and angle-estimation-error in RMSE values.

Note that this simulation assumes white-noise in phase-measurements of each antenna, but this is not necessarily the case for real data – figure 2.8 suggests that indeed IQ-measurements are not always corrupted by white noise. Nevertheless, simulations with white noise are suggestive in the performance of the estimator.

Listing 2.1: IQ generation procedure

```

double complex IQ[switchpatternlength];
double ran_cfo = gsl_ran_gaussian(cfo_sigma);
for(int i = 0; i < switchpatternlength; i++){
    // Calculate the 2-norm distance from receiver and transmitter
    // element in use for sample i
    double d = dist(receiver, transmitter[switchpattern[i]]);

    // Simulate phase at distance d, t=0
    // Add noise to samples, to simulate noise from antenna, filter
    // and analog-to-digital converter
    double _I = sin((2*PI/lambda)*d) + gsl_ran_gaussian(r, iq_sigma);
    double _Q = cos((2*PI/lambda)*d) + gsl_ran_gaussian(r, iq_sigma);

    // Convert to complex number
    IQ[i] = Q + I*_I;

    // Add CFO-effect based on sampletime of sample i
    IQ[i] *= cexp(-2*PI*I*ran_cfo*sampletime[i]);
}

```

A realistic value of σ for the IQ noise is around 0.1. This value is based on estimation from real measured data, but is not a very accurate estimate. The true value will vary with the signal strength which is affected by distance and reflections.

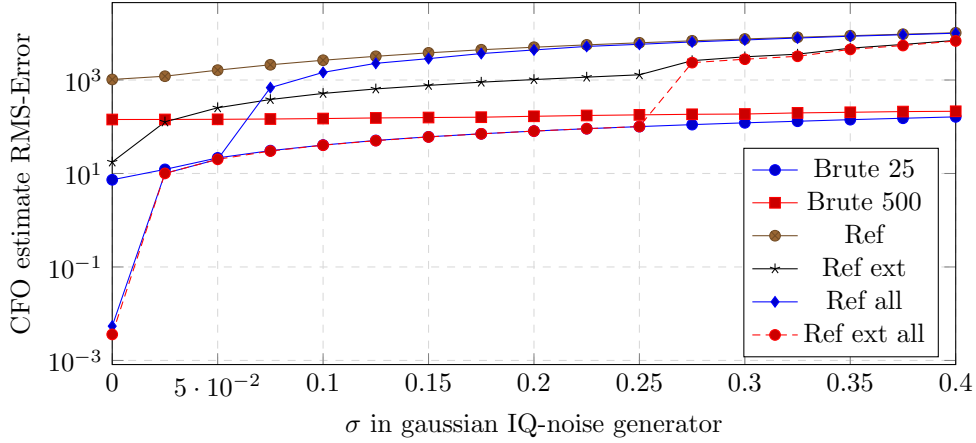


Figure 2.6: Monte-carlo simulation of CFO estimation performance with different levels of noise on the IQ-measurement. Brute n is the brute-force calculation using stepsize n . The others are BFGS2 with different cost function selections. Ref all shows that the reference period alone is too noisy to correctly prime the full estimator, and at around 0.27 the ref ext all also suffers.

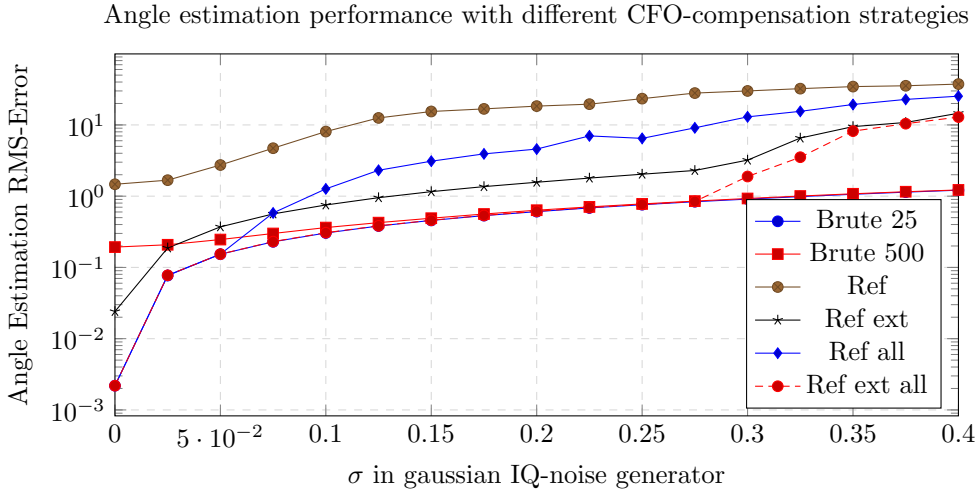


Figure 2.7: Evaluation of angle-estiation RMSE with different strategies and levels of noise in the IQ-measurement.

An important result shown in figure 2.7 is that a multimim strategy must use a mutli-step approach to deal with noise in simpler functions, and ref-ext-all is stable up to 0.27 which is good enough for real data processing. The figure also demonstrates that if one must use a brute force approach it is overkill to use a stepsize of 25 and that 500 is a more appropriate balance of accuracy and performance.

2.4 Impact of reflections on the CFO estimation

Assuming reflections follow a simple ray-tracing model then the superposition principle can be used to investigate the impact of reflecitons on the CFO-estimation step.

A wave and its n reflections are received at an antenna such that the recieved signal is:

$$A(t) = \sin(2\pi ft) + \sum_i a_i \sin(2\pi ft + \phi_i) \quad (2.17)$$

where a_i is the amplitude of the reflected components. The output of the IQ-sampling process (see figure

1.4) is the signal:

$$I(t) = \frac{1}{2} \left[\cos((\omega_1 - \omega_2)t + \theta) + \sum_i a_i \cos((\omega_1 - \omega_2)t + \theta - \phi_i) \right] \quad (2.18)$$

where θ is the offset between the local oscillator and the line-of-sight component, while ϕ_i are the phase-shifts due to the length-difference of each reflection.

Because of equation 1.4 any reflection with the same frequency will not affect the CFO-estimation step which will happily find the frequency $\omega_1 - \omega_2$ regardless of the number of in-frequency reflections. But what happens if there are components that are *not* the same frequency?

Imagine a reflection causes the signal to change frequency by some small amount such that the received signal is:

If there are multiple signals received at nearly the same frequency, then the output of the IQ-process is a sum of sinusoids of different frequencies. For instance, if there is a reflection that is offset by 1.00×10^1 kHz then the output of the IQ-sampler is:

$$A(t) = \sin(2\pi ft) + \sum_i a_i \sin(2\pi f_i t + \phi_i) \quad (2.19)$$

where a_i is the amplitude of the reflected components and f_i is the frequency of each component.

If the frequencies f_i are close enough to the channel center frequency f_c then they will pass through the channels bandpass filter and be a part of the IQ-downconversion. The output of the IQ process is then a sum of sines of different frequencies and the CFO estimation step will attempt to keep the average to zero but it could, depending on the amplitudes a_i and frequencies f_i , result in significantly affected phase-measurements. Some of this can be detected using outlier rejection and is explored in the next section.

2.5 Outliers

There are some cases where the CFO-estimation strategy fails; figure 2.8 shows one such example. Here the multimethod strategy and the brute force method failed – failed as in the estimate was far away from surrounding estimates. But one can hardly blame the optimization strategy for failing to estimate CFO here, as the data is obviously erroneous.

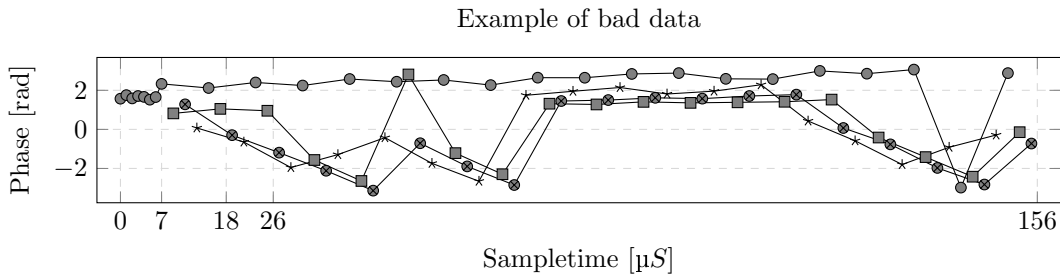


Figure 2.8: An example of bad data. This CFO packet (from real data) with 1us default switch pattern. Neither multimethod nor brute found a good solution. This is compensated for the CFO estimated at the previous packet to get a corrected output.

The cause of bad data may be off-frequency reflections explored in the previous section, a null in the RF-reception, disturbance from other transmitters or some other cause. Removing data like this is important to keep accuracy high.

2.5.1 CFO outlier-rejection strategy

Based on the model of CFO outlined in section 2.1 we can determine a strategy for outlier rejection. The key idea is that CFO values should not change significantly over a short period of time, and if the CFO estimation suddenly is far from previous estimates it is likely to be due to bad data¹².

¹²Example 2.8 was found due to a large offset from surrounding CFO estimates.

In [18], chapter 9, some strategies for outlier rejection in streaming data are outlined. For CFO the simplest method can be used: a distance-based windowed average strategy.

We create a First-In-First-Out (FIFO) queue with a maxlength of 10. At the end of each CFO estimation routine the estimate is compared to the median of the 10 previous CFO estimates in the queue and if it more than 500 hertz away is it rejected.

It is important that even rejected values are pushed into the queue to ensure that the rejection procedure is stable even if there is a significant time-delay between samples such that the value changes significantly or if there is truly a step in the CFO value. A robust estimator such as median should be used to avoid large outliers pushing the value away such that all subsequent values are rejected.

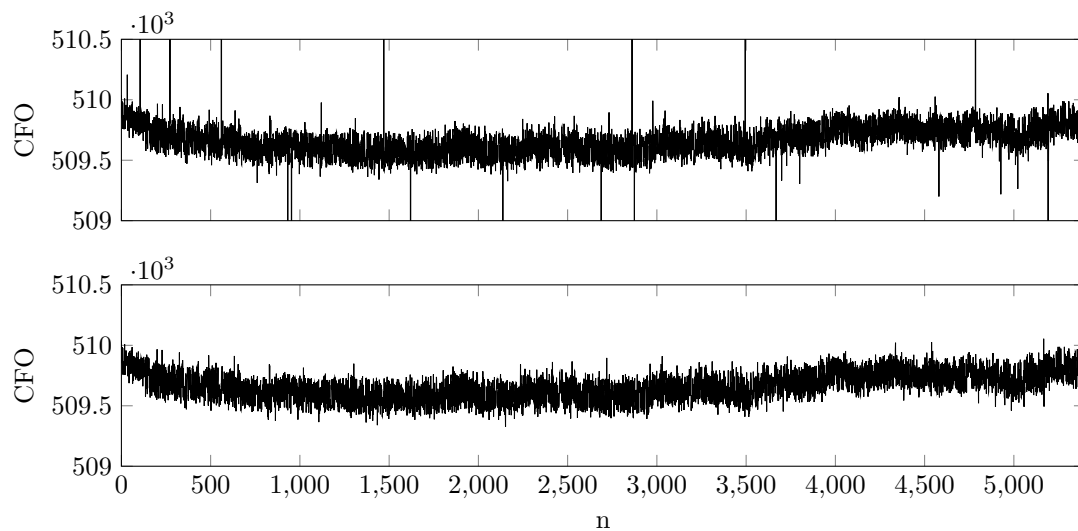


Figure 2.9: Before and after outlier rejection on real data

2.6 Switch-pattern dependence

In this chapter we have only shown results with a 4x1 antenna with a switching pattern: 1,2,3,4 repeating.

