Graz University of Technology

Master Thesis

Positioning Using a Massive MIMO Software Defined Radio

conducted at the
Signal Processing and Speech Communications Laboratory
Graz University of Technology, Austria

by
Andreas Fuchs, 01230349

Supervisors:
Assoc.Prof. Dipl.-Ing. Dr. Klaus Witrisal
Dipl.-Ing. Thomas Wilding

Graz, January 10, 2020

Statutory Declaration

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

date	(signature)

Mit herzlichem Dank an meine Eltern, Irene und Bernhard,
welche es mir ermöglicht haben dieses Studium zu absolvieren und
mich in jeglicher Hinsicht während der gesamten Studienzeit
unterstützt haben.

Abstract

This work focuses on positioning on a 2D plane using a massive MIMO software defined radio, yielding distance and angle of a movable *mobile station* relative to a stationary *base station*. Within the scope of this work, the software defined radio (built by National Instruments) was programmed to save *channel estimates* from real-world measurements of an 8 by 2 antenna MIMO OFDM transmission. The measurements were done outdoors and indoors, as to give results with and without significant *multi path components* and different environments. Hardware limitations, especially very small available bandwidth had to be taken in consideration. Furthermore, a mathematical model of the signal processing in the hardware, the channel, received signals and estimation of the distance and angle was formulated. This mathematical model was implemented and used with data from the measurements, yielding estimates for the two parameters. Lastly, focus shifted on a statistical evaluation of the results, comparing them to theoretical lower limits of accuracy imposed by the *Cramer-Rao lower bound*. Results showed that for two evaluated measurements the performance of distance estimation was sub-par, because the (unwired) synchronization between *base station* and *mobile station* was insufficient. In a third evaluated measurement, synchronization was done with an external cable, yielding results near the *Cramer-Rao lower bound* for distance estimates. The angle estimation generally performed more reliable, but showed systematic errors for indoor localization, which could be attributed to reflections being dominant over *line of sight* components.

Kurzfassung

Das Ziel dieser Arbeit bestand darin, mittels einem Massive MIMO Software Defined Radio, Winkel und Distanz (in einer Ebene) einer beweglichen *Mobile Station* relativ zu einer feststehenden *Base Station* zu ermitteln. Dieses Software Defined Radio, vertrieben von National Instruments, wurde softwareseitig modifiziert, um Schätzungen einer *Kanalübertrangungsfunktion* von Messungen einer 8-mal-2 MIMO OFDM-Übertragung zu speichern. Diese Messungen wurden im Inneren eines Gebäudes sowie draußen, auf einer weitläufigeren Wiese, durchgeführt, um die Effekte von *Mehrwegeausbreitung* und verschiedene Umgebungen zu betrachten. Einschränkungen der Hardware, vorrangig die geringe verfügbare Bandbreite, mussten in weiterer Folge beachtet werden. Weiters wurde eine mathematische Beschreibung der Signalverarbeitung der Hardware, des Kanals, der empfangenen Signale und der Schätzung von Winkel und Distanz gefunden, welche weitergehend in Software umgesetzt wurde. Diese Umsetzung nutzt Daten der schon besprochenen Messungen, und liefert Schätzwerte für die beiden Parameter. Im letzten Teil der Arbeit wurde eine statistische Auswertung der gefundenen Schätzwerte vorgenommen, in welcher die Daten mit einer unteren Grenze für die Genauigkeit, der *Cramer-Rao Lower Bound* verglichen werden. Die Ergebnisse für zwei von drei Messungen zeigten relativ hohe Abweichungen für die Schätzung der Distanz, vorwiegend weil die nötige Synchronisierung zwischen *Base Station* und *Mobile Station* nicht ausreichend gut war. In einer dritten ausgewerteten Messung zeigte sich, dass durch eine externe Synchronisierung mittels Kabel Ergebnisse im Bereich der *Cramer-Rao Lower Bound* für die Distanzschätzung möglich sind. Die Winkelschätzung war im Allgemeinen zuverlässiger, zeigte aber systematische Fehler bei den Messungen im Innenraum, welche auf Reflektionen zurückgeführt werden können, die die *Line of Sight*-Komponenten überlagern.

Contents

1 Introduction

1.1 Introduction

The aim of this Master thesis, as described in the abstract, was to use a software defined radio (SDR) available at the Signal Processing and Speech Communications Laboratory (SPSC) for indoor and outdoor positioning. This was achieved by first modifying existing code from National Instruments (NI) as described in Chapter 2, to save needed data, writing code in Matlab for finding the distance and angle of the measurements as described in Chapter 4 and lastly evaluating the estimation with regards to their theoretical lower limit as described in Chapter 5. The used SDR is a multiple input multiple output (MIMO) system, here configured with 8 antennas for the base station (BS) and 2 antennas for the mobile station (MS). The data of the 8 antennas on the BS is used to estimate the position of the MS. The system's main purpose lies in research for 4G and 5G communication. However, research in the direction of location-awareness [1] is ongoing, and is deemed necessary for some future applications. MIMO systems are one possibility to achieve satisfying resolution of localization, to overcome multi path propagation degrading the localization performance of single antenna solutions [2]. Most research done in this field relies only on simulations, and thus real-world measurements are needed to verify theoretical limits. This has been done for example by [3], but the antenna array used was a big cylindrical antenna array with 64 dual-polarized antennas. They were using hardware capable of directly measuring the channel impulse response (CIR), as opposed to this work using the transfer function (TF) from channel estimation, a by-product of the coding. The antenna array used here is an 8 antenna uniform linear array (ULA), which is a configuration explicitly allowed for example by Wi-Fi 6 (802.11ax) [4].

1.2 Overview of Chapters

In Chapter 2 there is a short overview of the used hardware and limitations imposed by it. Further, one can find an overview of the used transmission scheme, orthogonal frequency division multiplexing (OFDM), the hardware uses for data transmission, the principle of pilot symbols and reciprocity calibration. Lastly in Chapter 2, there is a short description of the changes made in code to get the data needed for further processing. In Chapter 3 a mathematical description of the processing done in the Hardware from NI, and the further signal processing done in the algorithms can be found. Chapter 4 focuses on the code written in Matlab which applies the previous mathematical description. It outlines used principles like grid-search, a modified maximum-likelihood (ML)-estimator and output-calibration of data. Also the measurements are described in more detail. Chapter 5 is focusing on the performance of the estimator, comparing it to the Cramer-Rao lower bound (CRLB). This is performed for one outdoor and two indoor measurements. One of the indoor measurements was done with wired synchronization between BS and MS instead of over the air (OTA) synchronization. In Chapter 6, there is a conclusion of this Master thesis, and further outlook as to what could be done to improve the positioning accuracy. The last Chapter 7 contains references, pictures, acronyms and some code.

2 Hardware and Implementation

2.1 Overview

The used hardware is an SDR from NI, consisting of multiple *USRP-RIOs*, which are dedicated antenna controllers. The key features of the massive MIMO system are a frequency coverage from 50MHz – 6GHz, a bandwidth of about 20MHz, 2 – 128 antennas for the BS (seen in Figure 2.2(a)) and up to 12 different MSs (seen in Figure 2.2(b)). In our configuration, there are 8 antennas for the BS and 2 antennas on the MS. These 2 antennas of the MS are handled as two different MSs by the software running on the BS. The different MSs are further called *layers*, shown also in Figure 2.1. The software running on both BS and MS is *Labview* from NI, a graphical programming language. This work is based on an existing program from NI, modified to save TFs. The program is able to transmit data between BS and MS using OFDM, synchronizing between them, visualizing TFs of all antennas as well as their corresponding CIRs. It communicates with the *USRP-RIOs* (seen in Figure 2.1), which are doing the actual signal processing in a field programmable gate array (FPGA) running custom software. The interesting parts are the channel estimates, which are saved as TFs, as they can be used for positioning. The actual transmission of data is not interesting, as this does not affect the channel estimation.

2.2 Base Station

The BS used for measurements in this thesis is capable of using up to 128 antennas, in our case it was configured for 8 antennas. Two antennas each are connected to a single *USRP-RIO*, a device combining an FPGA and two separate SDR modules. These *USRP-RIOs* only have two channels each. Four of these modules are connected via peripheral component interconnect extended (PCI-X) to another FPGA module, which pre-processes the data. Lastly, also via the PCI-X bus, the *USRP-RIOs* are connected to a regular x86-Server, running Labview. They are able to perform the transmission and channel estimation in the FPGAs, and are synchronized with each other, using an external clock. This synchronization is done by a separate clock server within the BS, but will not be described in further detail here. Also, they each have a global navigation satellite system (GNSS) module, which would be able to retrieve accurate time from satellites, but this is not included in this thesis.

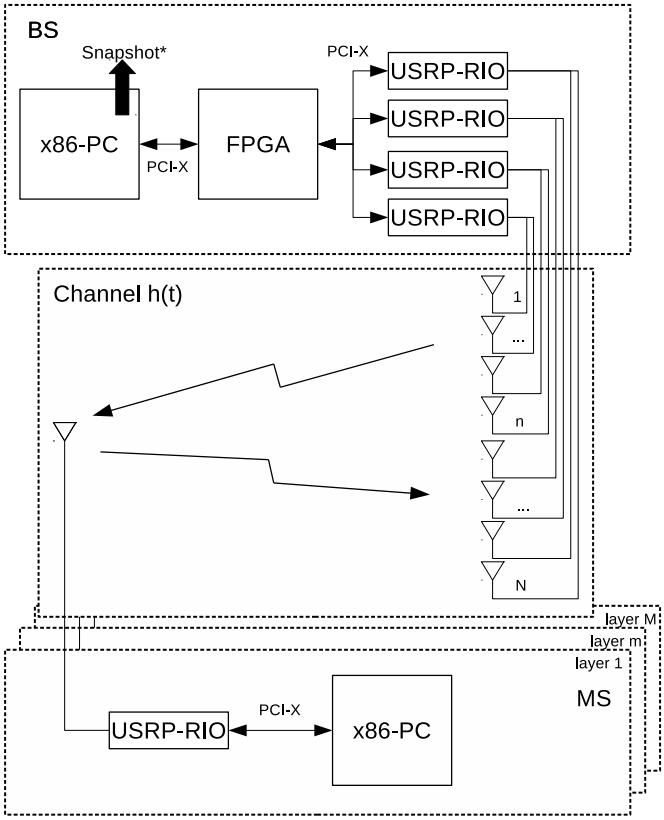
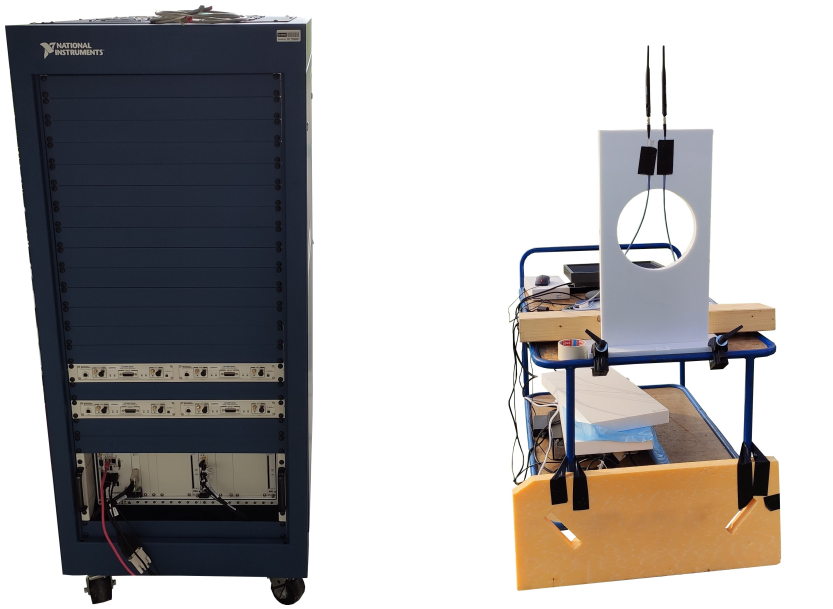


Figure 2.1: Simplified block diagram of the hardware



(a) Base Station

(b) Mobile Station in the used configuration

Figure 2.2: BS and MS in the used configuration

In this thesis, the antenna array was always configured in the same way, a ULA spaced $\frac{\lambda}{2}$ apart. This choice was made so several mathematical and practical simplifications could be made, as seen in Chapter 3. Also, the phase differences of the received signals are unambiguous if the spacing equals $\frac{\lambda}{2}$.

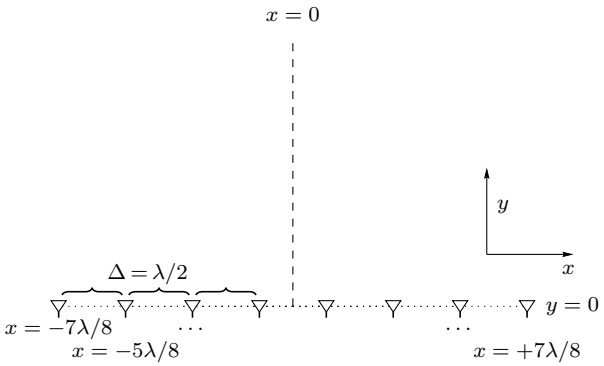


Figure 2.3: Array Geometry, ULA.

As seen in Figure 2.3, for our array geometry, with a ULA, the BS antennas are in-line. The antenna array which is used for actual measurements (Figure 2.4), is connected to the BS. The geometry of the array is described with the equation $x_n = -\frac{7\lambda_c}{8} + \frac{\lambda_c}{2}(n-1)$ depending on the antenna number n, with the leftmost being $n = 1$, and λ_c being the carrier wavelength.

2.3 Mobile Station

The MS used for this thesis, were two antennas connected to one *USRP-RIO*, which was directly connected via PCI-X to another x86-PC. The two antennas are treated as separate MSs for this thesis, but the software (also running Labview) on the MS processed data for both antennas simultaneously. The two antennas are treated in software as separate *layers*, working independently. The MS is assumed to be able to move on a plane. For the measurements there was a carriage in use, as depicted in Figure 2.2(b).

Figure 2.4: Actual used antenna array at the BS

2.4 OFDM

The modulation for data transmission implemented in the hardware from NI is OFDM. OFDM splits the data-stream into multiple parallel streams in the frequency domain. These parallel streams are called *subcarriers*. These subcarriers have to be orthogonal in order to be separated at the receiver, as described in [5, Ch 19]. The modulation on the subcarriers is pulse amplitude modulation (PAM), with rectangular pulses and complex-valued constellations. In the frequency domain, the spectrum of each modulated carrier has a $\sin(x)/x$ shape. At the receiver, the demodulation has to be done appropriately, by multiplying with $\exp(-j2\pi f_k t)$ with frequencies $f_k = kW/K$, where k is an integer subcarrier index, K the number of subcarriers and W the available bandwidth, and afterwards integrating over symbol duration.

Pilot symbols are a way for channel estimation, which is needed for reliable and good data transmission for OFDM. This method involves sending a dedicated known symbol over the channel, meaning data for every subcarrier is known. Afterwards, the known data is used in a linear minimum mean square error (LMMSE) estimator (see [6, Ch 12, p.379-418]), for getting an estimate of the channel. This is done in hardware for every pilot, and the channel estimates are saved to a file. This will be further described in chapter 3.

Subcarriers are spaced equally over the bandwidth of the hardware, and depend on the used layer m. For a description of this, $f_k^{(m)}$ is defined, where k is the number of the subarrier. This vector was retrieved by analyzing the software from NI, and has the form

$$f_k^{(m)} = (f_\eta + m f_\lambda + f_\delta(k-1)) + f_c \tag{2.1}$$

As one can see, the frequencies depend on the layer used, whereas each layer can be assigned to a different MS. f_c mentioned in (2.1) is the carrier frequency, f_η is the relative lower frequency of the spectrum, f_λ is a frequency shift between layers, as to differentiate between different MSs and the spacing between subcarriers is f_δ. The parameters of this are defined in Chapter 4 for the actual measurements. As a short example with values

f_c	f_η	f_λ	f_δ	m
3GHz	−9MHz	15kHz	180kHz	2

Table 2.1: Example values

$\mathbf{f}^{(m)}$ equates to $\mathbf{f}^{(2)} = \begin{bmatrix}-8.97\text{MHz} & -8.79\text{MHz} & -8.61\text{MHz} & \dots & 8.85\text{MHz}\end{bmatrix}^T + 3\text{GHz}$. If m would be changed by 1, $f_\lambda = 15\text{kHz}$ would be added or subtracted.

2.5 Reciprocity Calibration

At the BS, a reciprocity calibration is done between antennas, as the different RF transceiver chains are non-reciprocal. This causes random phase and amplitude differences. This is done every time the BS is powered on, and only once. Internally, single-hop reciprocity coefficients are measured at all subcarriers, which get averaged afterwards. Then, some antennas are chosen as reference antennas, for which estimates for all paths between a random antenna and the reference antennas are estimated. These estimates in turn are combined to get one reciprocity coefficient for every antenna. These coefficients are applied to every transmission (multiplication

of the data stream in the frequency domain).

2.6 Limitations of the Hardware

There are four main limitations imposed by this hardware, which specifically affect our use-case:

System bandwidth: The bandwidth of the system is limited to 17.82MHz. This bandwidth is split into 100 Subcarriers. Using $T_p = \frac{1}{W}$, with W being the total bandwidth and $d_{sample} = T_p c$, where c is the speed of light, this bandwidth and number of subcarriers leads to a resolution $d_{sample} \approx 16.82$m. This means that without any subsampling techniques applied, resolution in the time-domain is low, compared to, e.g., UWB, where bandwidths of up to 1GHz are available.

Snapshot rate: The hardware is limited by the snapshot interval in the time domain. In this work, a snapshot means a full measurement and calculation of TFs. With continuous measurement, there are only 30 to 40 snapshots per second calculated, meaning the channel estimation and following saving is done in this time period. This is due to the channel estimation being computationally expensive.

Number of antennas: The number of antennas on the BS is limited by the available bandwidth of the PCI-X bus, as the OFDM modulation is done at the *USRP-RIOs*. Without any further description, the number of BS antennas is limited to 128 by the data transmission needed to support 12 layers to the *USRP-RIOs*.

Number of layers: On the MS side, 12 antennas are the maximum. This comes from the modulation only differentiating between 12 layers, every layer corresponding to one antenna. The antennas can also be mounted to different MSs, effectively being able to have one user per antenna/MS.

2.7 Changes in Labview Code

The Labview code from NI was modified for this thesis to being able to save the TFs which are estimated by the pilot symbols. This is done (as stated in the previous section) 30 to 40 times per second. The speed of this is determined by the main execution loop. Beside the channel estimation for every antenna in the array, the reciprocity calibration data and timestamps (which are relative to the start of execution) are saved too. A proprietary format from NI is used, called *TDMS*. The usage of this format was a necessity, as other formats were not reasonably time-efficient, and thus would hinder execution of other important subprograms. For reference as to where this saving occured, see (Snapshot*) in Figure 2.1. In order to save these values, the main program and some of the subprograms had to be modified, to allow the routing of values to the correct subroutine. As Labview is a graphical programming language, but still holds on to the concept of scopes of variables, this was a rather tedious task. For some reference on the Labview code see Chapter 7.

3

Mathematical Description and Model

In this chapter there will be a short mathematical description of the channel, the signal model, and the optimization problem. This knowledge is fundamental for the algorithms and implementations done in Chapter 4.

3.1 Channel Model

For better understanding of the following mathematical description, the geometry of the antenna array at the MS relative to the BS is shown here:

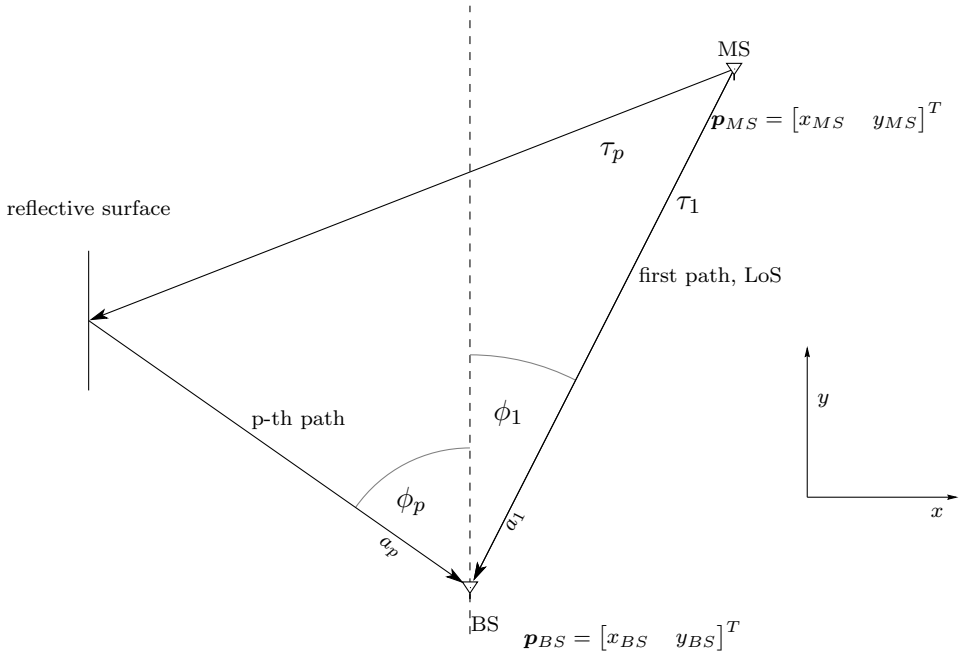


Figure 3.1: Geometry of BS and MS

Here, p denotes the index of the multi path component (MPC), with $p = 1$ being the line-of-sight (LoS) path (as depicted), and $p \geq 2$ are so-called specular MPCs. The parameters ϕ_p and τ_p are the angle and delay for the respective path or the LoS component. The vectors $\mathbf{p}_{BS}$ and $\mathbf{p}_{MS}$ consist of the x and y coordinates of the BS and MS. As an estimate of distance and angle is to be found, one can convert between the x and y coordinates by computing

$$\phi = -\arctan\left(\frac{y_{MS} - y_{BS}}{x_{MS} - x_{BS}}\right) - \frac{\pi}{2} \tag{3.1}$$

and

$$d = \sqrt{(y_{MS} - y_{BS})^2 + (x_{MS} - x_{BS})^2} \tag{3.2}$$

with d being $c\tau$. The MS shown in Figure 3.1 is a single antenna, the BS is a ULA, as described in Chapter 2. The channel is modeled as

$$x(t, \tau) = \sum_{p=1}^{P} a_p(t)\delta(\tau - \tau_t) \tag{3.3}$$

In (3.3), P is the maximum number of paths, $a_p(t)$ are the time-dependent complex amplitudes of the signal, τ is the delay relative to measurement time, and τ_t are the delays corresponding to the signal amplitudes $a_p(t)$. This is a model for a wide-sense stationary and uncorrelated scattering (WSSUS) channel, as described in [5, p. 111]

3.2 Signal Model

3.2.1 Received Signal

For this, we assume a received signal overlayed by some noise and transformed to the frequency domain for demodulation. Our interest here lies within the channel gain vector, a byproduct of the OFDM scheme. The received signal vector is written as

$$\boldsymbol{y}^{(m)} = \boldsymbol{H}^{(m)}\boldsymbol{a}^{(m)} + \boldsymbol{w} \tag{3.4}$$

$\boldsymbol{y}^{(m)} \in \mathbb{C}^{NK\times 1}$ for every layer m. N is the number of antennas, and K is the number of subcarriers. $\boldsymbol{H}^{(m)} \in \mathbb{C}^{NK\times K}$ is the (complex) channel gain matrix, which is of interest to us. $\boldsymbol{a}^{(m)} \in \mathbb{C}^{K\times 1}$ is the transmitted signal vector of size K, and $\boldsymbol{w}$ is a noise vector, which contains elements of a complex Gaussian noise process, distributed according to $\boldsymbol{w} \sim \mathcal{CN}(0, \sigma^2)$. It has the same dimension as $\boldsymbol{y}^{(m)}$.

The channel gain matrix $\boldsymbol{H}^{(m)}$ has the form

$$\boldsymbol{H}^{(m)} = \begin{bmatrix} h_{1,1}^{(m)} & 0 & \cdots & 0 \\ 0 & h_{1,2}^{(m)} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & h_{1,K}^{(m)} \\ h_{2,1}^{(m)} & 0 & \cdots & 0 \\ 0 & h_{2,2}^{(m)} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & h_{2,K}^{(m)} \\ \vdots & \vdots & \vdots & \vdots \\ h_{N,1}^{(m)} & 0 & \cdots & 0 \\ 0 & h_{N,2}^{(m)} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & h_{N,K}^{(m)} \end{bmatrix} \tag{3.5}$$

For limitations on M and N, see Chapter 2. Furthermore, a vector for the channel frequency response is defined as $\boldsymbol{h}_n^{(m)} = \begin{bmatrix} h_{n,1}^{(m)} & h_{n,2}^{(m)} & \dots & h_{n,K}^{(m)} \end{bmatrix}^T$. As $\boldsymbol{h}_n^{(m)}$ is a vector with length K, a corresponding frequency vector is needed, which is defined in Chapter 2.

The channel matrix has to be estimated. This is done in the Labview code by an FPGA, in a similar fashion as described in [5, Ch 19, p.426]. Adjusted to the MIMO environment imposed by the Hardware, the mathematical description is as following: $\tilde{\boldsymbol{h}}_n^{(m)} = \boldsymbol{R}_{hh\mathrm{LS}} \boldsymbol{R}^{-1}_{h\mathrm{LS}h\mathrm{LS}} \boldsymbol{h}_n^{(m)\mathrm{LS}}$ and shows the LMMSE estimate of $\boldsymbol{h}_n^{(m)}$, called $\tilde{\boldsymbol{h}}_n^{(m)}$. $\boldsymbol{R}_{hh\mathrm{LS}}$ is the covariance matrix of channel gains and the LS estimate of channel gains, $\boldsymbol{R}_{h\mathrm{LS}h\mathrm{LS}}$ is the autocovariance matrix of the LS estimates and $\boldsymbol{h}_n^{(m)\mathrm{LS}}$ is the LS estimate of the channel. This LS estimate is obtained from the received signal vector $\boldsymbol{y}^{(m)}$ on layer m, using the known pilot sequence $\boldsymbol{a}^{(m)}$. I.e. we get the k-th element of $\boldsymbol{h}_n^{(m)\mathrm{LS}}$ from

$$\left[\boldsymbol{h}_n^{(m)\mathrm{LS}}\right]_k = \left[\boldsymbol{y}^{(m)}\right]_{(n-1)K+k} / \left[\boldsymbol{a}^{(m)}\right]_k \tag{3.6}$$

where $[\boldsymbol{x}]_k$ denotes the k-th element of $\boldsymbol{x}$.

Furthermore, assuming that noise falls in the category of additive white Gaussian noise (AWGN), for every subcarrier k, meaning the subcarriers are uncorrelated, $\boldsymbol{R}_{hh\mathrm{LS}} = \boldsymbol{R}_{hh}$ and $\boldsymbol{R}_{h\mathrm{LS}h\mathrm{LS}} = (\boldsymbol{R}_{hh} + \sigma^2 \boldsymbol{I}_K)$, with $\boldsymbol{I}_K$ being a unit matrix of size K.

Similar to [5, Ch 19, p.426] we define:

$$\boldsymbol{R}_{hh} = \mathbb{E}[\boldsymbol{h}_n^{(m)} \boldsymbol{h}_n^{(m)H}] \tag{3.7}$$

with $\mathbb{E}[\ ]$ denoting the expectation operator.

3.2.2 Time-Domain Signal Model

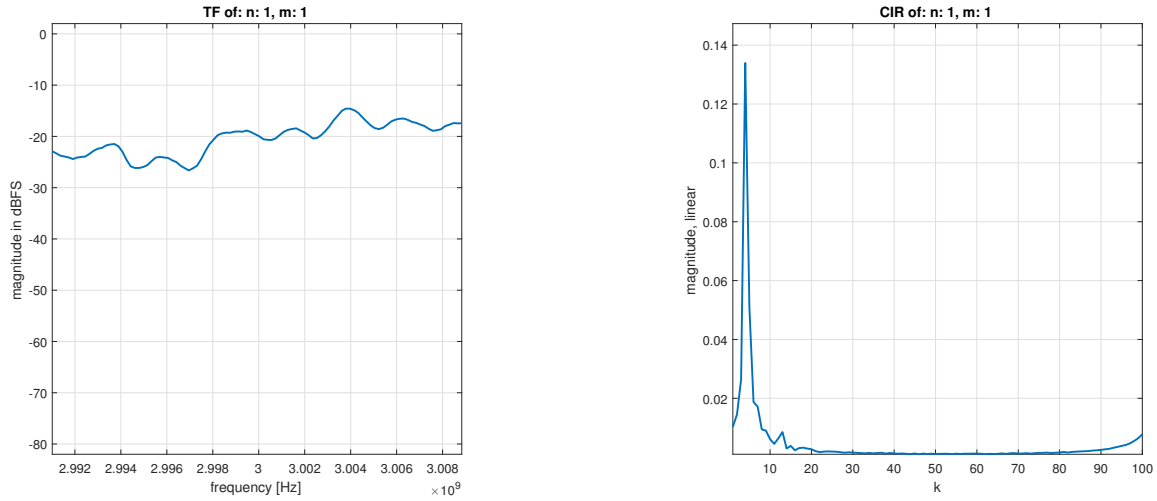
As a simplification, the dependence on m is neglected in further calculations. Also, the calibrated channel estimates $\boldsymbol{h}_n$ are transformed into the time-domain. This is not a necessary step, but allows for much easier visualization in the subsequent chapters. It is done by an inverse discrete Fourier transform (IDFT) operation:

$$\boldsymbol{r}_n = \mathcal{F}^{-1}\{\boldsymbol{h}_n\} \tag{3.8}$$

$\boldsymbol{r}_n$ can now be interpreted as a CIR, which is more comparable to other work done with a channel sounder [7]. Assuming a sum of pulses as a model for $\boldsymbol{r}_n$

$$\boldsymbol{r}_n = \sum_{p=1}^{P} a_p \mathbf{s}_n(\tau_p, \phi_p) \tag{3.9}$$

one can see the relation to the channel model (3.3). Note that a_p is not time-dependent anymore, and $\mathbf{s}_n(\tau_p, \phi_p)$ only depends on the corresponding delay and angle for a_p. If one can find an estimator finding the dominant component of (3.9), the delay τ_p of this can be used as an estimate for the delay, leading to the distance.



(a) TF

(b) CIR

Figure 3.2: TF and CIR of a single measurement from an outdoor environment

Figure 3.2(a) shows an exemplary TF for a single antenna and layer at one time. This corresponds to $\boldsymbol{h}_1$. Figure 3.2(b) shows the CIR of this, computed via (3.8). The main peak is assumed to correspond to the LoS component of the CIR.

The following signal model is calculated, depending on τ_p (for time of arrival (ToA))and ϕ_p (for angle of arrival (AoA)).

$$\mathbf{s}_n(\tau_p, \phi_p) = \mathcal{F}^{-1} \begin{bmatrix} \mathrm{e}^{-2j\pi f_1(\tau_p + \Delta\tau_n)} \\ \mathrm{e}^{-2j\pi f_2(\tau_p + \Delta\tau_n)} \\ \vdots \\ \mathrm{e}^{-2j\pi f_K(\tau_p + \Delta\tau_n)} \end{bmatrix} \tag{3.10}$$

$$\Delta\tau_n = \frac{x_n}{c} \cos(\phi_p) \tag{3.11}$$

Equation (3.10) describes the signal model for a single antenna, with length K. x_n used in (3.11) is defined for our geometry in Figure 2.3. $\mathbf{s}_n(\tau_p, \phi_p)$ is our signal atom, and can be described as a unit function shifted by the ToA τ_p and a vector $\Delta\boldsymbol{\tau} = \begin{bmatrix} \Delta\tau_1 & \Delta\tau_2 & \dots & \Delta\tau_N \end{bmatrix}^T$ depending on array geometry and AoA. $\Delta\boldsymbol{\tau}$ can be adapted to an arbitrary array geometry rather easily, but is skipped in this work. For arbitrary array geometries, see [5, Ch 9.3]. The signal model shown here is cyclic, which is a property of the IDFT used in (3.10). As our CIR retrieved from the channel estimation in (3.8) has the same mathematical operation applied, it is also cyclic. This is in contrast to a directly measured CIR from a channel sounder, but had to be taken in mind for the signal model as to retrieve reliable results.

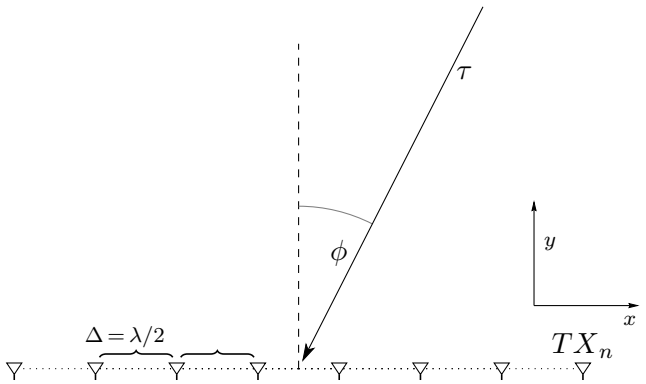


Figure 3.3: Array geometry, ULA.

Figure 3.3 shows the array geometry used in this thesis, consisting of eight antennas spaced $\lambda/2$ apart. The AoA ϕ is perpendicular when $\phi = 0$. TX_n shown here is the number of the antenna. From here it can be seen that the ToA depends on both the angle ϕ and and the position of the antenna in the ULA.