Significant efforts were directed at investigating relation to switch-pattern and angle estimation performance. It is clear that with a good CFO-estimation strategy the specifics of the pattern are irrelevant. The CFO cost function may have more or fewer optima, but the three-step method of optimization ensures that only the global optima is selected. As long as the cost function for the reference period and ref-ext are intact the other sample-periods are irrelevant. After a sample has successfully been compensated for CFO there is little to no influence in direction estimation.

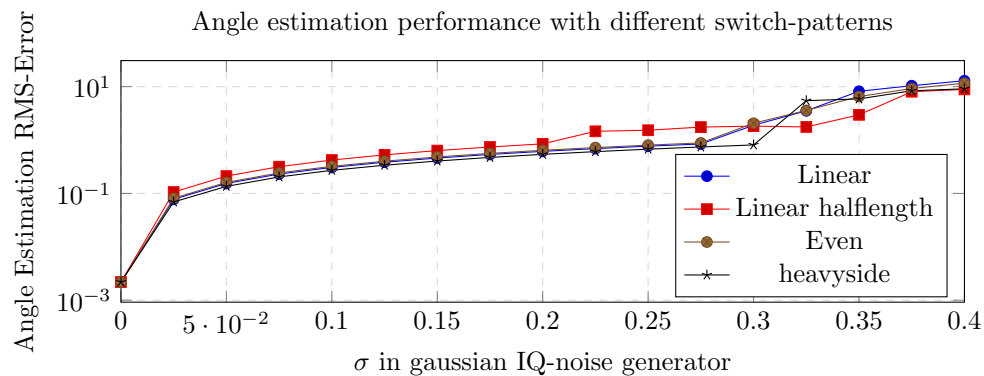


Figure 2.10: Estimator is invariant to different switch patterns. Small variations are due to the exact length of ref-ext. The halflength is slightly less performant, though more stable, due to it's total duration being half the length of the others. Even and heavyside are different variations on the 1,2,3,4 pattern distributing the numbers of samples evenly or more towards the sides of the array.

Chapter 3

Direction estimation

In chapter 1 the basic working principle of direction finding was explained. An array of antennas, as shown in figure 1.3, samples the same signal at different locations. Differences in their location in reference to the incoming plane-wave results in different phase-shifts of the received signal and by measuring this phase-shift¹ one can determine the direction of the incoming plane-wave.

The predicted phase at each antenna can be calculated from the arrays steering vector – the name comes from beamforming where one ‘steers’ radiowave-power in certain directions by transmitting simultaneously from multiple antennae at different phase and amplitude. The word ‘simultaneous’ is critical: it is the fact that multiple radiowaves *combine* that directs the power. Bluetooth is not capable of beamforming because each individual antenna is only active individually, but the math is the same and so we keep the terminology.

This chapter begins by selecting and implementing an algorithm for direction estimation from phase measurements, before moving on to simulate the effect of reflections to investigate a multi-frequency approach.

3.1 Direction Finding Algorithms

Several algorithms are found to be used in the research published on BLE Direction Finding. They are either very simple – [16] describes theirs as ‘Naive’ – or more involved but often used MUSIC [6][16][11].

In [5] a conventional beamformer is employed and they argue that MUSIC and ESPRIT are not suitable for BLE Direction Finding because there is only one signal and the computational simplicity of conventional beamformers make them more viable in embedded solutions.

In [6] several different DF techniques were considered and MUSIC was deemed inappropriate and is in fact worse than PDDA in all cases.

It is uncertain whether or not this complexity helps for the array styles used for bluetooth direction finding. [16] Finds that their ‘Naive’ algorithm (which is close to a conventional beamformer) performs better than MUSIC. [6] compared PDDA, MUSIC, SSS and ESPRIT – in their comparison PDDA outperforms the other algorithms in almost all cases and they conclude that it is the best for this application. PDDA from [19] is mathematically equivalent to a conventional beamformer, except that PDDA does some further processing which is not necessary for our implementation.

Because of the results in [6] and [16] we will use a simple classical beamformer strategy.

3.1.1 Frequency diversity

Bluetooth Low Energy has 40 channels, but the use of them varies in the literature. In [6] they point out the importance of using several channels, and combine the three advertising channels to minimize

¹A difficulty in measuring phase is Carrier Frequency Offset, which if not corrected will significantly affect direction estimation as is evident in figure 2.2. The entirety of chapter 2 is devoted to the correction of this error, and in this chapter we assume phase-measurements are received and corrected for CFO.

the effects of reflections – in [16] they also achieve better results by combining channels, but it is not as deliberate as [6] and they do not explicitly point out that they combine channels due to their different reflection characteristics.

Frequency is one of the key factors mentioned in wireless channel interference in [9] along with time, translation and orientation of receiver and transmitter as well as their individual antennas. As described in [9] there is a pesimistic and a optimistic way of thinking of these dependencies. The pessemistic view is that there are too many variables to even begin to model and understand all the dependencies. The optimistic view is that knowing at least some of the patterns of the dependencies can help in designing a more robust system.

We attempt to do just that: investigate some pattern on the dependency of frequency and position for an indoor situation.

3.2 The classic beamformer for direction finding

3.2.1 Cost function

The classic beamformer is a simple method of direction finding. Here we borrow notation from [5]. A 'steering vector' is used to calculate the phase-shifts between each antenna element to steer the beam to some direction. In this thesis the array is a uniform linear array and the steering vector can be expressed as done in [6]

$$\mathbf{a}(\theta) = [e^{-j\beta \sin(\theta)p_1} \quad e^{-j\beta \sin(\theta)p_2} \quad e^{-j\beta \sin(\theta)p_3} \quad e^{-j\beta \sin(\theta)p_4}]^T \quad (3.1)$$

where $\beta = \frac{2\pi}{\lambda}$ and p_i is the position of antenna i [6]. This steering vector is just a generalization of equation 1.2.

We use the beamformer pseudospectrum from [5] with slight notation change to fit our 2D scenario and minimization framework:

$$P(\theta) = -|\mathbf{a}(\theta)^H \mathbf{x}|^2 \quad (3.2)$$

where \mathbf{x} is the measurement vector and $\mathbf{a}(\theta)$ is the steering vector.

To combine multiple measurements we simply extend the measurement vector and steering vector to include all measurements. Computationally this can be done by simply summing up the cost function for each measurement.

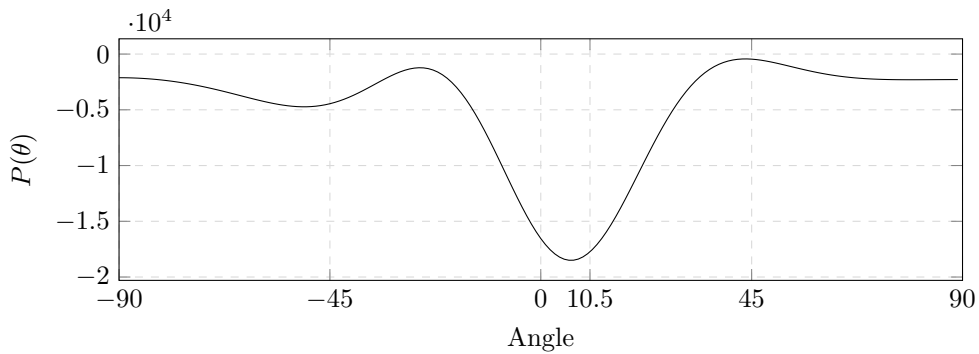


Figure 3.1: Example of a spectrogram for a 4x1 linear array. This is real data collected at the AutoStore test site. Expected angle is 10.50° and it is clear that a successful minimization strategy yields the correct answer

As done with CFO-estimation in chapter 2 we define the beamformer as a cost function which we will minimize to find the optimal value which will – assuming the beamformer works – be the direction of the signal.

$$\min_{\theta} P(\theta), \quad s.t. \quad -\pi \leq \theta \leq \pi \quad (3.3)$$

Efficient generation of steering vector

In [5] an efficient method of generating the array steering vector was presented². Here we present the same method, but in the one-dimensional array case and with our notation:

The phase difference between two adjacent antennas is:

$$a_x(\theta) = e^{j\beta \sin(\theta)\Delta x} \quad (3.4)$$

where β is defined in eq. 3.1 and Δx is the distance between two adjacent antennas, which in our case is 5.00 cm.

The steering vector can now be calculated as follows:

$$\mathbf{a}(\theta)_0 = 1 + 0j, \quad \mathbf{a}(\theta)_i = \mathbf{a}_{i-1}(\theta) \cdot a_x(\theta) \quad (3.5)$$

Doing this reduces the number of trigonometric calculations by 75% for our 4x1 linear array.

3.2.2 Deriving the derivative of the cost function

Deriving the derivative of the cost function was done in the same way as for Carrier Frequency Offset since the cost functions are very similar, see chapter 2.2.2.

3.2.3 Implementing optimization strategy

Similar to CFO estimation we use BFGS2 from [13] to optimize cost function 3.3. But in contrast to CFO estimation, where we develop a strategy to find the global minima, we care about each local minima – because, as discovered in the project thesis and which will be apparent in simulations later, several local minima may be the correct measurement so they all need to be considered by the tracking system.

To find all minima in the spectrum we exploit the limited resolution of the array – there will never be *many* solutions, typically around three to five. The full spectrum to search is 180° , and if we optimize at guesses $-90^\circ + 30^\circ \cdot i$, $\forall i = 0, \dots, 6$ and remove any duplicates the possible directions are found.

3.3 Reflection simulation – frequency and position

Reflections is a considerable issue for an indoor industrial environment. For an AutoStore system there are many potential sources of reflection: the grid structure itself along with the walls are made of aluminium, every robot is a big wall of aluminium that moves and warehouses are generally a big box that has plenty of multipath potential.

Modelling the environment – to some degree of accuracy – is certainly possible, but not as a part of this thesis. Nevertheless we want to investigate some simple reflections to characterize the behaviour and look for potential sources of improvement. To this end a simple ray tracing model was developed which is a method where radiowaves are modelled as particles and is a common method for simple modelling of radio-wave reflections [20][2][9]. Our model will only have the transmitter, receiver and fixed points that act to reflect the signal.

A ray is a path that the radio wave takes from point A to point B. The phase of the signal at the received location is given by the distance $A \rightarrow B = d$ and equation 1.1 with $t = 0$. A ray might reflect off some point, C , which extends the distance traveled to $A \rightarrow C \rightarrow B = d$. Multiple rays combine at the receiver following the superposition principle: they are linearly added together.

In the model we place the transmitting array at the origin and two reflectors at $(-4, 5)$ and $(5, 3)$ (which corresponds to angles -38.60° and 68.20° relative to the array) which reflect $\alpha \in \mathbb{R}^2$ amount of the power – we do not model power loss due to distance. We place the receiver at a line out from the array at 10.00° at varying distances.

²We adapt the same method for CFO, see section 2.3.1

3.3.1 Single reflection and frequency dependence

Lets start with a case where everything works as expected, and where the spectrum looks similar to the example in figure 3.1: We activate the reflector at -38.60° with varying reflection coefficients. Figure 3.2 shows the beamformer spectrum and it is clear that the beamformer always detects both signals and that the LOS component dominates whenever the reflection coefficient is below 1.

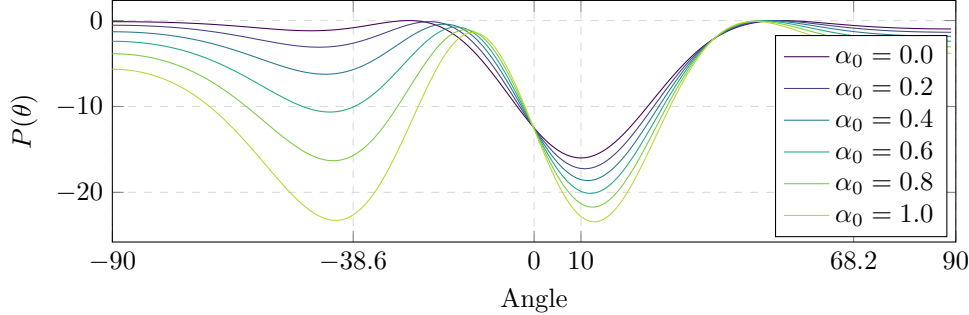


Figure 3.2: Spectra for channel 0 at distance 10.00 m with angle 10.00° , second reflector inactive $\alpha_1 = 0$

Applying a standard beamformer to this scenario will always estimate the correct direction, and the correct direction dominates so long as the reflection has less power than the reflection.

If we were hasty we might conclude that everything is fine and that we can simply apply the beamformer and go along our merry direction finding way... but we must not be hasty!

The nice behaviour of the beamformer shown in figure 3.2 does not hold for all channels. In figure 3.3 we set $\alpha_0 = 0.8$ and vary the channel: for channels 20-25 there is a warp of the spectrum, and the beamformer would indicate that the direction of the signal is at 0.00° – which is not the case for either the LOS signal nor the reflected signal!

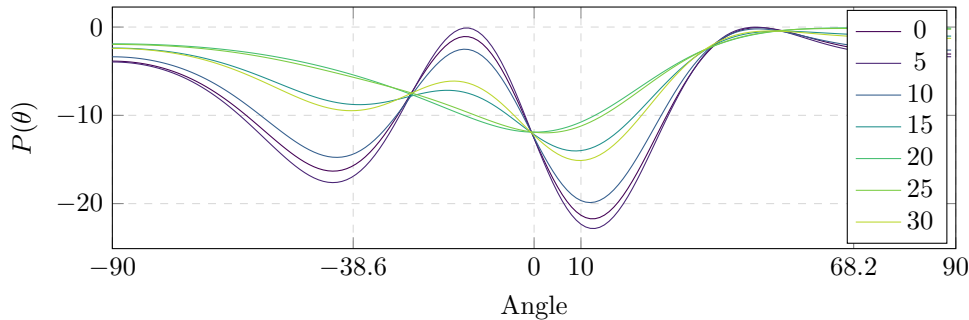


Figure 3.3: Spectra for varying channels at distance 10.00 m with angle 10.00° , second reflector inactive $\alpha_1 = 0$

To investigate what causes the shift we can compare the received signal vs. what would have been received without the reflection source – this is done in figure 3.4.

There is a periodic error in the phase of the signal, compared to the case without the reflector, and the magnitude of the signal at each antenna oscillates: the maximum value is 1.8 ($1+0.8$) and minimum is 0.2 and due to either constructive or destructive combination with the reflected signal.

We can investigate what happens in figure 3.3 at channel 20 and 25 using figure 3.4. At channel 20-25 the phase error of antenna 1 and 4 is small, but the magnitude of the signal is minimal – this means that the array is practically just antenna 2 and 3 because the natural weight of the beamformer³. There is a phase error for both of these antennae, but it is almost symmetrical and the result is that the signal looks like it came from 90.00° degrees.

³The classic beamformer we implemented uses the power of each signal, and therefore stronger signals more influential

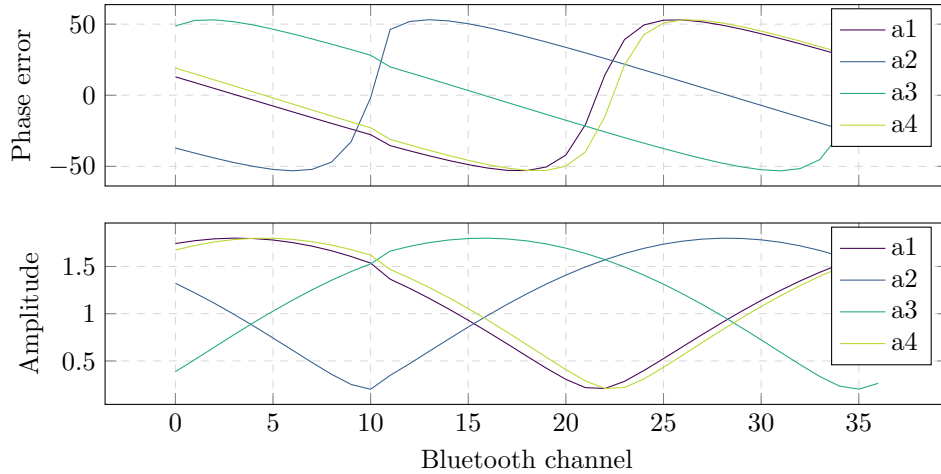


Figure 3.4: Comparing phase to the ideal case without reflection, and signal amplitude at each antenna. Depending on the channel in use the signal is either constructively or destructively combined with erroneous phase-changes. It is a coincidence that antenna 1 and 4 follow each other closely – it is due to the fact that the relative distance between each element, as seen from reflection point 0 is close to a wavelength.

It is scary to attempt to come up with strategies to fix this issue. Its clear that antenna 1 and 4 are actually correct, but ignoring antenna 2 and 3 and trusting two antennas which now have low signal-to-noise ratio is risky. How would one even know that antenna 1 and 4 are correct?

Another, more sane strategy, is apparent from figure 3.4: combine channels! If measurements are combined from every channel the average power goes to 1 and the average phase error goes to 0. It is clear that the power is maximal when the phase-error in the antenna is 0, which gives a nice natural weighted average.

It is interesting that there is a full period of the pattern for the 37 channels – but this is not the general behaviour and is due to the constructed nature of this example. The LOS path is 10 meters, while the reflected path is 13.9108 meters – changing from channel 0 to channel 37 changes the wavelength just enough so that a whole integer more waves have reached the receiver (one more for the LOS signal and two more for the reflected signal). Because of this, it is a perfect strategy to combine channels and the output of the beamformer is now perfectly consistent as shown in figure 3.5.

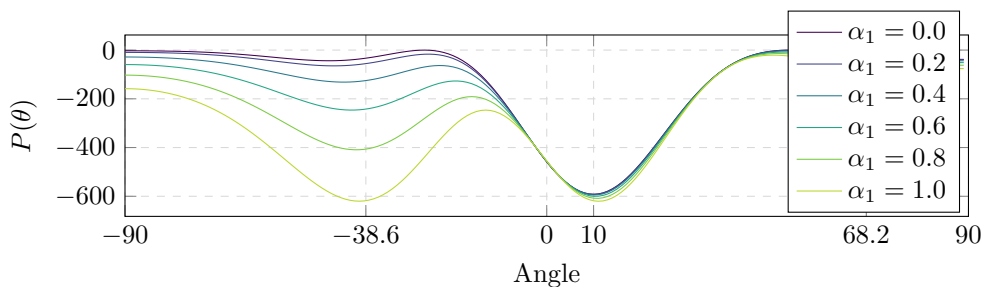


Figure 3.5: Combining all channels to get a single output. The estimator has correctly measured both signals

It may seem that figure 3.5 is the same as 3.2, but that was just because the channel used in that plot happened to be a 'good' channel and is by no means guaranteed – using all channels guarantees (at least in this case where the reflection-geometry is such that changing channels covers all reflections) that the output is consistent.

3.3.2 Single reflection and position dependence

In the same way that changing frequency changes the errors, the same happens with position changes. Similarly to how a slight change in wavelength changes each path differently, moving changes the total length of each path differently.

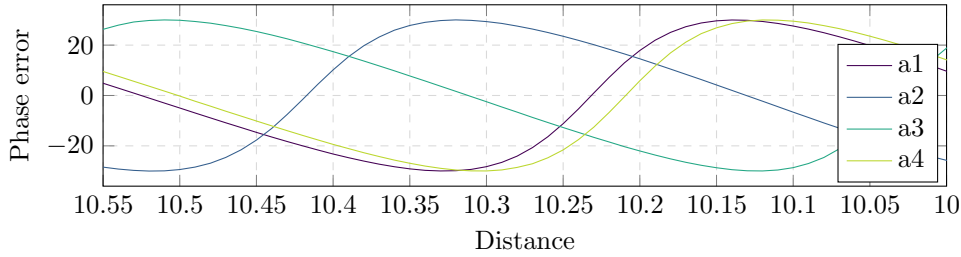


Figure 3.6: Position dependence on the phase error. In much the same way as frequency moving in space changes the reflection characteristics.

This graph is so similar to 3.4 because the mechanism is very similar. Each path changes its length at a different rate due to a position change, and so the reflections change their behaviour in much the same way as when changing frequency.

In this case a 0.55 m change in in position perfectly captures the reflections and results in a consistent beamformer. But this is highly position dependent. Moving closer to the receiver results in higher-frequency reflection dependence, and moving further away reduces the frequency.

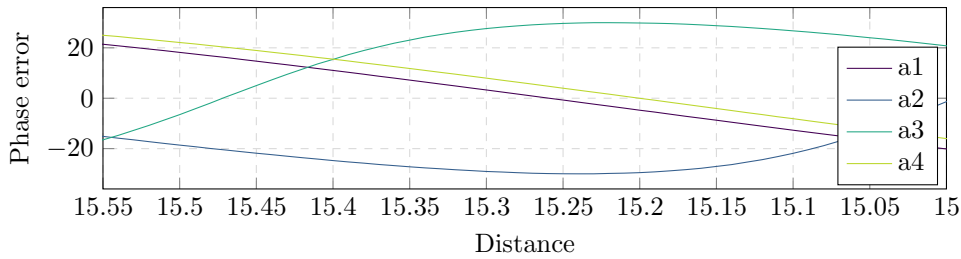


Figure 3.7: Position dependence on phase and amplitude error at a greater distance, which results in less difference between reflected and LOS path and thus lower frequency in the position dependence

3.3.3 Multiple reflections

A single reflection will never overpower the LOS component, but what about multiple reflections? Indeed in certain scenarios, two reflectors – each with less power than the LOS signal – can dominate the LOS signal and result in an erroneous result.