3.2.3 Likelihood of the Signal Model

Now, it is possible to calculate a likelihood function of the error between the CIR and the signal model:

$$p(\boldsymbol{r}_n; \tau, \phi) = \frac{1}{\pi^{KN} \det(\mathbf{C}_n)} \mathrm{e}^{-(\mathbf{r}_n - \alpha_n \mathbf{s}_n)^H \mathbf{C}_n^{-1} (\mathbf{r}_n - \alpha_n \mathbf{s}_n)} \tag{3.12}$$

describes the likelihood-function for our parameters. Here, $\mathbf{s}_n$ is the estimated signal from (3.10) parametrized by τ and ϕ, and $\mathbf{r}_n$ is the CIR, calculated from the channel estimates. Please note from (3.10) that τ and ϕ cannot be estimated from a single antenna measurement as described by (3.12). Multiple antenna signals are needed; see Section 4.2.3. This is analogous to [8], and needs the assumption that the signal model/estimate has the same first moment as the CIR, and thus the mean eliminates in the calculation.

For the estimation, a function which could be minimized had to be found. As an intermediate step, the amplitude of the estimated signal is estimated, which is as following:

$$\alpha_n = (\mathbf{s}_n^H \mathbf{s}_n)^{-1} \mathbf{s}_n^H \mathbf{r}_n \tag{3.13}$$

4 Algorithms and Measurements

4.1 Introduction

Implementation of the model described in Chapter 3 was done via *Matlab*, a programming language used often in engineering. In general, knowledge of the used programming language is not relevant for discussion of further points in this chapter. As an overview, the implementation consists of a part for pre-calibrating the data, an implementation of the equations used in Chapter 3, a grid search for finding the optimal parameters, and an output calibration as to offset systematic errors.

4.2 Algorithms

4.2.1 Frequencies and Parameters used for Measurements

The f_c used here was chosen because the used antennas are capable to transmit and receive at this frequency with a good attenuation, and the band is relatively unused. The other three parameters are imposed by the hardware/software from NI, and can't be changed easily. With the definition of the frequencies imposed by (2.1), the parameters are as following, for every further measurement:

f_c	f_η	f_λ	f_δ	N_{FFT}	M	K	N
3GHz	-9MHz	15kHz	180kHz	2048	12	100	8

Table 4.1: Frequencies and parameters used for measurements

The FFT Block Length N_{FFT} and the sample length K are fixed in length, and can't be changed without modifying significant parts of the Labview Code. The other parameters are imposed by the number of antennas (N) and the number of layers which are measured (M, altough only one was evaluated).

4.2.2 Data Import and Pre-Calibration

As the data recorded on the hardware from NI was saved in a proprietary format called *TDMS*, the first step in coding was to read data from this format. This was achieved by using code from [9]. The data was pre-processed, seen in (4.1), as to get an appropriate format for further calculations.

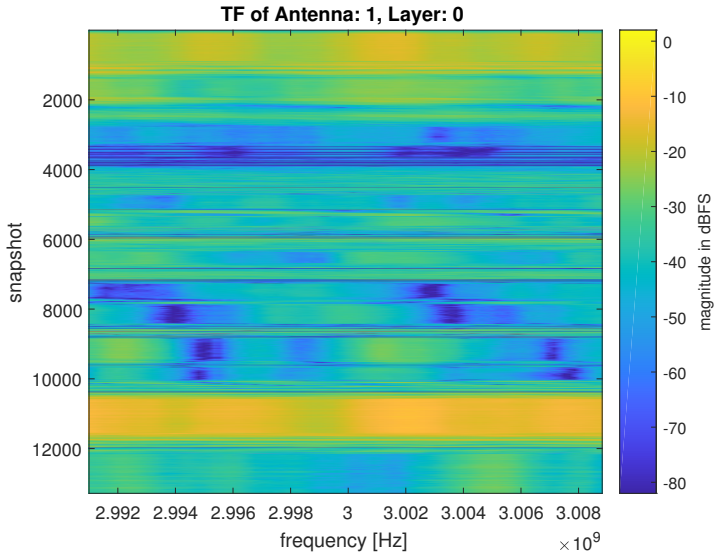


Figure 4.1: TF of one antenna and layer over multiple snapshots

As seen in Figure 4.1, the estimates of the TFs were saved for every snapshot. The first few snapshots showed no data for estimation in this figure because the MS corresponding to layer 1 was not active until a few snapshots later. The shown data corresponds to a measurement done indoors and with cable synchronization, which both will be explained later in this chapter.

After the import of the TFs, (4.4) was implemented. For this, the parameter ϕ_{cal} (see (4.1)) had to be estimated, which was done via taking a look at the data and trying a few values. This calibration value was constant over a measurement, and varied only slightly between different measurements.

For further calculations, these TFs were converted via (3.8) into CIRs.

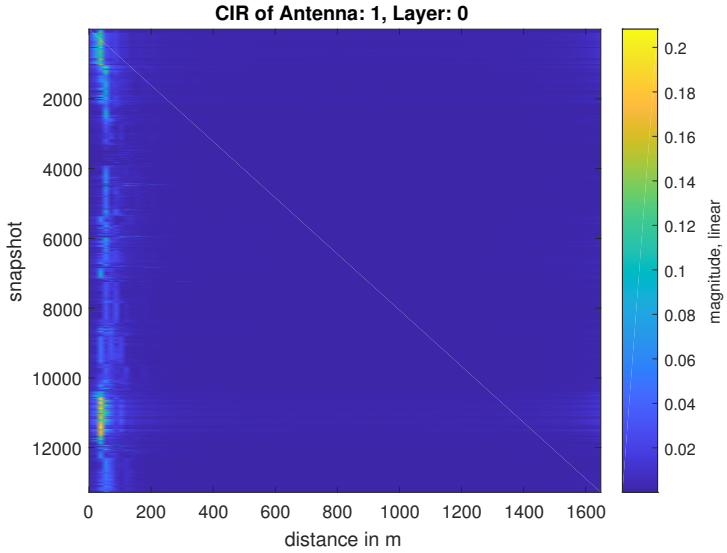


Figure 4.2: CIR of one antenna and layer over multiple snapshots

In Figure 4.2 the corresponding CIRs are shown, which show peaks in the areas of lower distances. This is due to distances in this example being small. For easier interpretation, the time-delay

was converted into a distance.

To compensate the channel estimation being systematically shifted in a direction, which is caused by the system or software from NI, pre-calibration had to be done in the frequency domain to compensate this error.

$$\Phi_k^{(m)} = f_k^{(m)} \frac{\lambda_c}{c} \phi_{cal} \tag{4.1}$$

$\Phi_k^{(m)}$ is a calibration value, which is depending on the frequency values $f_k^{(m)}$, which in turn depend on the layer m. λ_c is the carrier wavelength. ϕ_{cal} is the relevant part of the equation, and is a constant for calibration, different for every measurement, but constant within the measurement.

$$\Psi_n^{(m)}[k] = \mathrm{e}^{-2j\Phi_k^{(m)}(n-1)} \tag{4.2}$$

$$\boldsymbol{\Psi}_n^{(m)} = \begin{bmatrix} \Psi_n^{(m)}[1] & \Psi_n^{(m)}[2] & \dots & \Psi_n^{(m)}[K] \end{bmatrix} \tag{4.3}$$

$$\boldsymbol{h}_n^{(m)} = \boldsymbol{\Psi}_n^{(m)} \tilde{\boldsymbol{h}}_n^{(m)} \tag{4.4}$$

Equations (4.1), (4.2) and (4.4) describe the necessary pre-calibration. In principle, the calibration is equivalent to a beamformer, as described in [5, Ch 19, p.160]. The value n denotes the antennas in the ULA, with $n = 1$ being the first antenna. The equation is simplified because the spacing of antennas is assumed to always be $\lambda/2$ apart. As one can deduct from (4.2), there is a dependency on the number of the antennas. $\boldsymbol{h}_n^{(m)}$ are the calibrated channel estimations for every antenna and layer.

4.2.3 Distance and Angle Estimation

For the covariance-matrix $\mathbf{C}_n$, seen in (3.12), one can assume that the noise variance is equal over all subcarriers, so the covariance-matrix $\mathbf{C}_n$ can be written as $\mathbf{C}_n = \sigma^2 \boldsymbol{I}_K$, where $\boldsymbol{I}_K$ is the unit matrix of size K. As σ^2 and everything before the exponential seen in (3.12) can be treated as constant and not relevant for the optimization problem, and multiplying all antennas, this leads to

$$p(\boldsymbol{r}; \tau, \phi) = \prod_{n=1}^{N} p(\boldsymbol{r}_n; \tau, \phi) \propto \mathrm{e}^{\sum_{n=1}^{N} -(\mathbf{r}_n - \alpha \mathbf{s}_n)^H (\mathbf{r}_n - \alpha \mathbf{s}_n)} \tag{4.5}$$

being the function we would like to maximize, in order to solve the estimation problem. This equation further needs the assumption that antennas are uncorrelated and identically distributed, and thus $\mathbf{C}_n$ is equal for every antenna. For easier calculation, the log-likelihood is used, which is the natural logarithm of (4.5). This yields

$$\tilde{p}(\boldsymbol{r}; \tau, \phi) = -\ln\left(\mathrm{e}^{-(\mathbf{r} - \alpha \mathbf{s})^H (\mathbf{r} - \alpha \mathbf{s})}\right) \tag{4.6}$$

which can be further simplified to

$$\tilde{p}(\boldsymbol{r}; \tau, \phi) = (\mathbf{r} - \hat{\alpha} \mathbf{s})^H (\mathbf{r} - \hat{\alpha} \mathbf{s}) \tag{4.7}$$

an equation which can be implemented easily in software. $\mathbf{r}$ here is the received signal vector, defined as a stack of all received signals $\mathbf{r} = (\mathbf{r}_1^T, \mathbf{r}_2^T, \dots, \mathbf{r}_N^T)^T$, and $\mathbf{s}$ is our model for the signal, also stacked as $\mathbf{s} = (\mathbf{s}_1^T, \mathbf{s}_2^T, \dots, \mathbf{s}_N^T)^T$. $\hat{\alpha}$ seen here is the unique amplitude across all antennas,

which is needed for a joint estimation of τ and ϕ using the models from Chapter 3. It is obtained from $\hat{\alpha} = (\mathbf{s}^H\mathbf{s})^{-1}\mathbf{s}^H\mathbf{r}$.

Lastly, to get an estimate of the parameters τ called $\hat{\tau}$ and ϕ called $\hat{\phi}$, function (4.6) is minimized:

$$(\hat{\tau}, \hat{\phi}) = \underset{\tau,\phi}{\operatorname{argmin}}\, \tilde{p}(\boldsymbol{r}; \tau, \phi) \tag{4.8}$$

This is only valid for the LoS path of the signal. If further MPCs are to be estimated, the p-th path can be estimated via recursively calculating:

$$(\hat{\tau}_p, \hat{\phi}_p) = \underset{\tau_p,\phi_p}{\operatorname{argmin}}\, \tilde{p}(\boldsymbol{r} - \sum_{a=1}^{p-1} \boldsymbol{s}(\hat{\tau}_a, \hat{\phi}_a)) \tag{4.9}$$

The ML estimation was done via implementing (4.8). This was done using *Matlab* and a grid search.

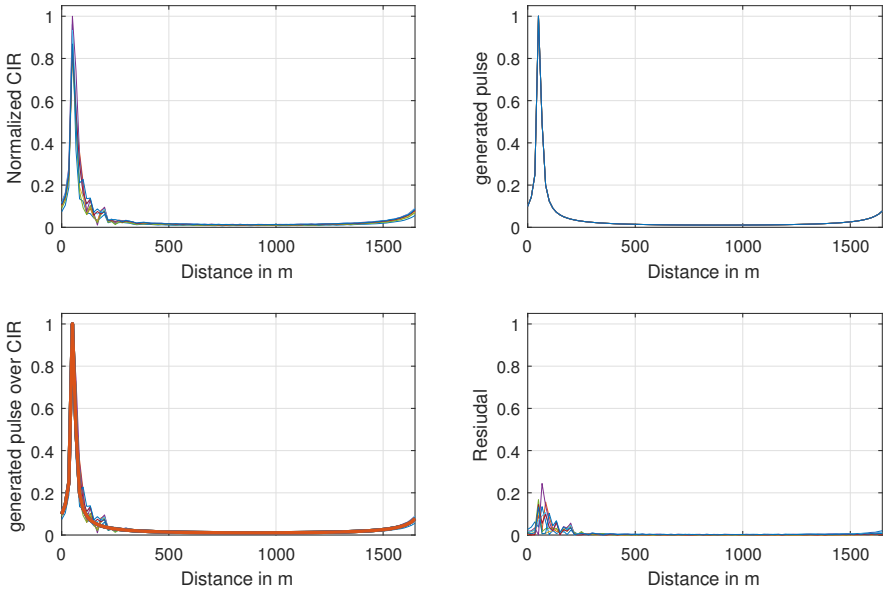


Figure 4.3: CIR of all antennas, generated signal, matched generated signal and after substracting the estimated pulse, yielding the residual

Figure 4.3 shows a single (normalized) CIR for all 8 antennas in the upper left. The upper right depicts a generated pulse with (3.10). The cyclic properties of the measured and the generated pulse can be seen in detail in these sub-figures. The lower left of Figure 4.3 shows the generated pulse (in red) layered over the pulse achieved via (3.10). Note that the generated pulse here was already matched using the estimator with the signal. Lastly, the residual of the measurement is shown, which is the received signal minus the previous estimated signal. The residual was not used for further calculations, but it is illustrated to show the error of estimation.

The parameter estimation itself is done via a grid search over ϕ and τ, yielding joint estimates $\hat{\phi}$ and $\hat{\tau}$. The grid was chosen depending on the geometry, meaning this is prior knowledge from floorplans and the environment. Used grids for the measurements discussed in this thesis were:

Measurement	ϕ_{range}[°]	ϕ_{step}[°]	d_{range}[m]	d_{step}[m]
Outdoor	-80 ... +80	0.5	40 ... 80	0.5
Indoor without Sync.	-80 ... +80	0.5	40 ... 120	0.5
Indoor with Sync.	-80 ... +80	0.5	52.5 ... 77.5	0.1

Table 4.2: Grids used for estimation

In Table 4.2, values for ϕ_{range} are the upper and lower search boundaries for the angle ϕ, values for d_{range} are the upper and lower search boundaries for the distance $d = \tau c$, where τ is the delay defined in Chapter 3, and ϕ_{step}, d_{step} are the step sizes used in grid search for the respective parameters. These values were chosen as to minimize computation time, but still yield good results for the estimation. The grid search was also done recursively three times, with the first grid having bigger step sizes, yielding a minimum over the whole range, then an estimate of the angle ϕ and distance d was found in the area around the first minimum as a second run, and lastly the estimate for the distance was found in an even finer grid, but the angle was left at the value found in the second run. This grid search was done for every snapshot of the three measurements, yielding estimates $\hat{\phi}$ and $\hat{d}$. The code shown here next, is a simplified pseudo-code for this. For further code, see the appendix.

```
min = inf;
for i = 1:1:num_samples
  %first run
     dist_est_grid = 40:5:80;
     alpha_est_grid = -80:2:80;

     for k = 1:1:length(dist_est_grid)
          for m = 1:1:length(alpha_est_grid)
               minvar = estimator_joint_ml_aoa_toa([alpha_est_grid(m) dist_est_grid(k)],y);
               if minvar < min
                    min = minvar;
                    minplace = [m, k];
                    dist_min(i) = dist_est_grid(k);
                    alpha_min(i) = alpha_est_grid(m);
               end
          end
     end
     min = inf;

  %second finer run
     dist_est_grid = (dist_min(i)-2.5):1:(dist_min(i)+2.5);
     alpha_est_grid = (alpha_min(i)-(1*pi/180)):(0.5*pi/180):(alpha_min(i)+(1*pi/180));

     for k = 1:1:length(dist_est_grid)
          for m = 1:1:length(alpha_est_grid)
               minvar = estimator_joint_ml_aoa_toa([alpha_est_grid(m) dist_est_grid(k)],y);
               if minvar < min
                    min = minvar;
                    minplace = [m, k];
                    dist_min(i) = dist_est_grid(k);
                    alpha_min(i) = alpha_est_grid(m);
               end
          end
     end
     min = inf;

  %third run, only for distance
     dist_est_grid = 40:0.5:80;
     alpha_est_grid = alpha_min(i);

     for k = 1:1:length(dist_est_grid)
          for m = 1:1:length(alpha_est_grid)
               minvar = estimator_joint_ml_aoa_toa([alpha_est_grid(m) dist_est_grid(k)],y);
               if minvar < min
                    min = minvar;
                    minplace = [m, k];
                    dist_min(i) = dist_est_grid(k);
                    alpha_min(i) = alpha_est_grid(m);
               end
          end
     end
     min = inf;
end
```

Listing 4.1: pseudo-code of the grid search

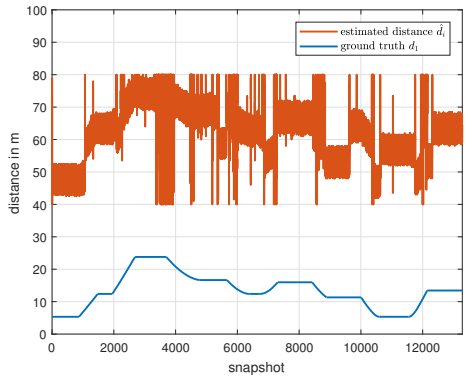
4.2.4 Post-Calibration of Estimates

As the values estimated have a systematic shift for distance and angle estimation, some post-processing had to be done to achieve viable estimates. This post-processing consisted of a simple calibration to a known good sample. For this, a ground truth of the data for the algorithms was measured in person by hand. This data was written into code, and as times of movement were known (via videos of the measurement), absolute positions of stand-still and linearly interpolated positions of movement were evaluated to achieve a ground truth for ϕ and d of the measurements. The reason for these shifts is a delay in the code from NI, which leads to a shift in synchronization. Calibration itself was done via:

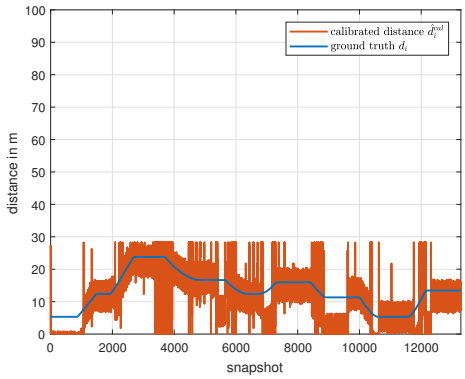
$$\hat{d}_i^{cal} = \hat{d}_i - d_{ref} \tag{4.10}$$

$$\hat{\phi}_i^{cal} = \hat{\phi}_i - \phi_{ref} \tag{4.11}$$

where d_{ref} and ϕ_{ref} are obtained from the ground truth measurements of the MS positions, $\hat{d}_i$ and $\hat{\phi}_i$ are the estimates for every snapshot i.