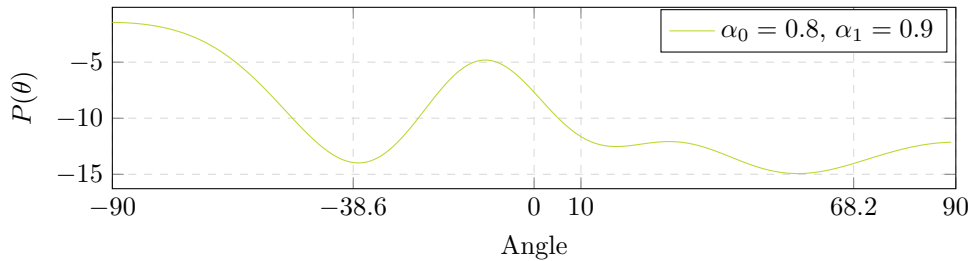


Figure 3.8: An example of a case where the reflections have lower power than the LOS component but where the reflections act on the array in such a way to cancel some of the LOS power and result in an erroneous result

As with single reflections the characteristics are frequency dependent:

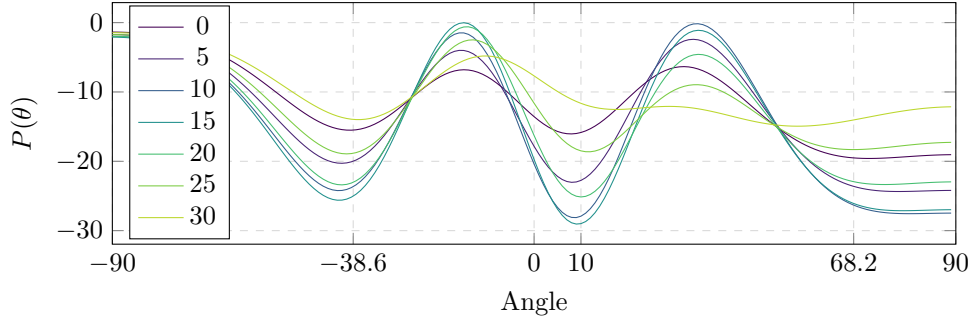


Figure 3.9: Spectra for channel 0 at distance 20.00 m with angle 10.00°, second reflector inactive $\alpha_1 = 0$

But critically, in contrast to the single-reflection case, using multiple frequencies does not in this case yield an unbiased result. We can investigate this by looking at the phase and amplitude errors as we did with the single reflection case. It is evident from 3.11 that the period of the reflection pattern is too long for the range of the BLE frequency spectrum to correct, as such – in contrast to the single reflection case in 3.4 – combing all channels does not correct the issue.

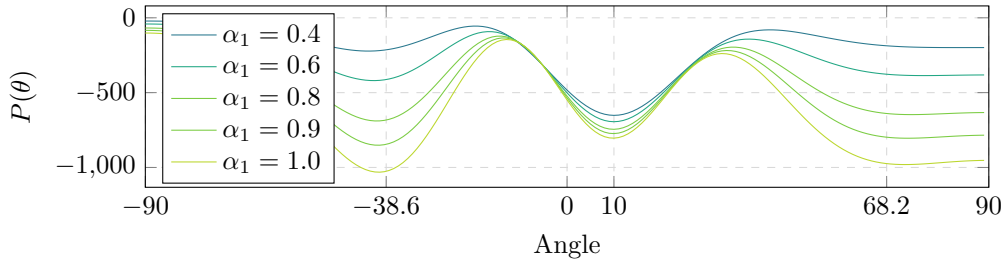


Figure 3.10: Combining all channels into a single beamformer spectrum. When the reflections are more than 80% they overpower the LOS signal and thus simply combining all channels does not 'fix' the issue. A multiple peak strategy would find the correct direction, but this hints that it is not always sufficient to just use multipel frequencies.

3.3.4 Combining frequency and position dependence

If using all BLE channels is not enough to correct for reflection errors one could also use multiple positions – errors are also affected by position as shown in 3.6. In figure 3.12 both multiple frequencies and positions are used: the receiver is sampled at 5 cm intervals in a 0.50 m, 37 channels are used and all data is combined into one spectrum.

3.4 Switch-pattern dependence

Assuming that the Carrier Frequency Offset estimation and compensation step is performed well there is little impact of different switching patterns.

If the only cause of error was white noise in IQ-measurement then a good strategy would be to sample the edge-antennas more, as they have the biggest influence on the direction estimation and more samples increases signal-to-noise ratio in white-noise situations.

However, in real settings, white noise in IQ-measurements are *not* the primary cause of error – rather reflections are the primary issue. In this case it is a better strategy to sample each antenna evenly so that all antennas have a good signal-to-noise ratio and the natural weighing in the phasepower pattern is optimal.

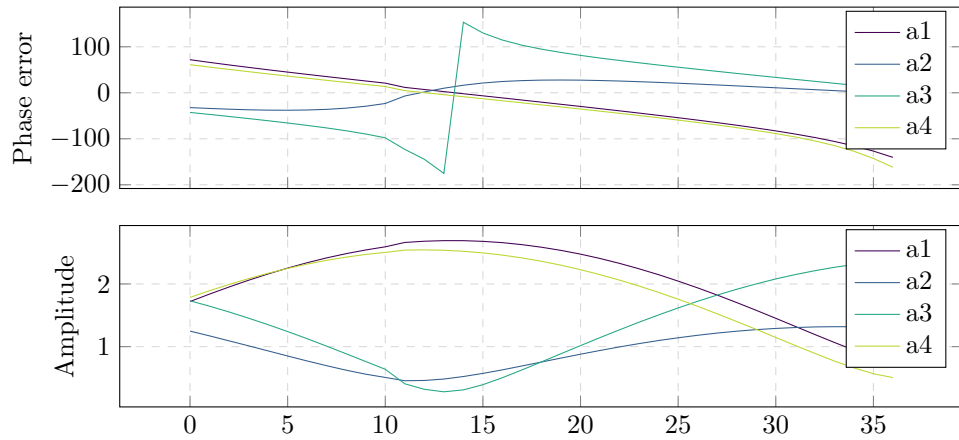


Figure 3.11: Phase-error and signal amplitude with different frequencies in this multi-reflection scenario. It is clear that the range of frequencies covered by Bluetooth is not sufficient to correctly measure the whole reflection pattern and therefore the estimate will not be consistent. It is still better to use multiple frequencies, but it does not solve the problem as well as for the single-reflection case.

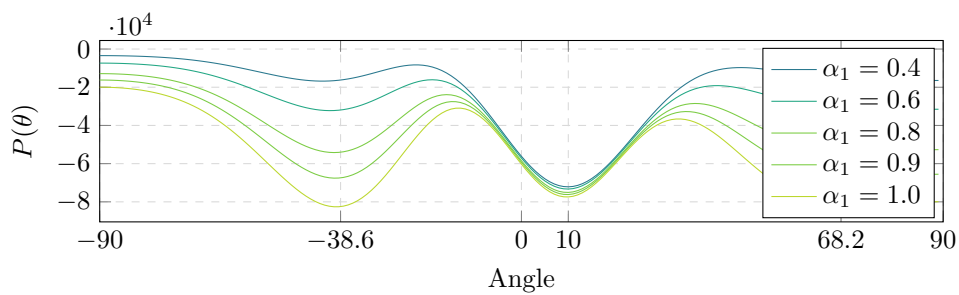


Figure 3.12: Using both multiple frequencies and multiple positions to reduce the impact of reflections. With this strategy the beamformer has correctly identified the correct direction and given it the most power so long as the reflections are weaker than 0.9 – which is much more consistent than using a single position where multiple frequencies were unable to account for all reflection geometry.

Chapter 4

Experimental setup

AutoStore is an automated storage and retrieval system (ASRS). The system uses robots to pick and place boxes in a cube structure. The cube consist of cells which are 50.50 cm wide and 70.50 cm long. The robots navigate on this structure by counting cells – they do not need an indoor positioning system for normal operation, but is a perfect testbed for one: AutoStore allows easy movement of antennas in a large space with decent accuracy¹.

AutoStore is a suitable testbed for an indoor positioning system because the robots are easy to control and have precise localization due to the construction of the grid surface – this allows measuring hundreds of locations with a known location.

In this work a robot will be equipped with an antenna and log CTE samples from three arrays placed around the grid. The true position of the robot is synchronized and logged into the same system.

4.1 Placement of arrays at the test-site



Figure 4.1: Robot with antenna receiver. The robot is currently on cell (39, 8) and array F is visible on the far end and array D is visible on the right hand side. Note that we use gripper-centric coordinates which is where the receiver is placed, the body of the robot is on cell (40, 8)

A custom wooden enclosure was made to hold the arrays, which when delivered are just flat PCBs. The enclosure holds the PCBs from side and are designed to not be anywhere near the four active antennas

¹The accuracy is limited to the tolerance of which the grid structure is built. In this work it is assumed to be perfect

on the top, as seen on the right there is nothing but air around the four top antennas.

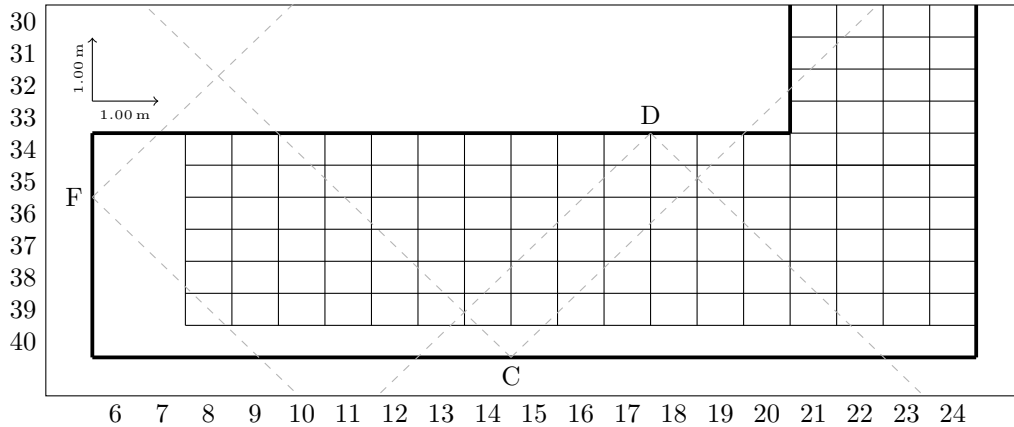


Figure 4.2: Diagram of the grid used for the experiments. The cells that are marked out are the area that are used for our experiments. The thick lines are the grid walls, which are aluminium constructions, see figure 4.1. The three arrays are identified by the first character in their unique IDs: F, D and C. they are oriented respectively east, south and north in this view. We stick to AutoStore coordinates, the short edge is x and long edge is y – therefore this plot has Y coordinates on what is usually the X axis and opposite, for example antenna C is on coordinate (40.5, 14.5)

4.2 Movement of the robot on the grid

There are many ways to command controlled movements of the robot. We use both a grid-scan method and scripting to control the robot to move to specified cells and stay there for a specified duration.