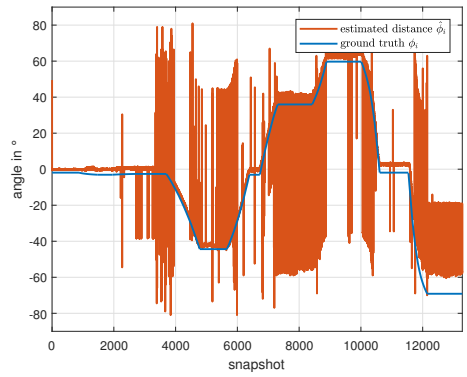
(a) Uncalibrated distance estimate and ground truth



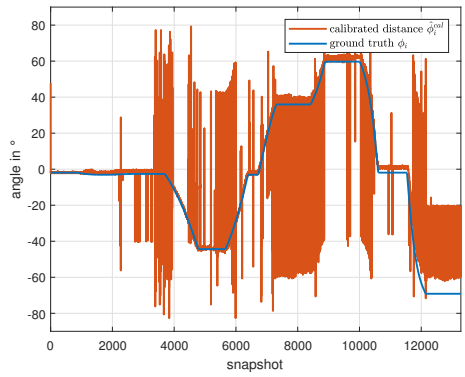
(b) Calibrated distance estimate and ground truth

Figure 4.4: Uncalibrated and calibrated distance estimates and ground truth for one measurement

Figure 4.4 shows the calibration process of (4.10). The shown measurement is from the outdoor measurement.



(a) Uncalibrated angle estimate and ground truth



(b) Calibrated angle estimate and ground truth

Figure 4.5: Uncalibrated and calibrated angle estimates and ground truth for one measurement

Figure 4.5 shows the calibration process of (4.11). The shown measurement is from the outdoor measurement. As the pre-calibration done previously already corrects for most of the angular error, the shift in angle from the calibration can not be seen this easily. Further effects seen in this estimation will be discussed in an individual section for the measurement.

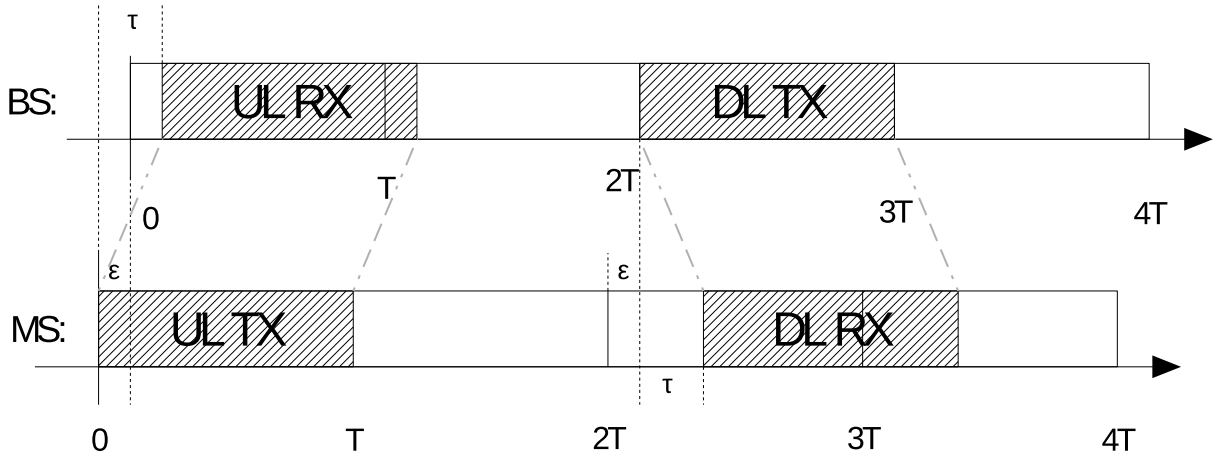


Figure 4.6: Transmit and receive scheme of BS and MS

Figure 4.6 describes the transmission scheme between BS and MS. This explains why the calibration is needed. Assuming time t for the BS and time $t' = t + \epsilon$ for the MS, one can see that the receive time at the uplink is $t = \tau - \epsilon$. This leads to a measured delay of $\tau - \epsilon$ at the uplink. The receive time at the downlink, for the MS equals $t' = 2T + \tau + \epsilon$, where T is the packet duration.

As the BS accounts for the delay at the uplink (meaning that $-\tau + \epsilon$ is added to the transmit time), this leads to a measured delay at the downlink receive of $t = (-\tau + \epsilon) + \tau$ or $t' = 2\epsilon$, which can be used as an estimate for the synchronization offset ϵ.

The excess delay $-\epsilon$ is the reason for the post-calibration.. As ϵ can not be computed exactly without synchronized measurement data from the MS, the solution using a reference distance was chosen.

4.3 Measurement 1 – Outdoor

4.3.1 Environment and Conditions

This measurement was performed outdoors, as to minimize the influence of multipath propagation by being some distance away from surrounding buildings. The place chosen is a small park at one side of the building *Inffeldgasse 16c*, on the campus *Inffeldgasse* of *TU Graz*. The relative position is shown in Figure 4.7 by a blue rectangle.

Figure 4.7: Position of measurement at campus Inffeldgasse, adapted from [10]

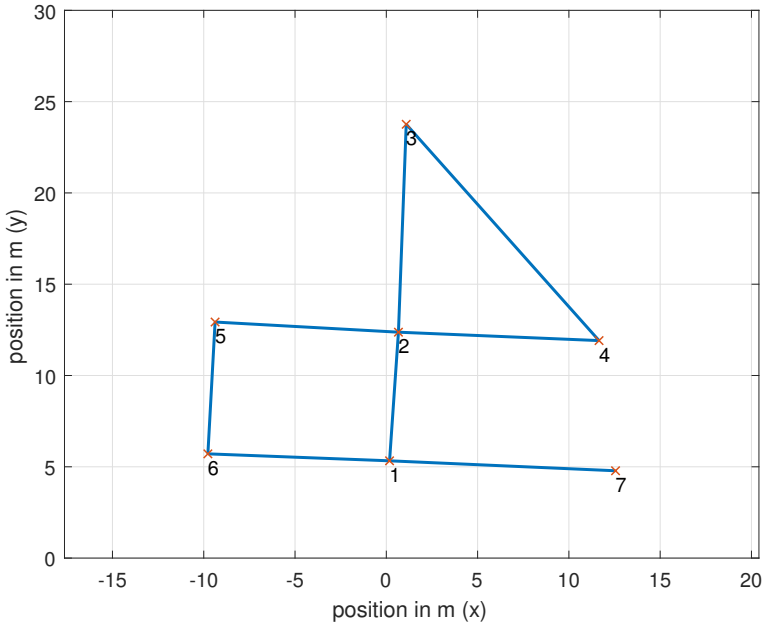


Figure 4.8: True path of measurement 1, BS is at position (0,0)

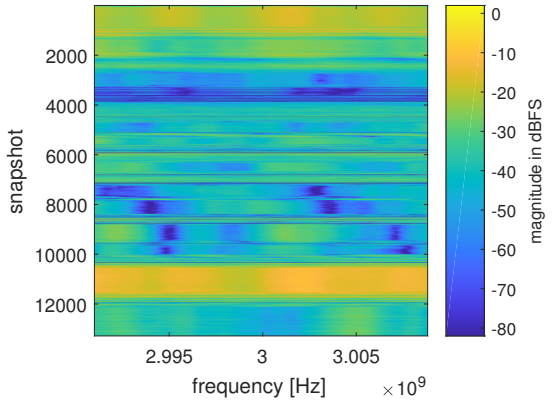
Figure 4.8 shows a bird's-eye view of the true path taken by the MS during this measurement. Numbers from 1 to 7 depict different positions where the MS was stationary. When the MS was moved, it moved from number to number in ascending order, with two intermediate stops at number 2 between 4 and 5, and number 1 between 6 and 7. Position (0,0) is the reference point (center) of the ULA, the x axis is parallel to the ULA.

The measurement was done without cable synchronization between BS and MS, because the distance between these was too large. The effects of the cable synchronization are examined in the indoor measurements.

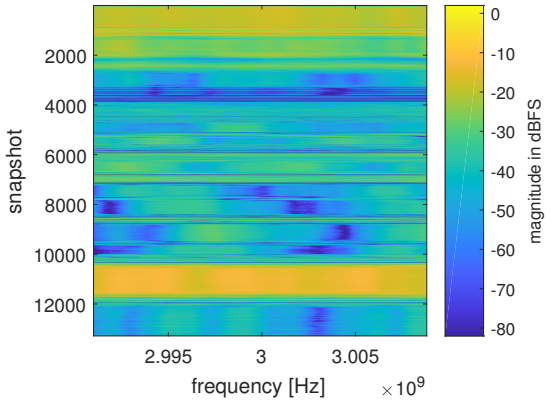
4.3.2 Evaluated TFs and CIRs

The following Figures 4.9 and 4.10 show the TFs of all 8 antennas of the BS in Figure 4.9 and the corresponding CIRs in Figure 4.10. In Figure 4.9, one can see the magnitude of the TFs changing over time. This is due to the distance between BS and MS changing, and thus with greater distance the magnitude is decreasing and vice versa. Also, for example at approximately snapshot 8000, channel fading can be seen.

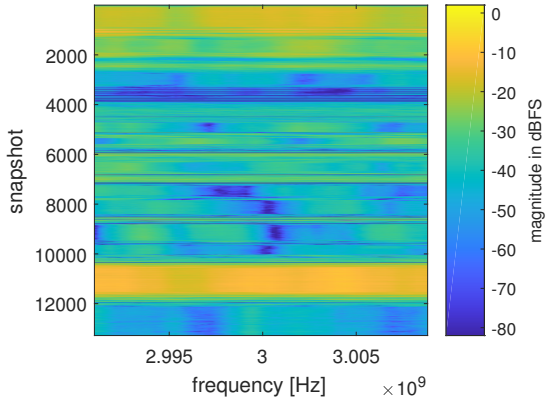
Translated to Figure 4.10, one can see that the CIRs still have their biggest peaks at low distances. These peaks correspond to wave-fronts arriving at the BS antennas.



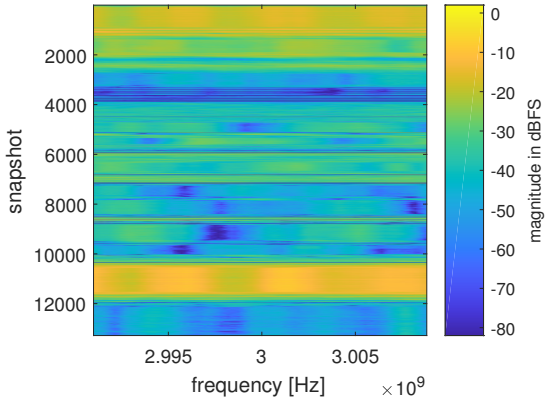
(a) TF of antenna 1, layer 0



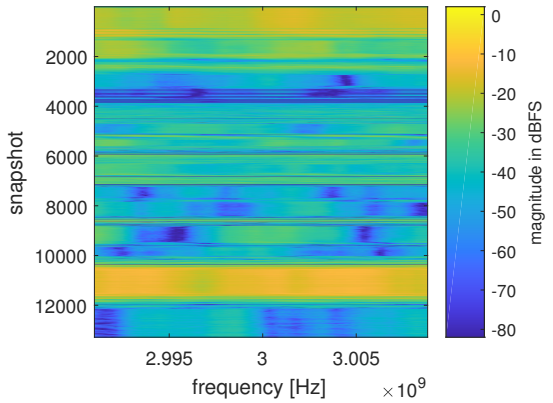
(b) TF of antenna 2, layer 0



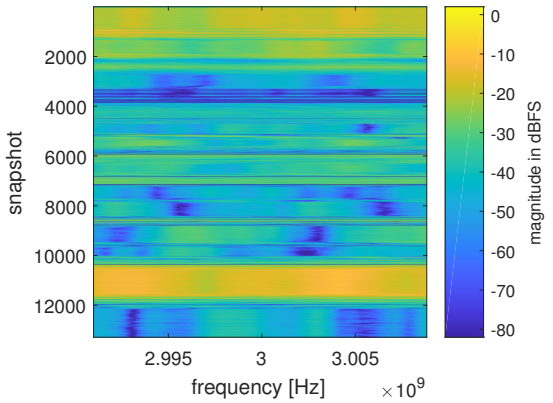
(c) TF of antenna 3, layer 0



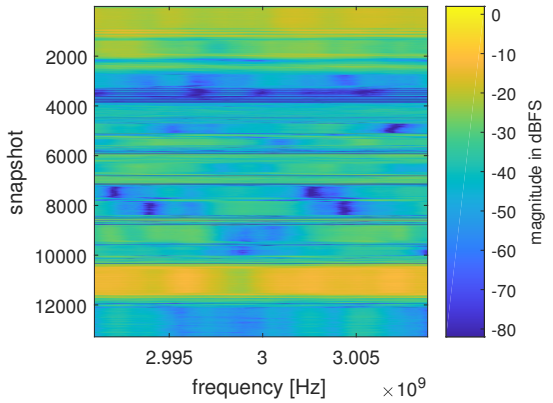
(d) TF of antenna 4, layer 0



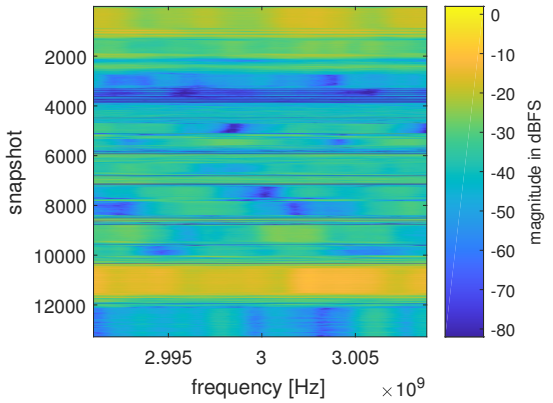
(e) TF of antenna 5, layer 0



(f) TF of antenna 6, layer 0

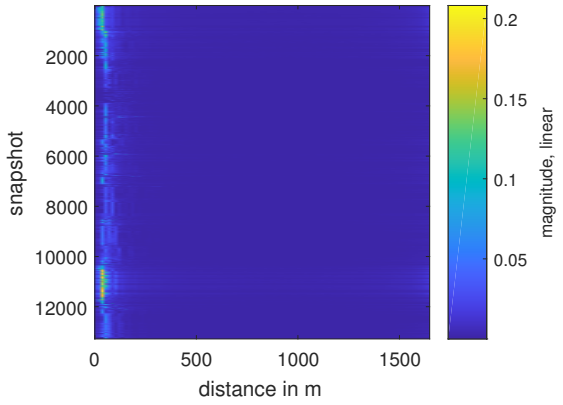


(g) TF of antenna 7, layer 0

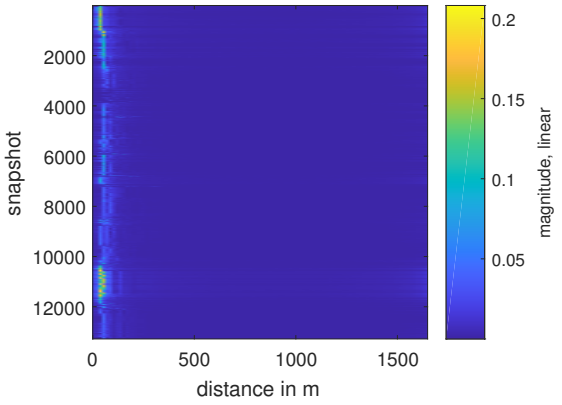


(h) TF of antenna 8, layer 0

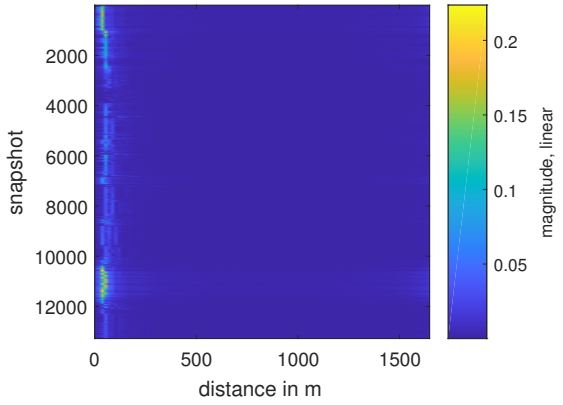
Figure 4.9: TFs of all 8 antennas and layer 0 for measurement 1