Several measurements were performed, but two are primarily used: in the first only array F is active to ensure maximal data, the robot was stationary for one minute per cell and 96 cells were measured. In the other run all three arrays were active, the robot was stationary for one minute per cell and 118 cells were measured

4.3 Antenna-array configuration

4.3.1 Hardware – Insight SIP-1907 AoA-DK

The arrays purchased for this project are the InsightSIP AoA antenna arrays. They are a copy of an implementation of a design by Nordic Semiconductor, the Nordic Semiconductor version was used in [5], with two minor modifications documented in Insight SIP application note [21]: The transmission line lengths were adjusted to ensure equal length. And the ground plane was changed from layer 4 to layer 2 (such that the ground plane is now directly underneath the antenna instead of on the other side of the PCB) and the antenna feed was correspondingly changed to the bottom layer in order separate the feed from the antenna which according to Insight SIP caused highly non-uniform amplitude and phase response. In my opinion both of these are welcome changes.

In this work it is not necessary to use twelve antennae. Since the AutoStore robots move on the surface of a plane it is only necessary to estimate one angle, assuming that antennae are mouted on the same elevation as the robots – which is the most feasible orientation anyway.

4.3.2 Antenna-array hardware bugs

Of the four antenna-arrays purchased for this project² two were unusable due to hardware issues – one of these two boards were fixed:

²One was financed by NTNU, three were financed by AutoStore

The hardware bug was a manufacturing issue. Two signal lines which drive the selection of antenna in the RF-switches were shorted. The RF-switches are (4×4 millimeters) with an exposed ground-pad with small pins – in my personal opinion the vias to connect each signal is probably a bit too close for comfort, and could probably be moved to reduce soldering issues like these.

In one of the bad arrays the error was that two of the switching-lines were shorted³. To locate the specific RF-switch that caused the issue a combination of logic and DC-resistance-measurements were used; since the lines that were shorted belonged to the outer-ring of switches it was apparent that it was one of those and was identified by measuring the DC-resistance close to each RF-switch. Luckily it was the one responsible for switching antennas 5-7, and so is not needed for this research and was removed in the ugliest solder-job ever performed⁴:

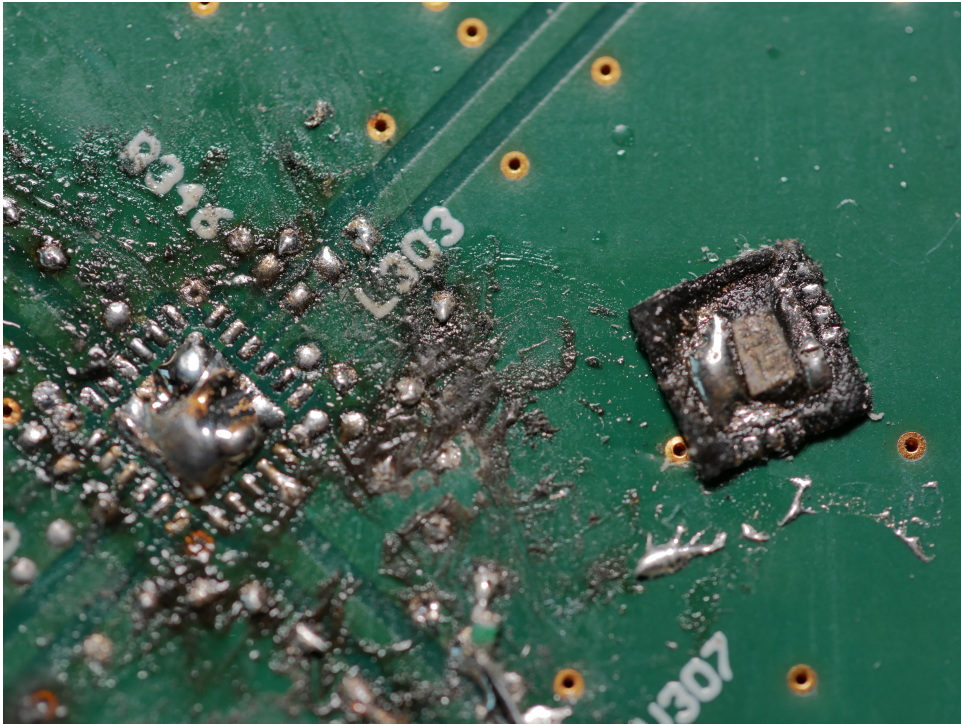


Figure 4.3: Desoldering of RF-switch to fix shorted signal lines – the offending switch is the black square. The area on the left is where the chip was mounted and two of the small lines were shorted. Believe it or not this horrendous soldering job works! Photo: Jon Rogne

4.3.3 GPIO changes

An undocumented change in the Insight SIP design is that the GPIOs that drive the RF-switches was changed from the Nordic design. It's hard to debug CFO estimation when the received signals are wrong in the first place and I recommend anyone working with Bluetooth direction finding to verify that the switching pattern in correct in hardware before attempting to make it work in software.

To find the new GPIO configuration oscilloscopes and guessing based on datasheets were employed. To anyone using the Insight ISP antenna arrays in the future the GPIO configuration in the overlay file should be:

```
dfegpio0-gpios = <&gpio0 2 0>;
dfegpio1-gpios = <&gpio0 3 0>;
dfegpio2-gpios = <&gpio0 4 0>;
dfegpio3-gpios = <&gpio0 5 0>;
```

³Measured DC-resistance on the testpoints was around 1.2ohms

⁴In my defence: The RF-switch is 4×4 millimeters, has a ground pad connected to a ground plane with a high thermal capacity and I only had a normal soldering iron to work with – I had to scrape off the plastic housing of the RF-switch to transfer enough energy to the ground plane to desolder it. And it works damnit!

4.3.4 Switching pattern

Switching pattern is not, according to our results in chapter 2 and 3, a critical variable and we use a linear repeating scan: 1, 2, 3, 4. We use switch-sample length of $1.00\ \mu\text{s}$ and use the full length of $160.00\ \mu\text{s}$ resulting in a total of 82 samples of the four antennas.

4.3.5 Firmware

Example firmware by Nordic was used with a modification to solve a major issue: when a receiver has connected to an array the array will never be connectable by another receiver. This is an issue because if it loses connection or the receiver is restarted there will never be data from the array again. This is however reset by a power-cycle or system reset.

To circumvent this issue each array will, after having connected to a receiver, trigger a complete system reset after ten seconds – when restarting the array will delay for 10 seconds before advertising to allow another array to connect to the receiver first.

4.4 Receiver configuration

4.4.1 Hardware – NRF52833DK

We use a development kit from Nordic Semiconductor, the NRF52833DK, as the receiver. We use the integrated PCB antenna instead of an external antenna for all tests. The reason we use the integrated PCB antenna is mostly due to time-constraints – we assume that the design on the development kit is quite good and beginning to choose different antenna and mounting was not prioritized in this work.

4.4.2 Firmware

Example firmware by Nordic was used for CTE reception. Similarly to the array there were significant issues with multiple connections: the receiver was not able to connect to multiple CTE arrays at the same time, and similarly to the array it would never reconnect to any array even after disconnecting from one array. Therefore a similar workaround was chosen.

After having connected to an array the receiver logs data to the logging system described in 4.5, when the array disconnects (due to its 10 second period running out and it triggering a system reset) the receiver also triggers a system reset and immediately scans and connects to an array. This will be probably be a different array due to the 10 second delay in the arrays startup.

This is a rather ugly workaround but it is sufficient for this test where the robot is stationary in each cell for an extended period of time.

4.5 Logging and interfacing to the AutoStore system

Two variables must be logged in real time: CTE data from the receiver and robot-positions. This is not trivial as the receiver is on the robot, and connecting directly to the robot to get its position data is not straight-forward. Robot positions are however readily accessible by connection via ethernet to the AutoStore driver computer. Therefore two laptops were added to the setup: one is accessing data from the AutoStore driver through a custom internal tool and the other, which is stationed on the robot, gets robot positions from the first computer and transfers those to the logging module.

The logging module is a custom device made by the author for a different project. It has an Inertial Measurement Unit (IMU) logging at 200 samples per second, a power source and an SD card for data storage. A custom connector was made to establish communication to the receiver and a custom communications protocol was designed to transfer data reliably between receiver and logging module.

The result is a binary file on the logging module with realtime IMU, CTE samples from three arrays and robot positions.

Chapter 5

Results

We start the results chapter by comparing some beamformer spectra with multiple frequencies to compare to what the simulations in chapter 3 shows – that using multiple frequencies is a good idea. Then we investigate the relationship between number of frequencies and estimation performance before finally showing the best estimate of each cell using all data from the three arrays.

Tens of thousands of measurements per channel was collected, and there are an almost unlimited number of ways to combine channels together. We can't visualize all of these here, but we have analyzed a large quantity of these and selected only a few to show here. They show the general picture, while being easy to look through.

5.1 Frequency dependence

In simulations we show that the error is heavily dependent on the frequency used at the measurement instant. This is also shown in real data by for instance [6]. Here we begin by showing some beamformer spectra for some channels in a cell, and show that:

H1 Reflections are highly frequency dependent in indoor situations due to the geometry of the reflection and direct path: A slight change in wavelength changes the output of each path differently.

H1.1 Two measurements with the same channel is comparable. If not then H1 is not meaningful

Figure 5.1 shows that H1 is true. Changing channel *does* change the errors significantly and two measurements with the same channel are similar. It is also clear that the frequency dependence is more rich and varied than that of the simulations, which is not a big surprise as the simulation only had two reflection sources while it is probable that there are a great deal of reflections in an AutoStore environment.

H2 Combining multiple channels compensates for reflection-caused errors in some cases

First we will look at some beamformer spectra when all channels are combined. This is done in the beamformer cost function, but since the operation is linear it is just a sum of all spectra for each channel.

We show several such examples. The first, figure 5.2, is the same cell as shown in figure 5.1 and shows that the combined spectrum approaches the correct solution. Then a few others are shown: a reflection with less power than the true signal: figure 5.3, a reflection with more power than the true signal: figure 5.4 and 5.5, and a situation with clear line of sight where the combined spectrum does not approach the correct solution: figure 5.6.

It is important to note that figures, 5.2, 5.3, 5.4, 5.5 and 5.6 are one outcome of the Bluetooth pseudorandom frequency hopping for each cell. All measurements are combined into one, but this does not capture the random nature of the selection and it is possible that different selections yield different results. To investigate this better a monte carlo approach is performed in section 5.2.