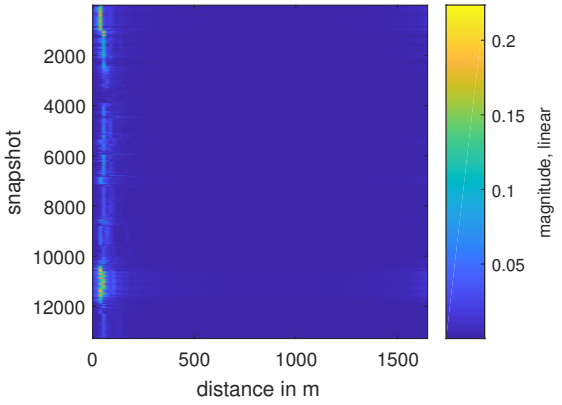
(a) CIR of antenna 1, layer 0



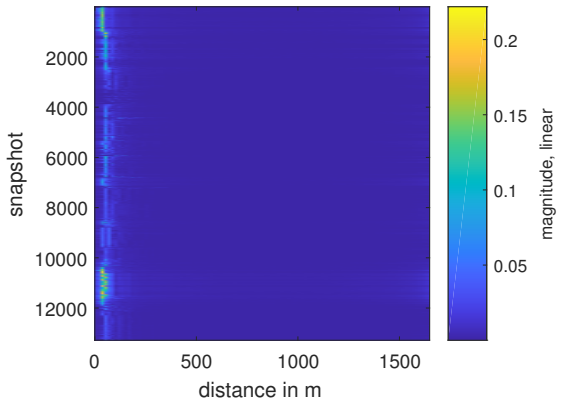
(b) CIR of antenna 2, layer 0



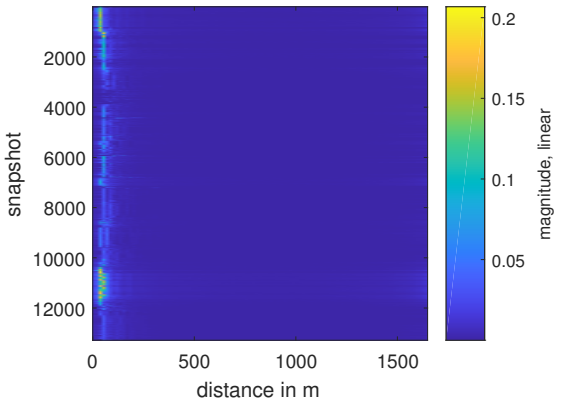
(c) CIR of antenna 3, layer 0



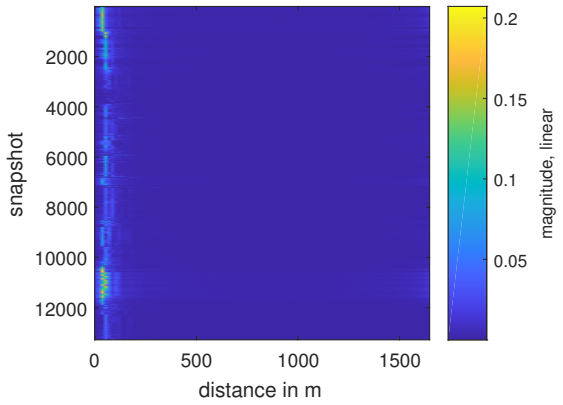
(d) CIR of antenna 4, layer 0



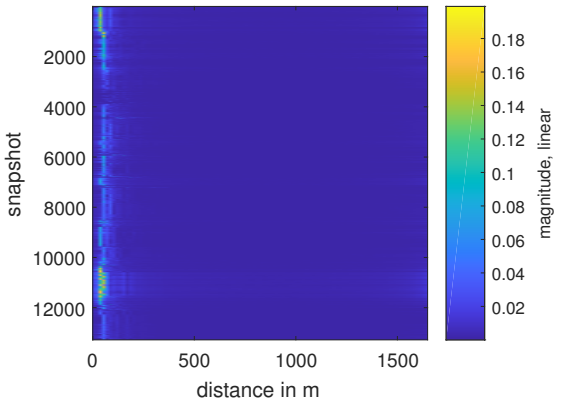
(e) CIR of antenna 5, layer 0



(f) CIR of antenna 6, layer 0



(g) CIR of antenna 7, layer 0



(h) CIR of antenna 8, layer 0

Figure 4.10: CIRs of all 8 antennas and layer 0 for measurement 1

Within the measurement done here, a second layer (corresponding to the second antenna on the MS) was measured, but will not be discussed. As the CIRs shown within Figure 4.10 do not show great detail for smaller distances, a further plot is shown here for illustration:

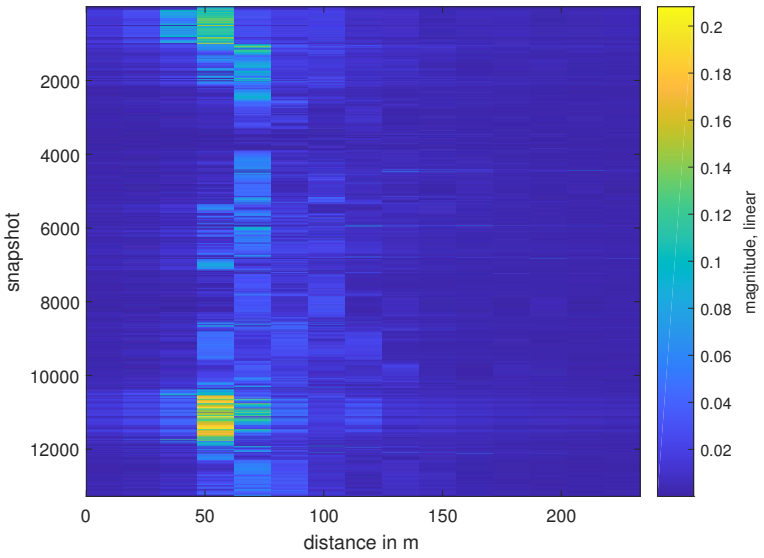


Figure 4.11: CIR of antenna 1, layer 0, measurement 1, detail view

Figure 4.11 shows that the main peaks of the signal occur for distances between about 40 and 80m. This explains why the search parameters for the grid search were chosen as seen in Table 4.2. Other smaller peaks can be seen, which correspond to reflections from the surrounding buildings.

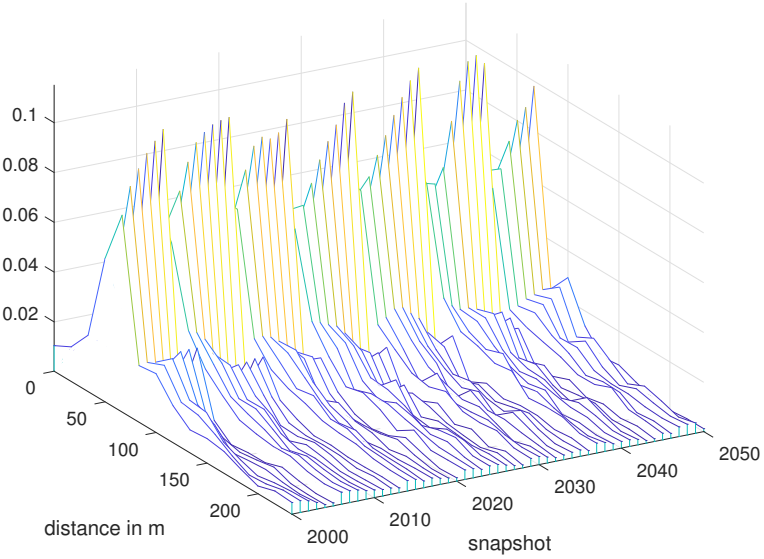


Figure 4.12: CIR of antenna 1, layer 0, measurement 1, detail view, waterfall plot

Figure 4.12 shows an additional view of the CIRs, with less snapshots. Here, the problems with synchronization can be seen rather well, causing variation in the peaks of the CIRs with some periodicity.

4.3.3 Amplitude of the TFs

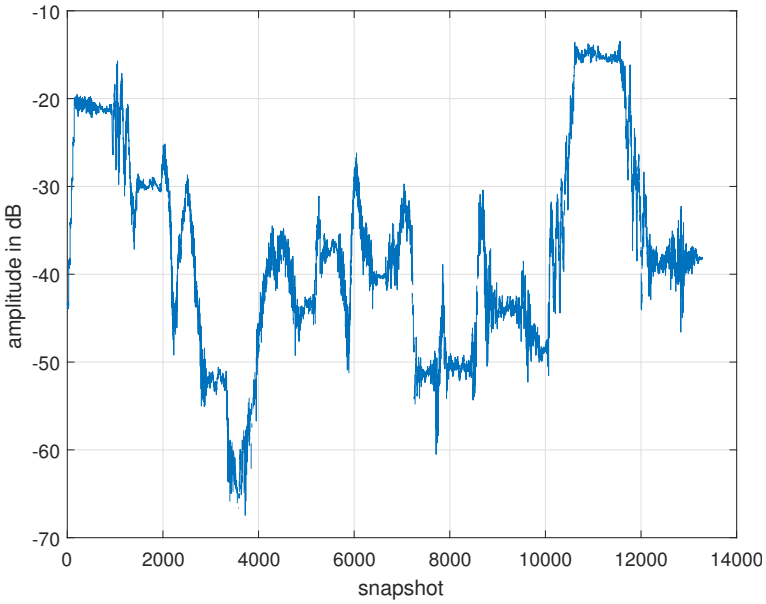


Figure 4.13: Mean amplitude from TFs of antenna 1, layer 0, measurement 1

Figure 4.13 shows the mean amplitude of the TFs, defined as $\sum |\mathbf{h}_n^2|$, which correlates to some degree to the estimated distance. This was not used further in this work, and is only for illustration.

4.3.4 Estimates

As one can see from Figure 4.15(a), calibration (as explained in (4.10)) was a necessity for this measurement. In Figure 4.15(b) some effects of the hardware/software from NI can be seen: The distance estimation seems to jump between two whole samples of the CIR best seen at the snapshots at the beginning, jumping at snapshot of approximately 1000 to the correct values. Also, the estimates seem to be distorted by noise. Both of these problems can be interpreted to come from inaccurate synchronization of the sampling time between BS and MS, because these effects are not visible when the system is synchronized via cable, as seen in measurement 3. The distortion between snapshots can be mitigated by a simple smoothing filter, in this case computing the mean of 101 snapshots, as seen in Figure 4.15(c).

This can still not resolve the big jumps between whole samples of the CIR. As can be seen, between snapshots of approximately 3500 to 4000, the distance estimate is not reliable. Here, the magnitude of the TFs was too small for a reliable estimation, and the grid search did not converge. This was at larger distances between BS and MS, so it is reasonable to assume that this effect is attributable to small signal-to-noise ratios (SNRs). Also, the BS had just turned around before this happened (position 3), as can be seen in some still-frames in the appendix. It could be possible that there was shadowing of the other antenna before, and then after turning shadowing was less of a problem.

Another problem seen here is that the estimation was getting less accurate over time. An interpretation for this is that measurements were done outdoors in August, when it was exceptionally hot for Austria. As the sun was shining brightly on the SDR, one can assume that it was heating up while measuring, affecting for example amplifiers in the SDR. As a rise in temperature also

affects the coefficients of the reciprocity calibration (see Chapter 2), but this calibration only is done once when starting the measurement, the estimation of the TF is less accurate with rising temperature [11, Ch 5].

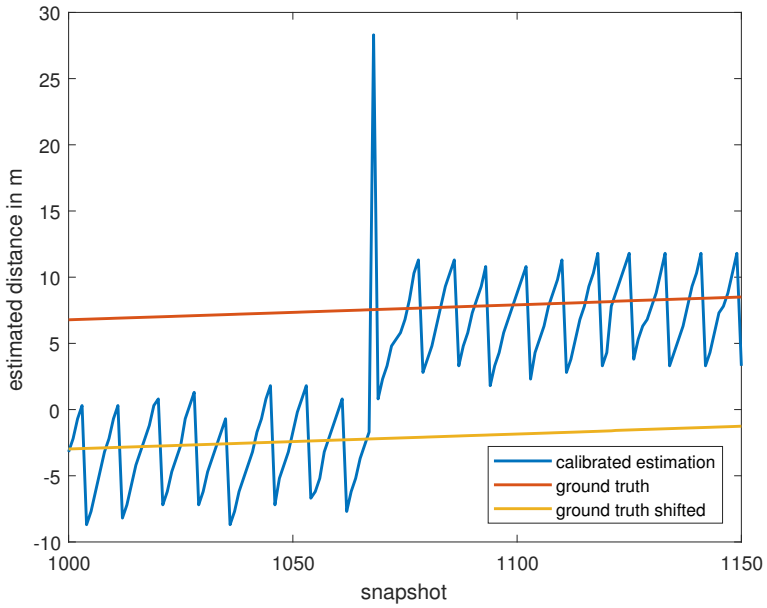
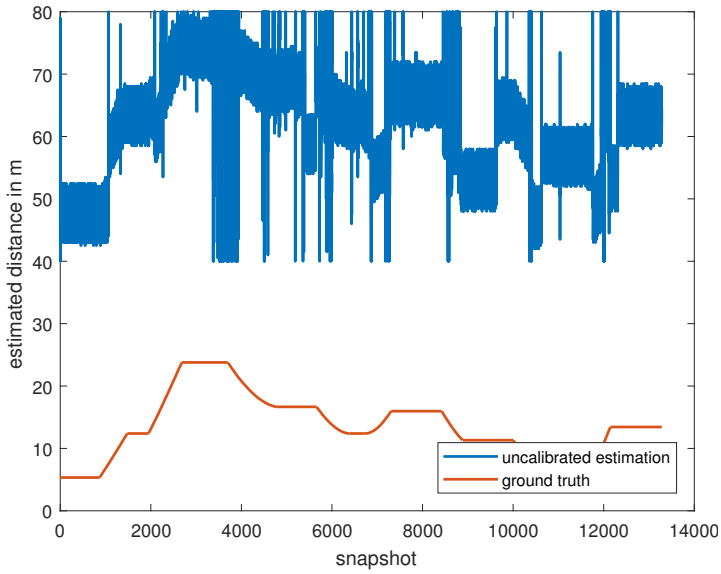


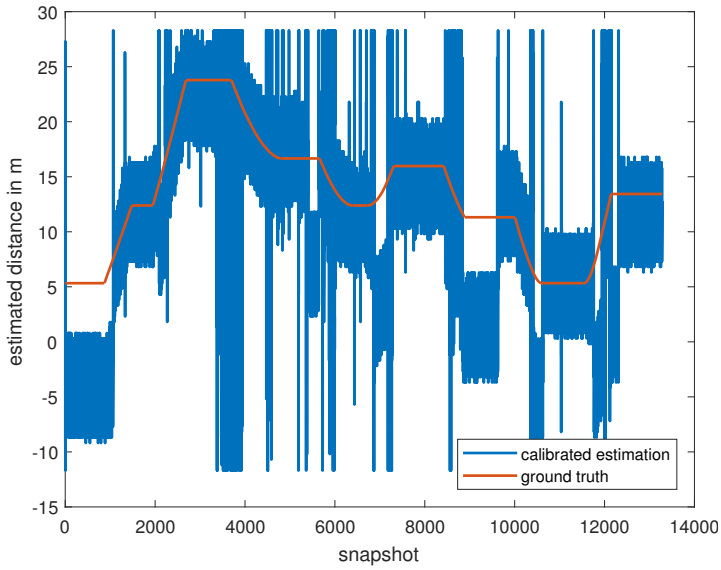
Figure 4.14: Distance estimation of measurement 1, calibrated, detail view

Figure 4.14 shows a detail view of a jump between two samples, at approximately snapshot 1070. The ground truth is in there for reference, ground truth shifted is the ground truth minus the error one clock cycle offset would have. This comes from an internal sampling rate of the *USRP-RIOs* of $f_{sampling} = 30.72\text{MHz}$, leading to an offset for one clock cycle of $c/f_{sampling} = 9.76\text{m}$.