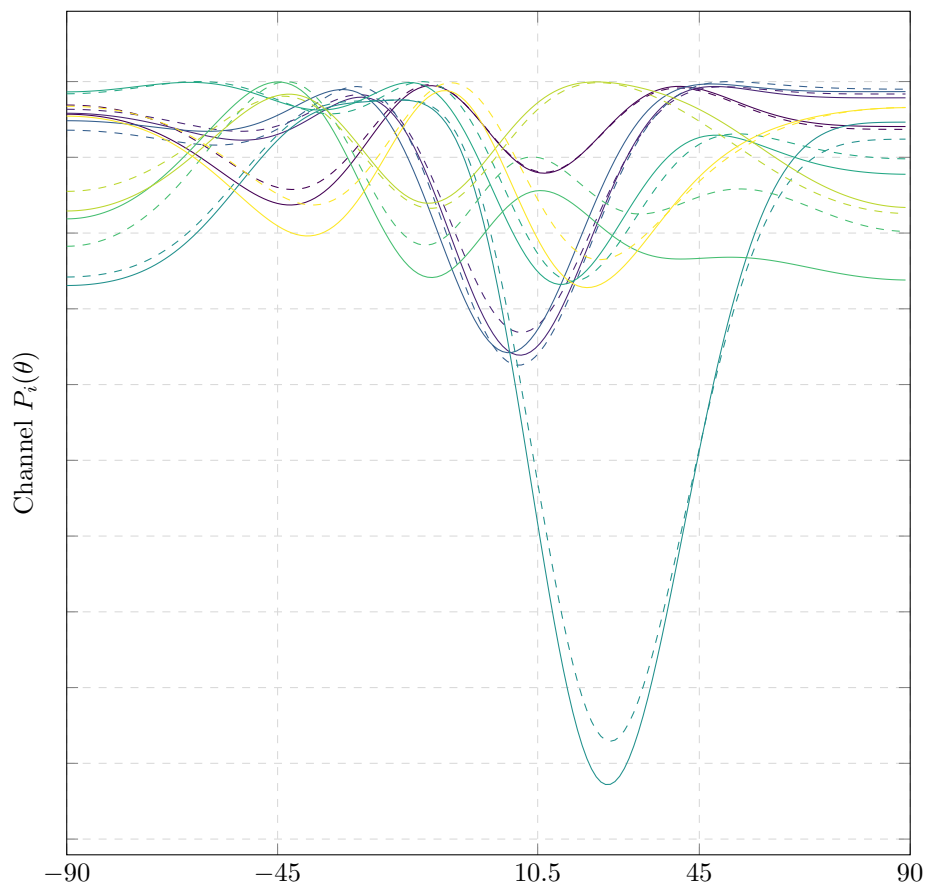


Figure 5.1: Comparing beamformer spectra from channels in the range 0-35 for the cell (39, 19) with array F. Each channel is plotted twice: one solid and one dashed line. This clearly shows that H1 is true since each channel yields different results but are consistent with itself

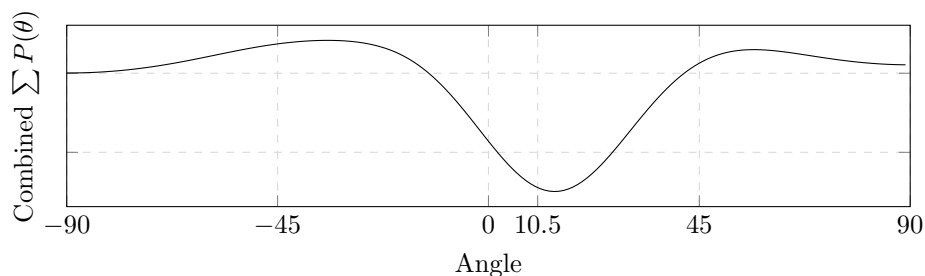


Figure 5.2: Combining all measurements of cell (39, 19) with array F. It is clear that the correct angle around 10.5° is detected, indeed it is the primary angle detected by the beamformer and any reflections are too noisy to be significant when all channels are combined.

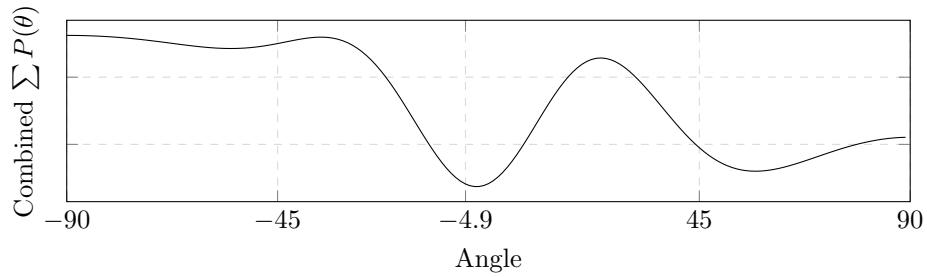


Figure 5.3: Combining all measurement of cell (34, 18) with array F. The correct angle -4.90° is detected, but a reflection at around 60.00° is also detected – indicating that the reflection is shared by enough channels such that it is picked up in the when all channels are combined.

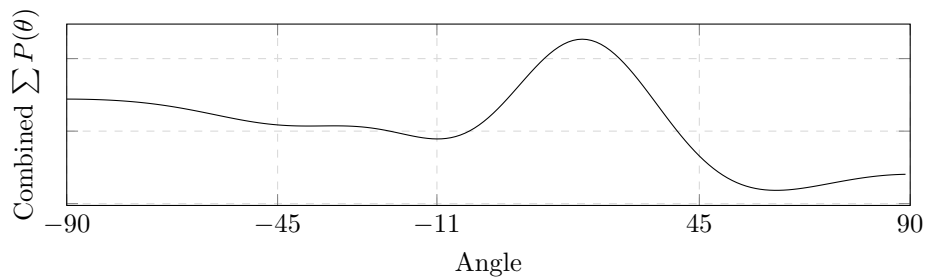


Figure 5.4: Combining all measurements of the cell (34, 11) with array F. The correct angle -11.00° is barely detected, and a reflection has more power than the true direction. This indicates that combining multiple frequencies is barely enough to compensate for reflections in this case.

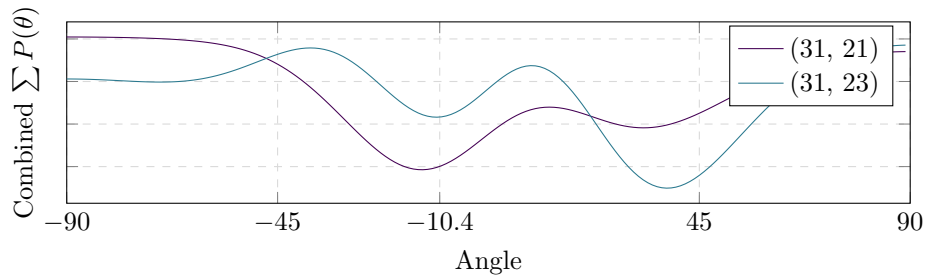


Figure 5.5: Cell (31, 23) and (31, 21) are occluded for antenna F (there is a metal fence between the array and receiver), but it is still able to estimate the correct direction – for cell (31, 23) the correct direction has less power than a reflection and multiple optima detection must be used to find it

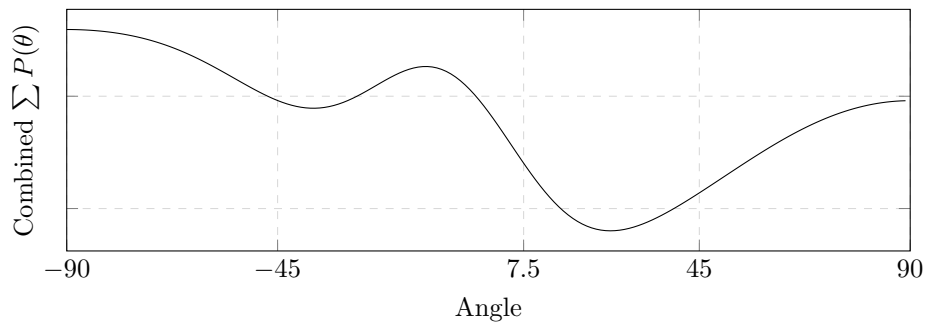


Figure 5.6: Cell (38, 19) is not occluded and one would expect good measurements of the cell, but this is not the case. Using multiple frequencies is not capable of compensating for the reflections and the total direction estimate is far away from the correct solution direction 7.5°

5.2 Monte carlo RMSE with N frequencies

To investigate further how multiple channels improve the estimation we perform a monte carlo simulation where a random number n of random channels are combined and the direction estimated. The goal is to quantify the RMS error for angle estimation with varying number of channels. It is important that channels are randomly selected to prevent bias and to give a more general picture than the Bluetooth pseudorandom frequency hopping gives in the previous results.

The method works as follows:

1. Prepare 37 lists to store estimates, one per number of channels used – from 1 to 37
2. loop many¹ times
 - (a) Generate a random channel selector (a random number of random channels)
 - (b) Try to get one measurement per channel in the cell. Note how many actually were found
 - (c) Calculate best estimate using the measurements
 - (d) Store the estimate in the list for the corresponding number of channels actually used
3. Calculate RMSE for the 37 lists. Each will be the RMSE with n channels in use

Due to the datastructures used this will get an effectively random collection of channels to estimate direction, and the relation between number of channels in use and the RMSE value can be investigated.

Note especially that this method shows the result with n channels, only one measurement per channel is used per calculation. It must also be noted that this does not explicitly capture the *spread* of frequencies, only the number of frequencies in use.

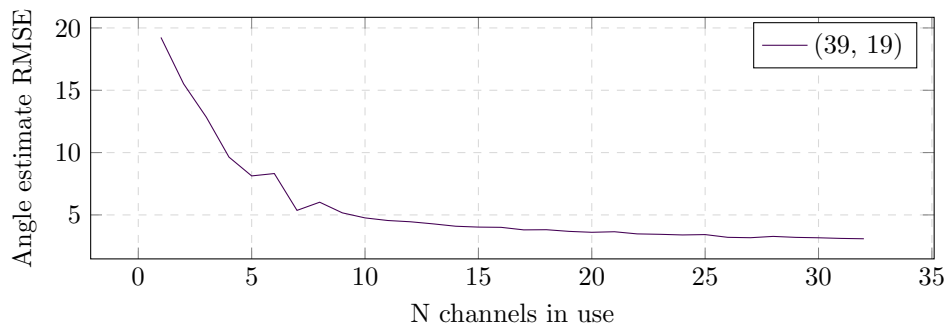


Figure 5.7: It is clear that when there is a good signal increasing the number of channels in use results in good performance. With 32 channels the RMSE is 3.00° , but with just 10 the RMSE is 5.00° showing clear diminishing returns.

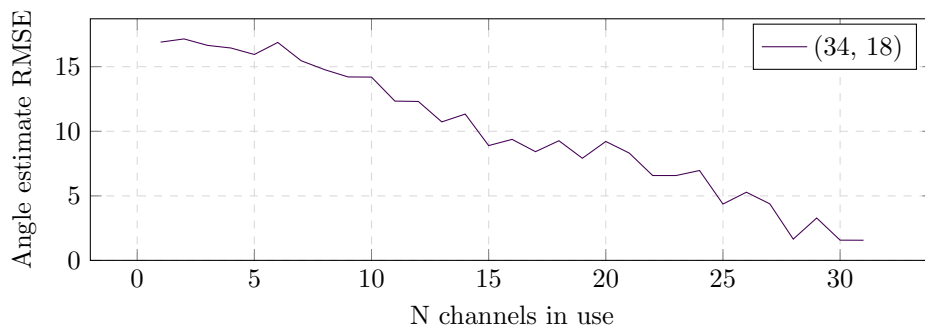


Figure 5.8: Again more channels gives better results, and in this case where the signal is weaker in relation to some reflections more frequencies are required to reach good estimates.

¹we loop 20k times in this work

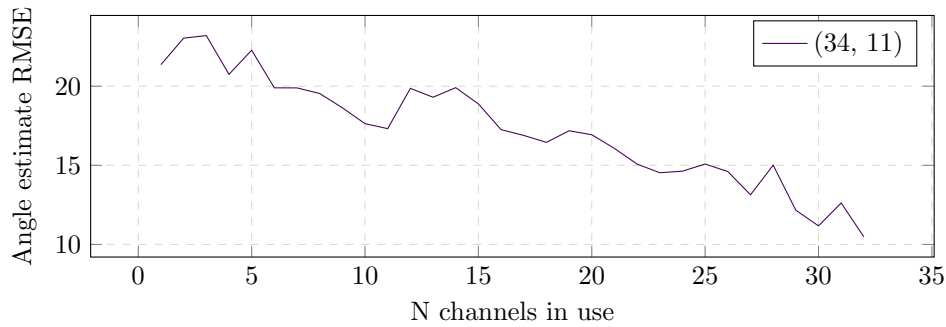


Figure 5.9: Cell (34, 11) was, as shown in figure 5.4 barely able to estimate the correct direction. In this plot we show that the RMSE angle estimate is 10.00° with 32 channels and figure 5.4 was probably lucky in the exact combination of channels selected by the pseudorandom channel hopping.

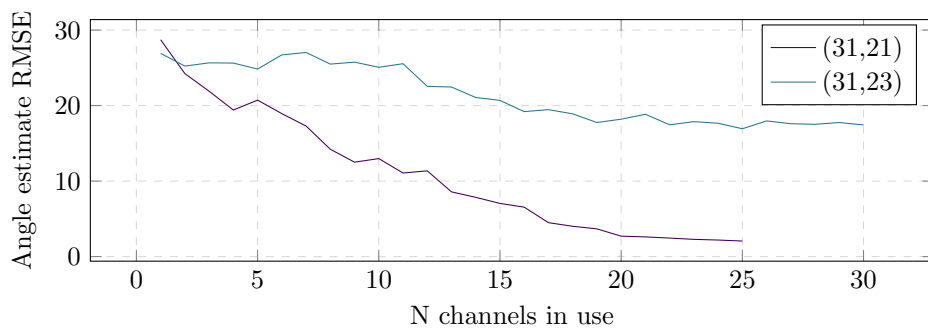


Figure 5.10: Cell (31, 21) and (31, 23) are both occluded and figure 5.5 showed that both could yield good estimates. This is only partially true, for the random selection of channels in figure 5.5 both were detectable, but in the average case cell (31, 21) is much more accurate than (31, 23)

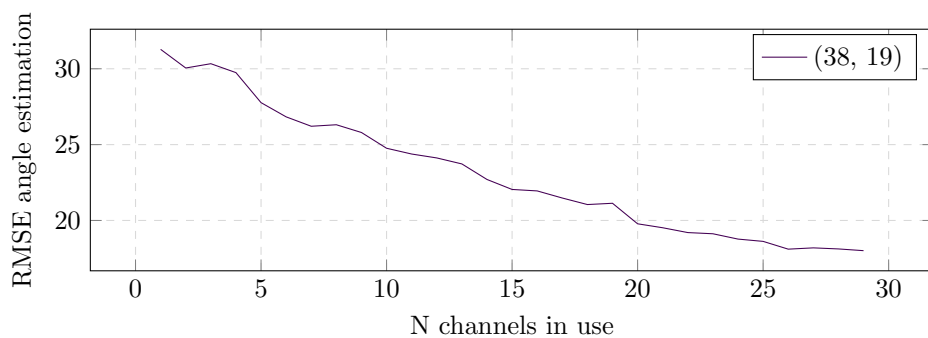


Figure 5.11: Cell (38, 19) was shown in 5.6 to not be very detectable. This is also true in the average case. Still it is better to use multiple channels

5.3 Grid measurements

Finally we show the results of measuring the grid in figure 5.14, 5.15, 5.12 and 5.13. The direction estimate error is shown in the corresponding cell.

The expected direction is calculated from the array and receivers position and orientation. The measured direction is the best of the multiple results from the direction estimator – this assumes that a tracking system is able to correctly identify the correct direction when given a list of three or four possible directions²

We only show directions within each arrays 70.00° angle. This is because the direction finder often gives results close to 90.00° , and the assumption that a tracking system can pick the correct direction breaks down – at close to 90.00° the array is not really able to distinguish directions due to the lower resolution. Therefore the errors at angles close to 90.00° are suspiciously close to the true solution and removed. The selection of 70.00° is fairly arbitrary.

Additionally, if there are fewer than 20 CTE samples collected at a cell it is not shown in these figures.

²This is a fair assumption and many techniques could be used to identify the most likely direction in a tracking system.

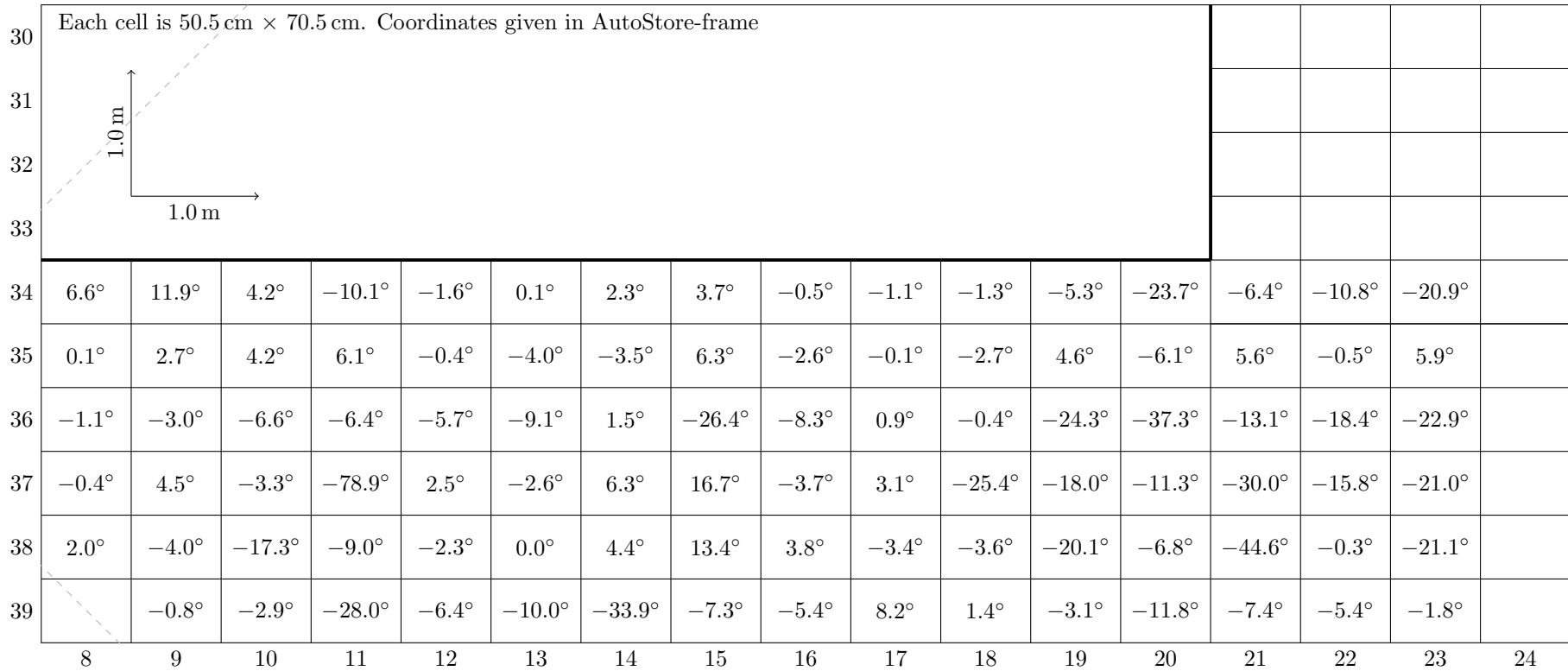


Figure 5.12: Angle estimation errors for the first measurement with antenna F (mounted on coordinate (35.5, 5.5) pointing east). In this run this is the only array active and each cell was measured for 60 seconds

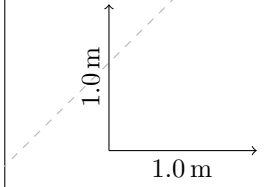
30	Each cell is 50.5 cm \times 70.5 cm. Coordinates given in AutoStore-frame												-52.0°	12.0°	-4.7°	46.7°	
31													2.5°	-1.4°	0.7°	3.8°	
32													-4.4°			2.8°	
33													2.8°	-26.2°	-30.1°	9.4°	
34													4.9°	4.6°	4.7°	-0.2°	-2.4°
35	-0.2°	1.2°	3.3°	3.4°	-1.2°			-2.2°	-2.9°	-1.4°	-0.9°	-29.1°	10.3°	2.4°	-0.3°	6.4°	-0.3°
36	-2.3°	-3.9°	-7.4°	-12.1°	-4.8°	-6.5°	2.7°	-19.9°		1.1°		-27.7°	-35.4°	-10.0°	-18.6°	-12.2°	-18.7°
37	-1.8°	1.9°	-0.9°	-18.2°	1.3°	-0.3°	7.2°	14.7°	-5.1°	2.3°		-20.9°	-16.3°		-7.7°	-14.9°	5.7°
38	2.4°	-5.8°	-14.6°	-8.4°	-4.6°		2.3°		0.4°	-1.6°	-0.6°	-18.7°	-9.1°	22.0°	16.8°	-19.3°	-12.9°
39	3.1°	0.7°	-2.8°	-28.3°	-6.9°		-36.8°	-5.2°	-5.8°	5.6°	-5.4°	-3.6°	-9.1°	-5.2°		5.5°	5.9°
	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24

Figure 5.13: Angle estimate errors for the second measurement with array F. In this run all three arrays are active which results in an average of 20 seconds measurement time per array per cell