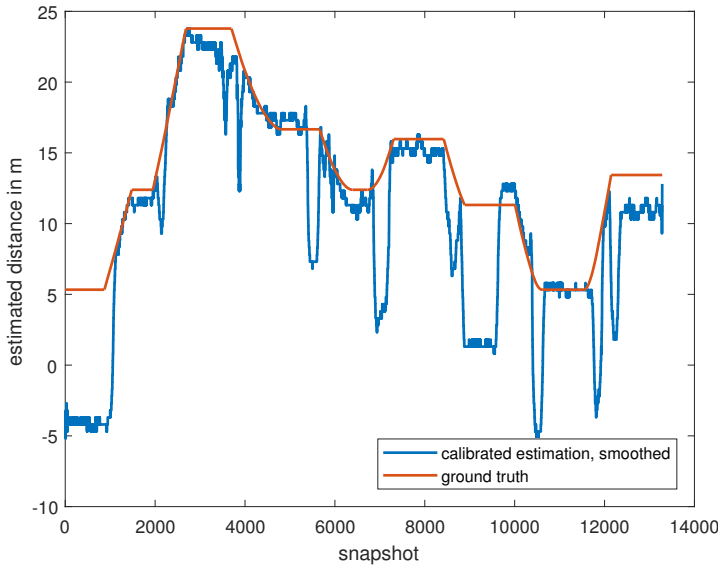
With angle estimation, similar effects can be seen as with distance estimation. Altough, jumping between samples of the CIR is not a relevant problem here, as estimation of the angle is dependent on the relative ToAs between antennas, which are not affected by synchronization. Calibration, as seen happening between Figure 4.16(a) and Figure 4.16(b) is not doing as much as for distance, as pre-calibration already resolves most of the error in angle. But another effect can be seen well after smoothing in Figure 4.16(c): There are systematic jumps to false values, as for example before snapshot 8000. As the antenna array can be seen as a beamformer depending on antenna spacing and number of antennas (as seen in [12, p. 529]), side lobes in angular signal energy are to be expected. These sometimes overshadow the main lobe in estimation, and thus lead to systematic false estimates sometimes. This also happens after snapshot 12000, where even jumps between two lobes can be seen.



(a) Distance estimation of measurement 1, uncalibrated

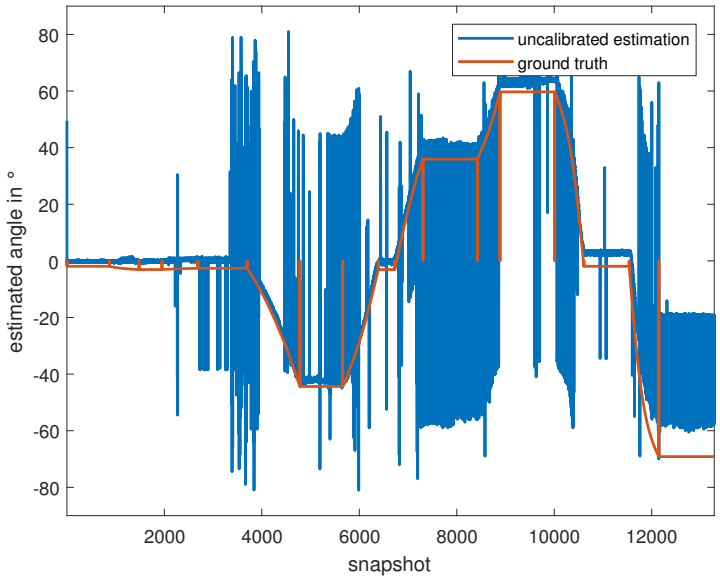


(b) Distance estimation of measurement 1, calibrated

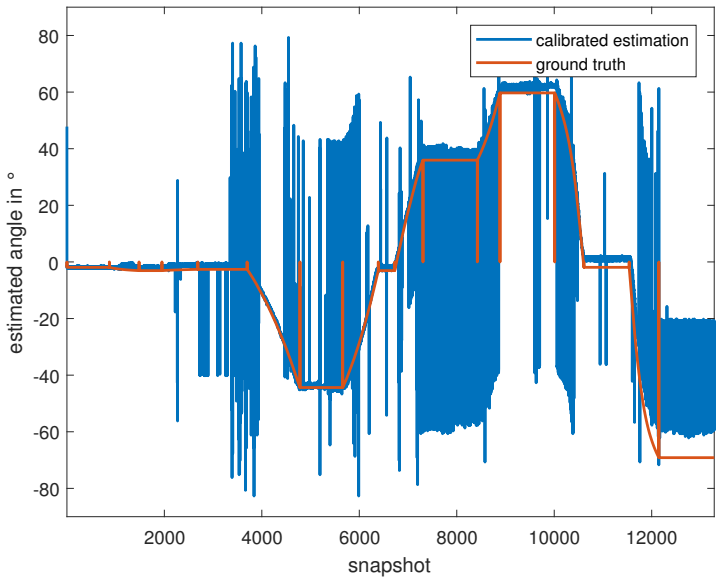


(c) Distance estimation of measurement 1, calibrated and smoothed

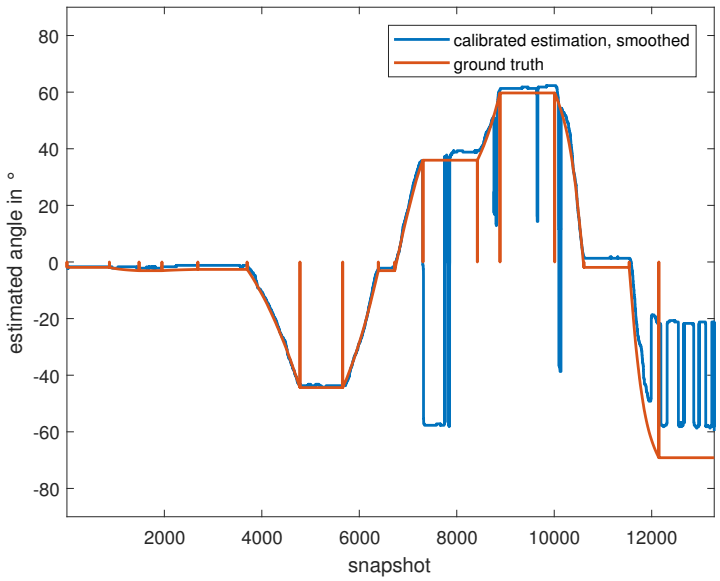
Figure 4.15: Distance estimates, uncalibrated, calibrated and smoothed over ground truth



(a) Angle estimation of measurement 1, uncalibrated



(b) Angle estimation of measurement 1, calibrated



(c) Angle estimation of measurement 1, calibrated and smoothed

Figure 4.16: Angle estimates, uncalibrated, calibrated and smoothed over ground truth

4.4 Measurement 2 – Indoor without Synchronization

4.4.1 Environment and Conditions

This measurement was done indoors, in the building *Inffeldgasse 16c*, the building marked in red in Figure 4.7. As the measurement was done indoors, it was a logical conclusion to partake the measurement in the largest space available. This was the corridor, stretching most of the building.

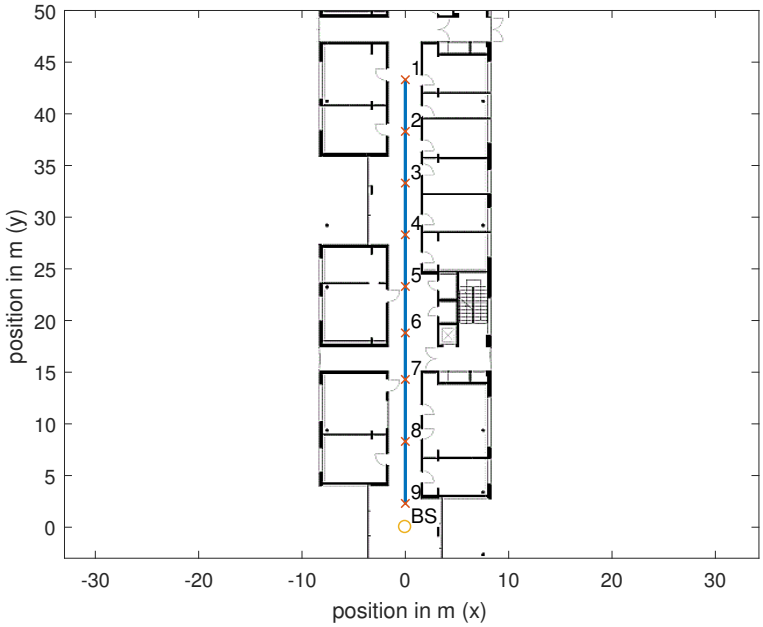
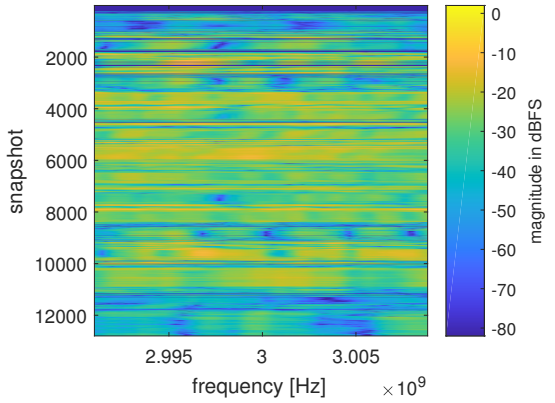


Figure 4.17: True path of measurement 2

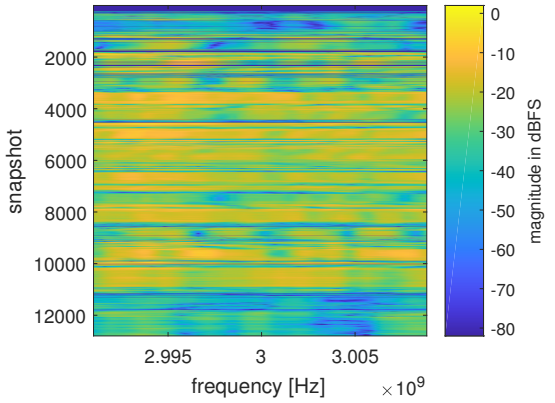
Figure 4.17 shows the true path within the building, relative to the BS. It should be noted, that the angle to the BS ϕ was always 0 for this measurement. The path taken was from mark 1 to mark 9 and back, with intermediate stops at every mark except at the BS itself. The corridor has largely varying width (as can be seen in the Figure) and height (as the corridor stretches over 3 floors, with passages over the corridor), and thus the corridor allows for diverse MPCs. For this measurement, again, no cable-synchronization was used, as distances were again too far. For positions 5 to 9, a measurement with cable-synchronization was done, which will be shown in measurement 3.

4.4.2 Evaluated TFs and CIRs

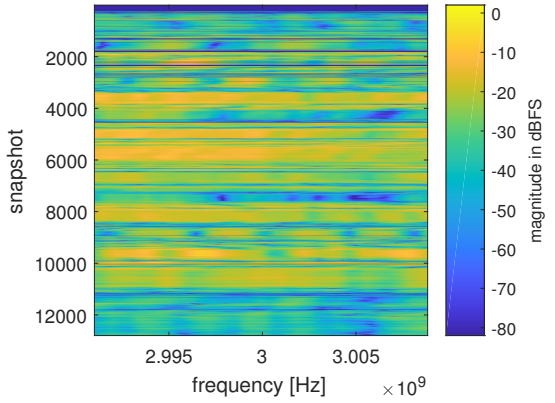
In Figures 4.18 the TFs for the indoor measurement without cable-synchronization are shown. In general, they are comparable to the outdoor TFs, but show much more fading over the bandwidth. For example, this can be seen well at snapshot 9000. In the beginning of the measurement, until around snapshot 300, the channel estimation didn't seem to work. This could be due to a low SNR, as the distance was at a maximum for this measurement. The Figures 4.19 are of more interest here, as several MPCs can be seen as additional (smaller) peaks in the CIR. These could be attributed to reflections from walls etc. in the measurement environment, i.e. the corridor.



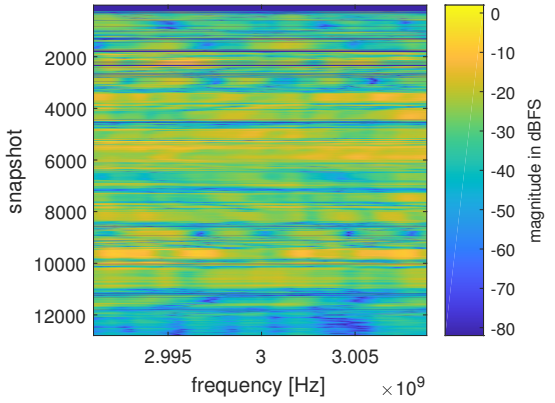
(a) TF of antenna 1, layer 0



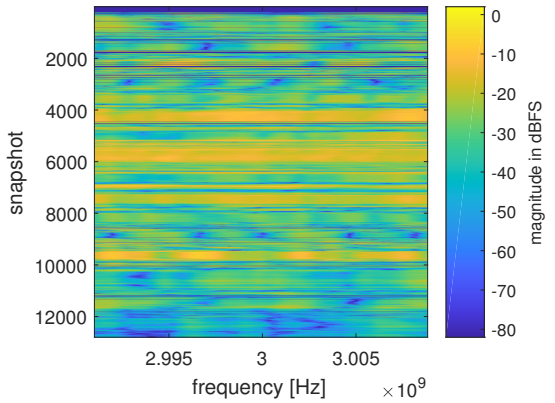
(b) TF of antenna 2, layer 0



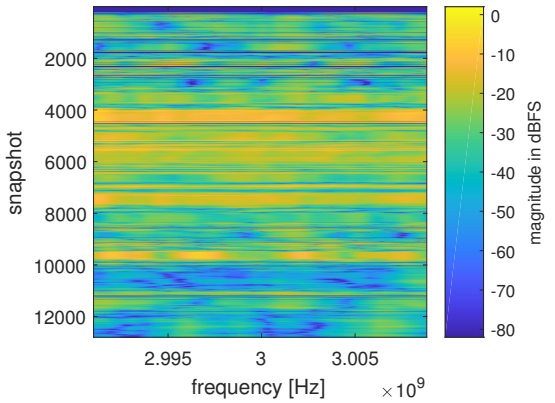
(c) TF of antenna 3, layer 0



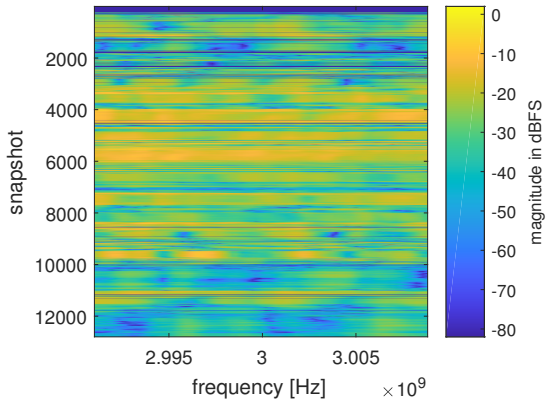
(d) TF of antenna 4, layer 0



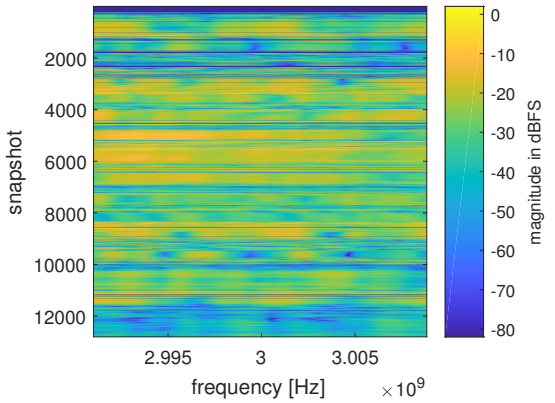
(e) TF of antenna 5, layer 0



(f) TF of antenna 6, layer 0

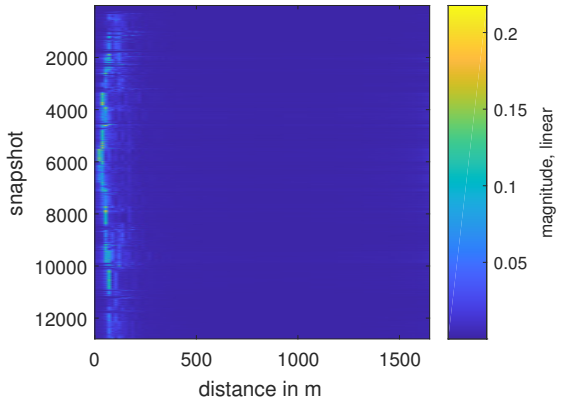


(g) TF of antenna 7, layer 0

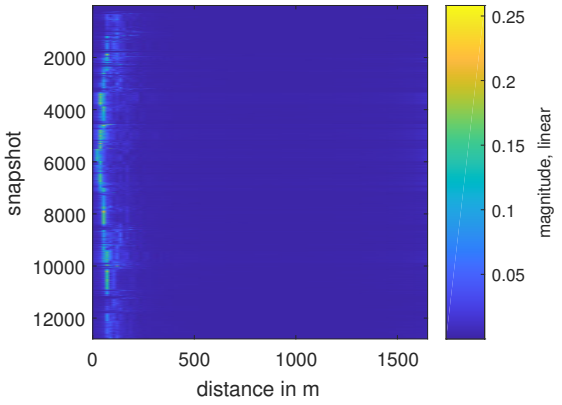


(h) TF of antenna 8, layer 0

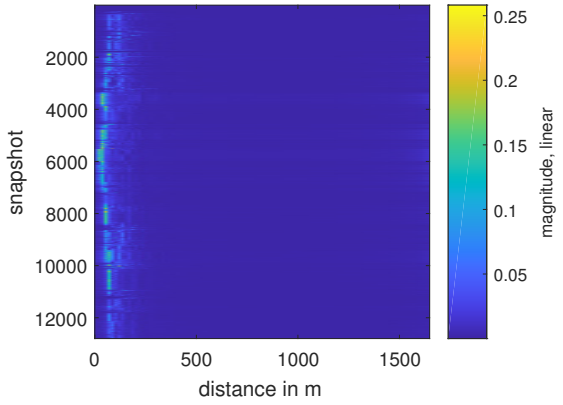
Figure 4.18: TFs of all 8 antennas and layer 0 for measurement 2