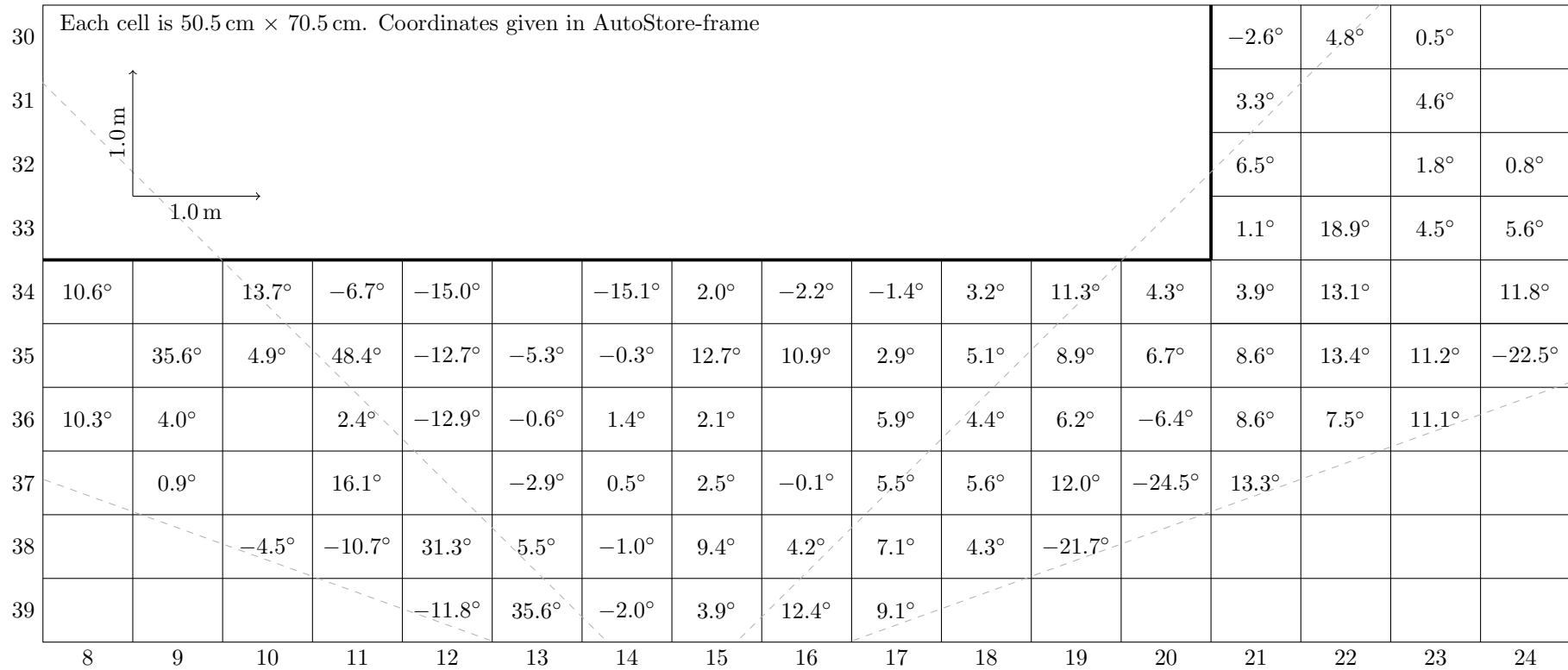


Figure 5.14: Angle estimate errors with antenna C (mounted on coordinate (40.5, 14.5) pointing north)

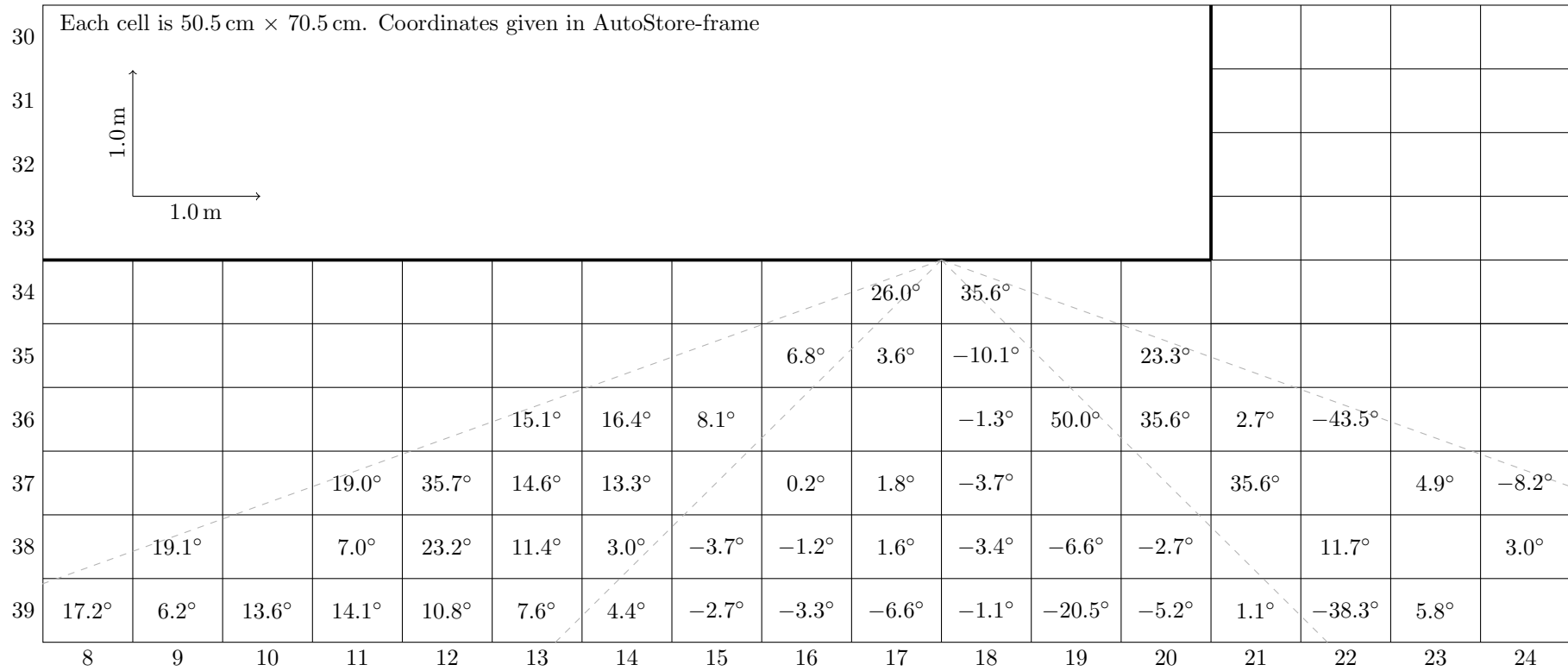


Figure 5.15: Angle estimate errors with antenna D (mounted on coordinate (33.5, 17.5) pointing south). Cells (34, 17-18) may be bad due to the plane-wave assumption breaking down at short distances.

Chapter 6

Conclusion

The CFO estimator from [5] was improved with numerical optimization techniques to enable real-time operation on a microcontroller – it is shown to be stable and orders of magnitude faster. If one however does not want to implement BFGS2 or similar we have shown that a stepsize of 500 is a better tradeoff than 25 for speed and accuracy.

Reflections were simulated and shown to be a major contributor to angle-estimation errors. However we also show that both frequency and position influence the reflections such that utilizing the full Bluetooth spectrum yields significantly better results and can, as shown in section 5.2, sometimes reduce the RMSE error from 20.00° to 3.00° . Sometimes, however, the Bluetooth spectrum is not wide enough to compensate for all reflections – in this case one could use the position dependence to improve performance, something which a tracking system is likely able to exploit.

Finally it is shown that for a system such as AutoStore there will be areas that are not measured well, even with multiple frequencies, such that a tracking system must be able to reject bad data and rely on forward-estimation until the data improves.

6.1 Future work

Since multiple frequencies yield better results it would be a major benefit if a CTE could be sent using multiple frequencies. Assuming that the transmitter and receiver can change channel quickly enough one could design a multi-channel CTE such that each channel is represented at the same position. This would increase the temporal relation to each channel instead of the current solution where each channel is separated in time. This is more important for tracking moving targets.

Additionally a tracking system must be developed to take advantage of multiple frequencies, find multiple hypothesis in the beamformer spectrum, and deal with periods of bad data. As such it is likely to either need sensors such as an IMU onboard or more advanced tracking techniques that work with multiple hypotheses simultaneously.

For a system such as AutoStore it is desirable to have local tracking at each robot, where an IMU is probably readily available but the tracking algorithm should be able to run on a microcontroller.

Bibliography

- [1] H. D. Young and R. A. Freedman, *University Physics*, vol. 2. Pearson Education Limited, 14 ed., 2015.
- [2] D. H. Johnson and D. E. Dudgeon, *Array Signal Processing*. P T R Prentice-Hall, Inc., 1993.
- [3] L. Euler, *Introductio In Analysin Infinitiorum Vol.1.* -, 1748. Translation by Ian Bruce can be found at <http://www.17centurymaths.com/contents/introductiontoanalysisvol1.htm>.
- [4] R. R. Lyons, "Sum of two sinusoids," 2011.
- [5] M. L. Sollie, K. Gryte, T. H. Bryne, and T. A. Johansen, "Outdoor navigation using bluetooth angle-of-arrival measurements," *IEEE Access*, vol. 10, pp. 88012–88033, 2022.
- [6] V. Sesma and V. Egorov, "Improved accuracy for indoor positioning with bluetooth 5.1: From theory to measurements," Master's thesis, Lund University, 2020.
- [7] A. Goldsmith, *Wireless Communications*. Cambridge university press, 2005.
- [8] *Bluetooth Core specification 5.1*.
- [9] G. D. Durgin, *Space-Time Wireless Channels*. Prentice Hall PRT, 2003.
- [10] D. M. Pozar and D. H. Schaubert, *Microstrip Antennas*. John wiley and Sons, 1995.
- [11] L. Yao, "Bluetooth direction finding," Master's thesis, Delft University of Technology, 2018.
- [12] J. Nocedal and S. J. Wright, *Numerical Optimization*. Springer, 2 ed., 2006.
- [13] G. M. et al, *GNU Scientific Library Reference Manual*, 3 ed.
- [14] P. Moose, "A technique for orthogonal frequency division multiplexing frequency offset correction," *IEEE Transactions on Communications*, vol. 42, no. 10, pp. 2908–2914, 1994.
- [15] M. Woolley, *Bluetooth Direction Finding, A Technical Overview*, 2021.
- [16] C. Stefan D., "Bluetooth low energy direction finding on embedded hardware by mitigating carrier frequency offset and multipath fading," Master's thesis, Eindhoven University of Technology, 2021.
- [17] C. E. Shannon, "Communication in the presence of noise," *Proceedings of the IRE*, vol. 37, 1949.
- [18] C. C. Aggarwal, *Time Series and Multidimensional Streaming Outlier Detection*, pp. 273–310. Cham: Springer International Publishing, 2017.
- [19] M. A. G. Al-Sadoon, N. T. Ali, Y. Dama, A. Zuid, S. M. R. Jones, R. A. Abd-Alhameed, and J. M. Noras, "A new low complexity angle of arrival algorithm for 1d and 2d direction estimation in mimo smart antenna systems," *Sensors*, vol. 17, no. 11, 2017.
- [20] R. J. Mailloux, *Phased Array Antenna Handbook*. Artec House, 2 ed., 2005.
- [21] I. SIP, *ISP1907-AOA-DK Application note AN210105*.



 **NTNU**

Norwegian University of
Science and Technology