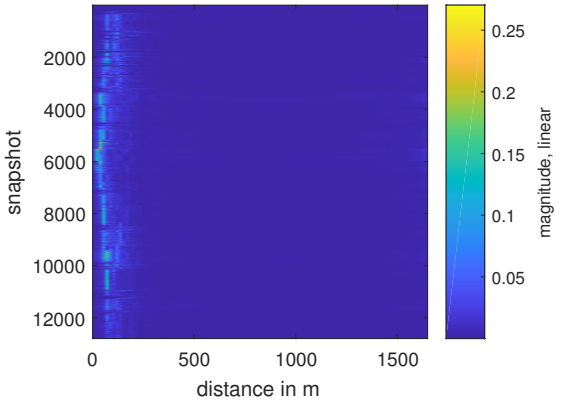
(a) CIR of antenna 1, layer 0



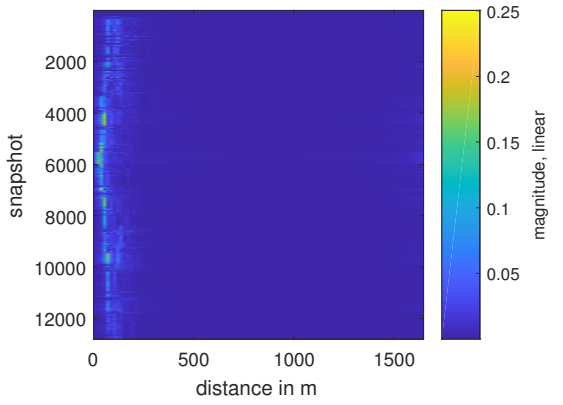
(b) CIR of antenna 2, layer 0



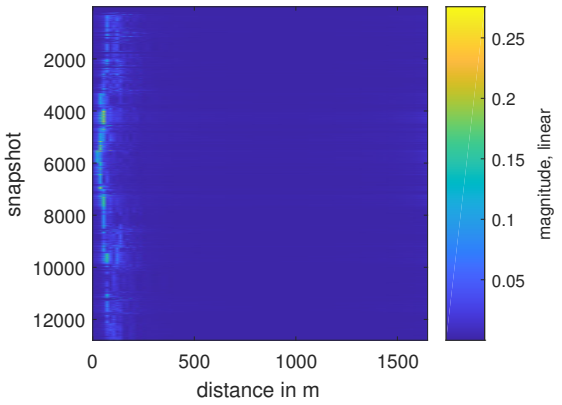
(c) CIR of antenna 3, layer 0



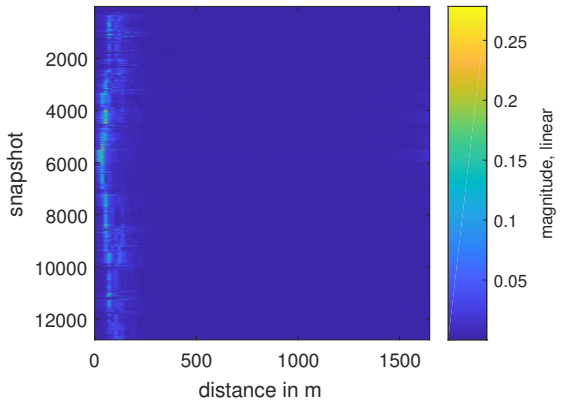
(d) CIR of antenna 4, layer 0



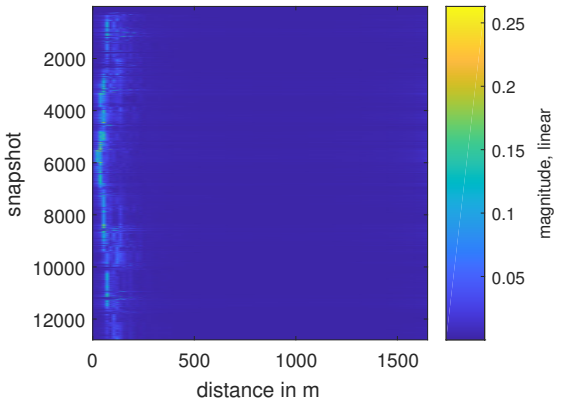
(e) CIR of antenna 5, layer 0



(f) CIR of antenna 6, layer 0



(g) CIR of antenna 7, layer 0



(h) CIR of antenna 8, layer 0

Figure 4.19: CIRs of all 8 antennas and layer 0 for measurement 2

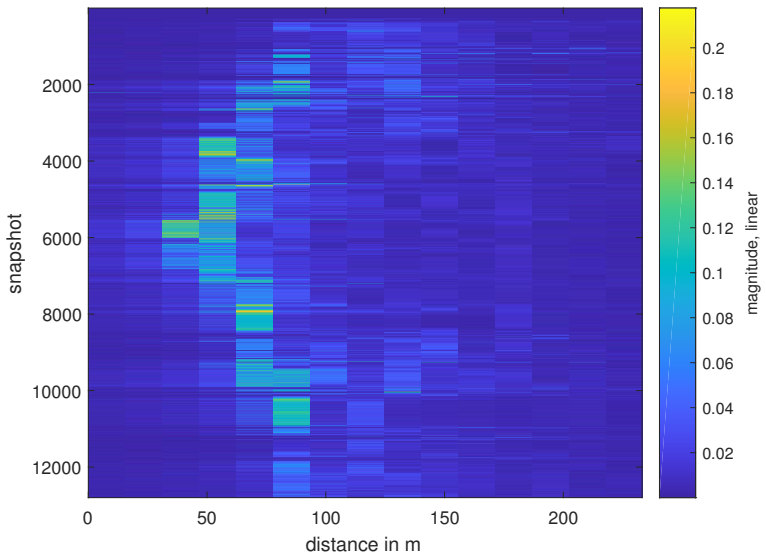


Figure 4.20: CIR of antenna 1, layer 0, measurement 2, detail view

As done for measurement 1, Figure 4.20 shows another smaller excerpt with less distance for the CIR of antenna 1. Here, MPCs can be seen really well, especially one component which shows up as a distorted mirror image with less magnitude of the main component. The mirroring happens at about 100m here. Taking in mind that these distances are not calibrated yet, and thus have a bias in them, the mirroring distance corresponds rather well to the distance from the back-wall of the corridor to the BS.

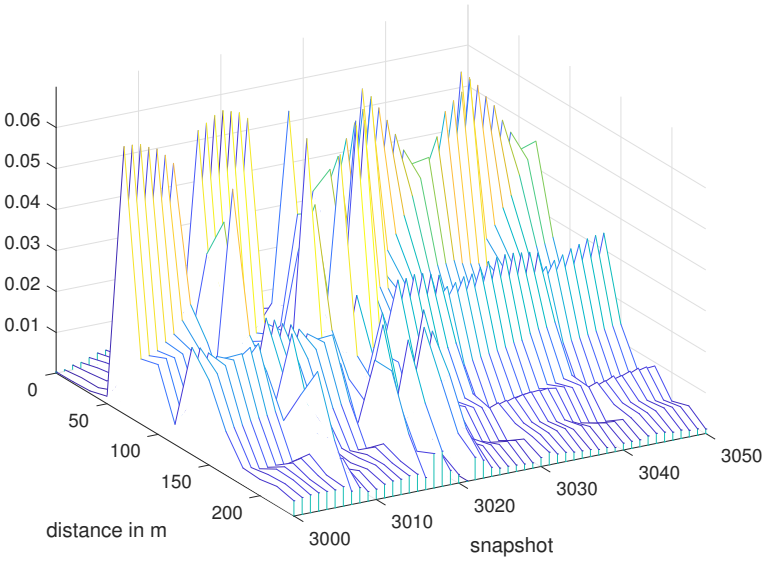


Figure 4.21: CIR of antenna 1, layer 0, measurement 2, detail view, waterfall plot

As seen for measurement 1, Figure 4.21 shows the same problem again: Varying peaks in the CIRs, caused by bad synchronization between BS and MS.

4.4.3 Amplitude of the TFs

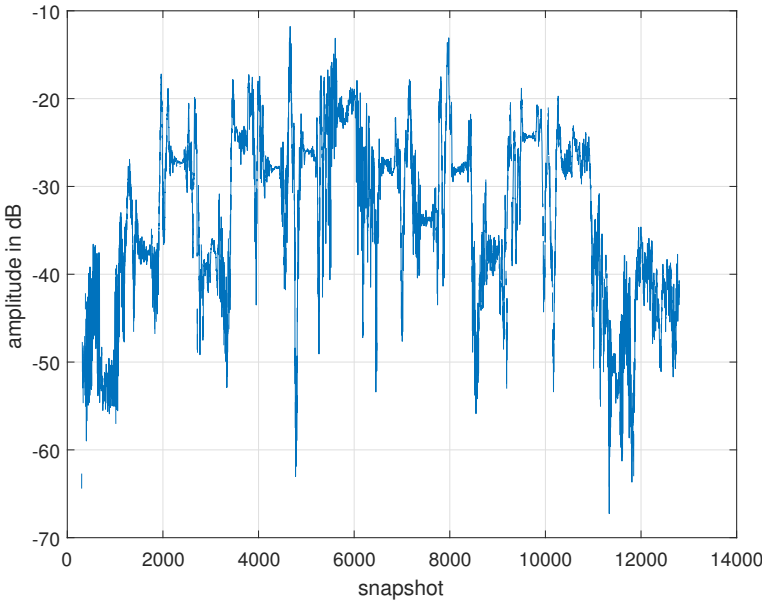


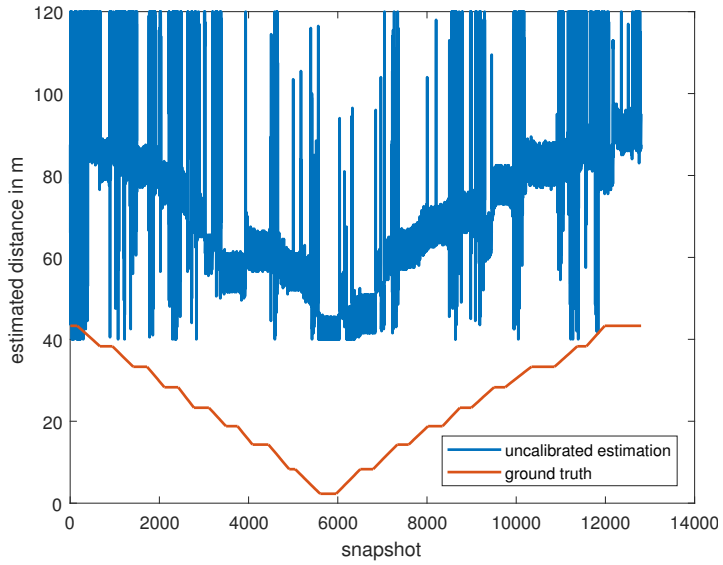
Figure 4.22: Mean amplitude from TFs of antenna 1, layer 0, measurement 2

Figure 4.22 shows the mean amplitude of the TFs, similar to measurement 1 not used in this work.

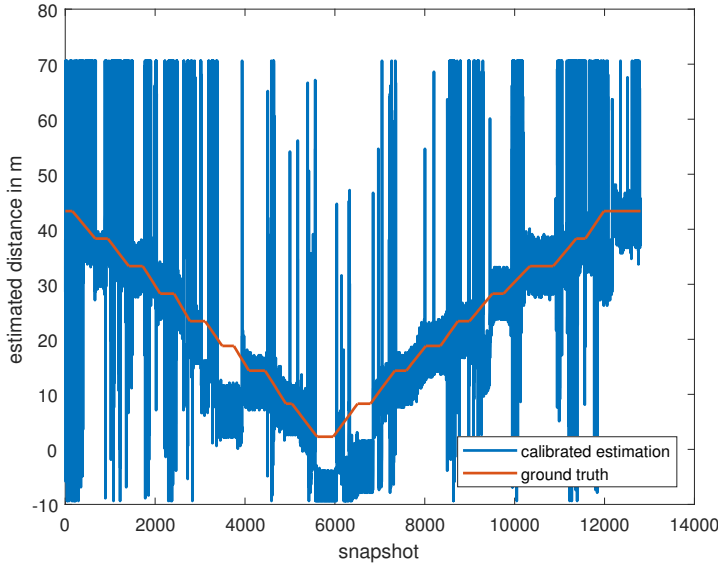
4.4.4 Estimates

In Figure 4.23(a) one can see again that distance calibration is a necessity for this measurement. More interesting, as seen in Figure 4.23(b), the distance estimation seemed to not converge well when the MS was in movement. This could be explained in part by the Doppler-shift lowering SNR because it leads to inter carrier interference [6, Ch. 19.7]. Again, problems with synchronization between BS and MS led to similar effects as in measurement 1, with jumps between samples of the CIR leading to jumps in the distance estimation, and noise within one sample of the CIR. In Figure 4.23(c), the distance estimation is again smoothed by calculating the mean over 101 snapshots. This led to a relatively good estimate for distance, with most outliers being either within periods of movement or from the jumps explained earlier. The error getting bigger over time can not be seen here, at least partially confirming the theory of temperature rise leading to an error in reciprocity calibration within measurement 1.

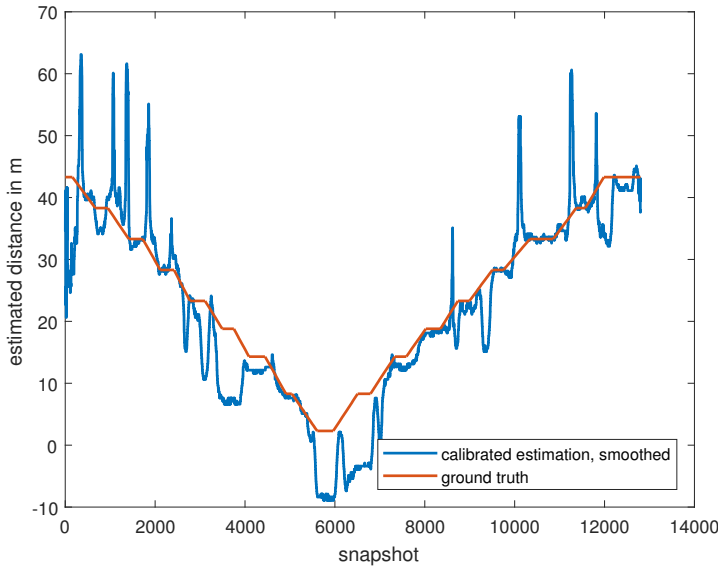
The estimation done via grid search led to the following estimates:



(a) Distance estimation of measurement 2, uncalibrated

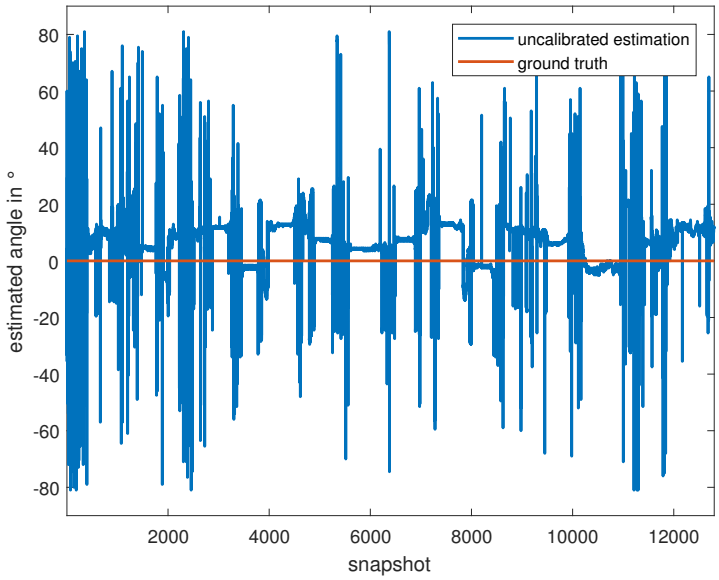


(b) Distance estimation of measurement 2, calibrated

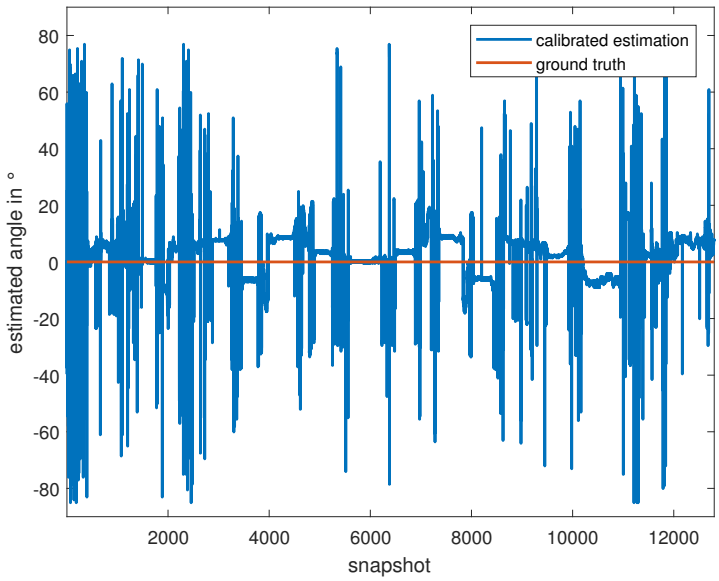


(c) Distance estimation of measurement 2, calibrated and smoothed

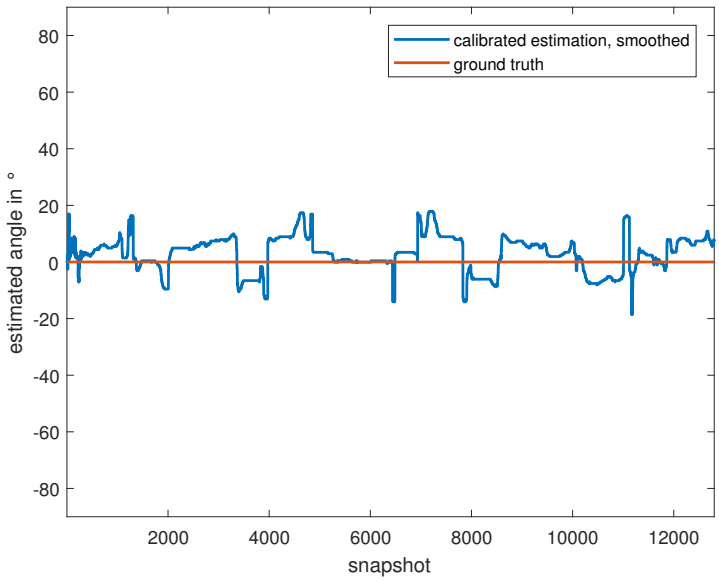
Figure 4.23: Distance estimates, uncalibrated, calibrated and smoothed over ground truth



(a) Angle estimation of measurement 2, uncalibrated



(b) Angle estimation of measurement 2, calibrated



(c) Angle estimation of measurement 2, calibrated and smoothed

Figure 4.24: Angle estimates, uncalibrated, calibrated and smoothed over ground truth

Similar to measurement 1, Figure 4.24(a) and Figure 4.24(b) do not differ significantly. What can be seen in both of them, is that the angle estimation seemed to (as seen also for distance estimation) have problems when the MS was in movement, for the same reason. In Figure 4.24(c), with smoothing applied, the angle estimation does seem to differ from the ground truth for a few positions. Leaving out the errors when the MS was in movement, these angle estimates could be a result of a reflection from the wall parallel to the path being the dominant component of the wave. As the measurements were done in a path to the BS and back, snapshots from about 6000 onwards are from the way back. The path back seems to be almost mirrored from the path forward, and thus leads to the conclusion that these errors depend on the geometry of the room, supporting the idea that the dominant path is a reflection.

4.5 Measurement 3 – Indoor with Synchronization

4.5.1 Environment and Conditions

This measurement was done indoors, in the same corridor as measurement 2, with the intention that this would lead to comparable results, and thus a means of getting to know which errors are caused by erroneous synchronization. The path itself although had to be shortened, as the cables used for synchronization would not allow for the long distances seen in measurement 2.

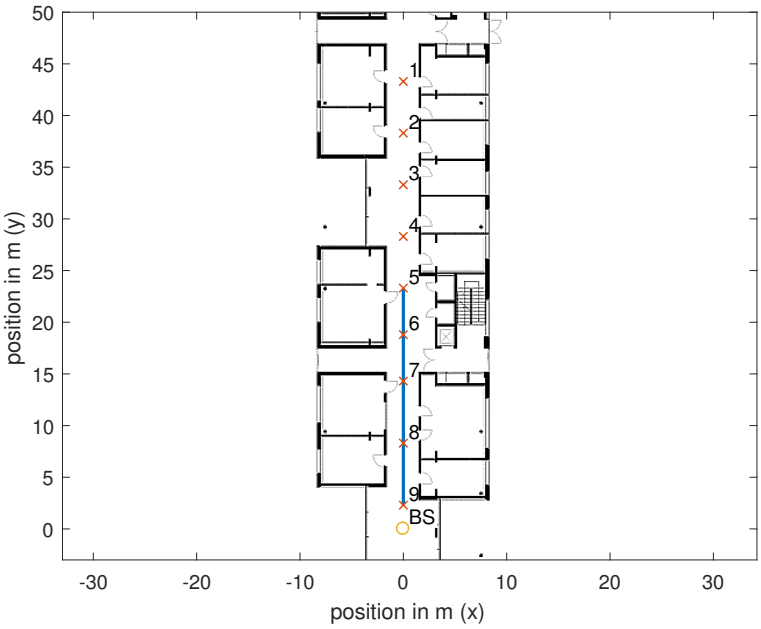
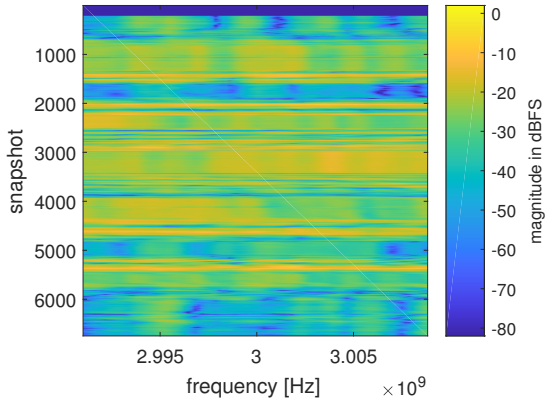
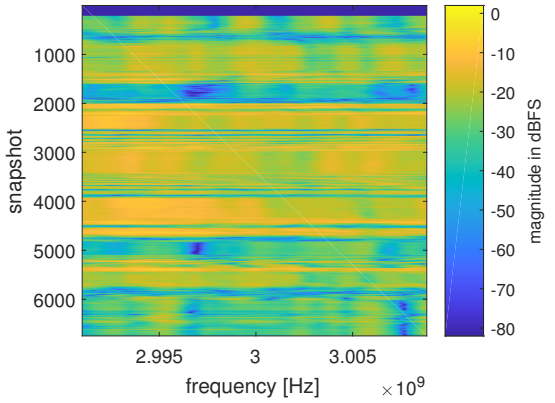


Figure 4.25: True path of measurement 3

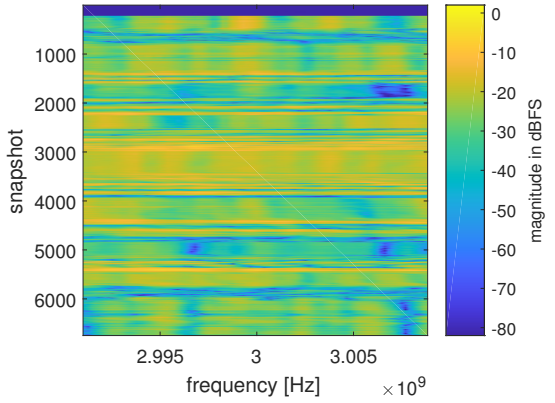
Reiterating, Figure 4.25 shows the true path of measurement 3, for consistency with the same marks as measurement 2. The path taken was almost identical with measurement 2, the difference being that the starting point was at position 5, going to position 9 and back to 5, with intermediate stops at positions 6, 7 and 8.



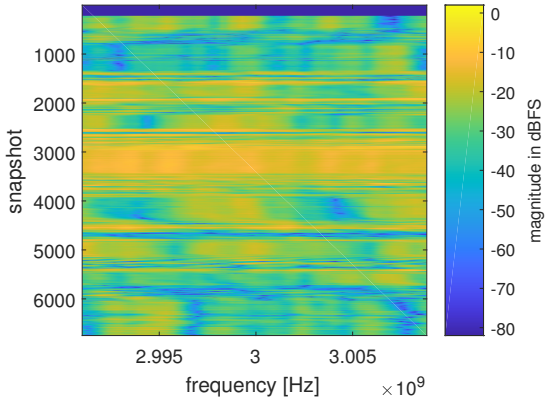
(a) TF of antenna 1, layer 0



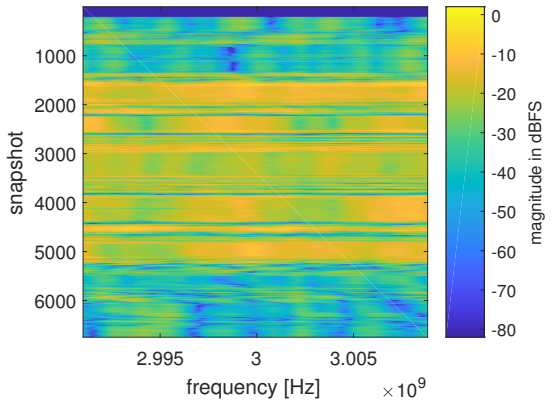
(b) TF of antenna 2, layer 0



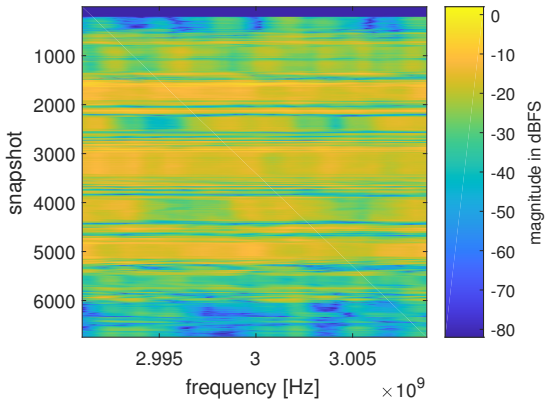
(c) TF of antenna 3, layer 0



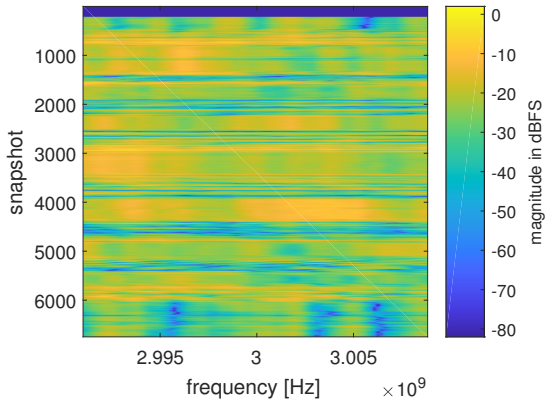
(d) TF of antenna 4, layer 0



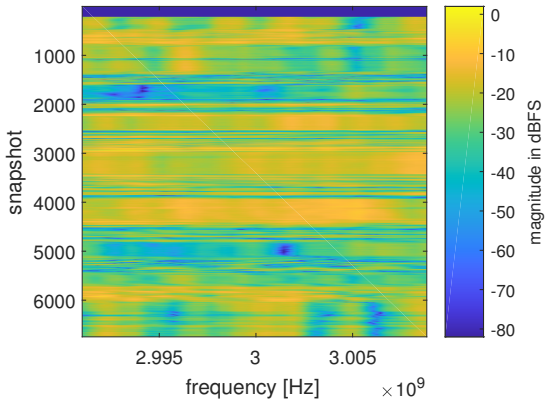
(e) TF of antenna 5, layer 0



(f) TF of antenna 6, layer 0

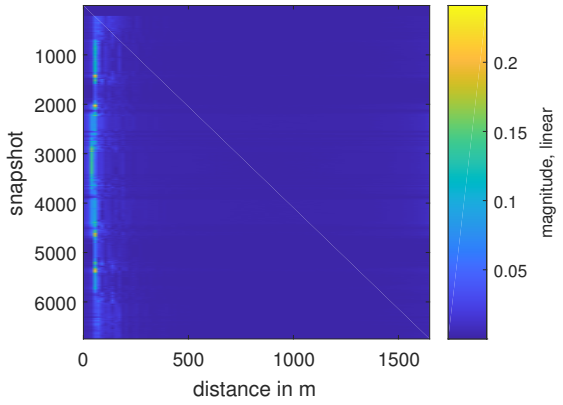


(g) TF of antenna 7, layer 0

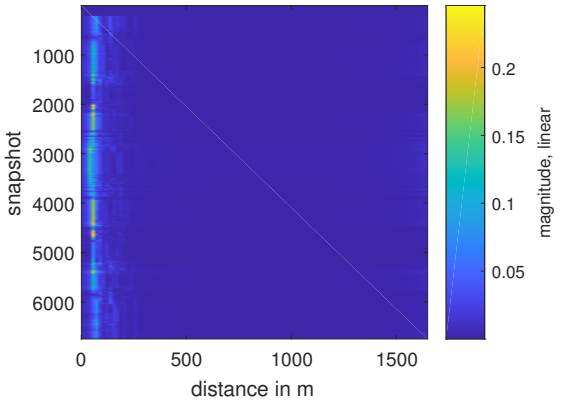


(h) TF of antenna 8, layer 0

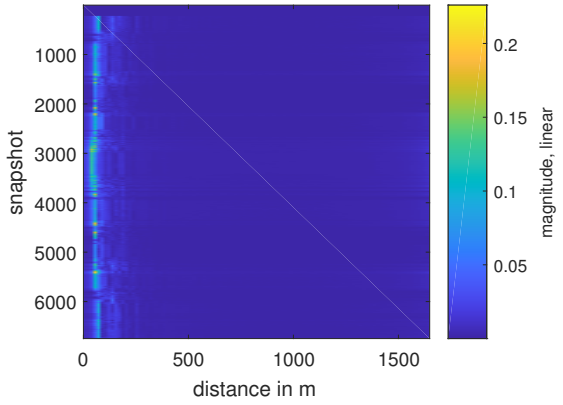
Figure 4.26: TFs of all 8 antennas and layer 0 for measurement 3



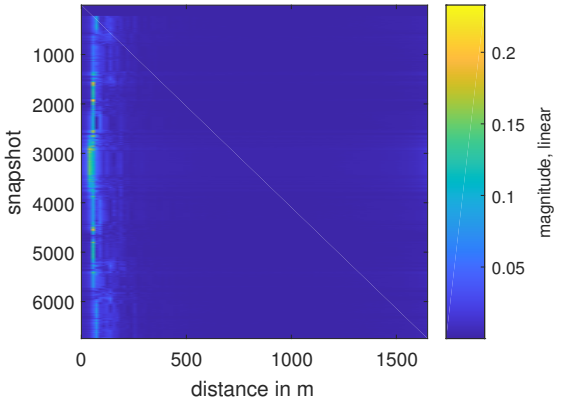
(a) CIR of antenna 1, layer 0



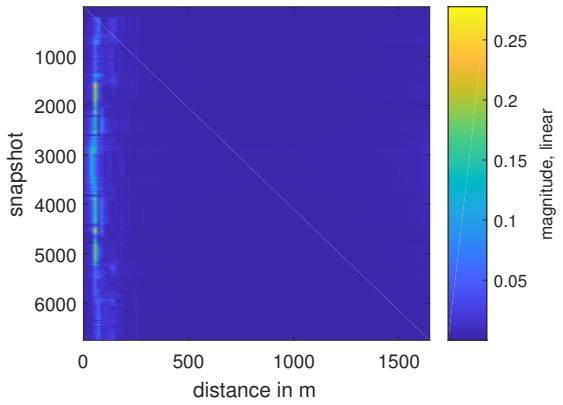
(b) CIR of antenna 2, layer 0



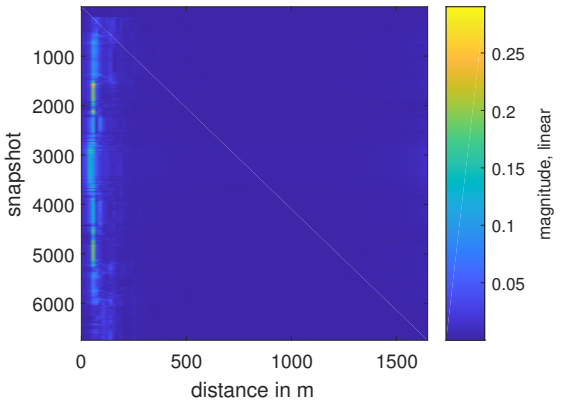
(c) CIR of antenna 3, layer 0



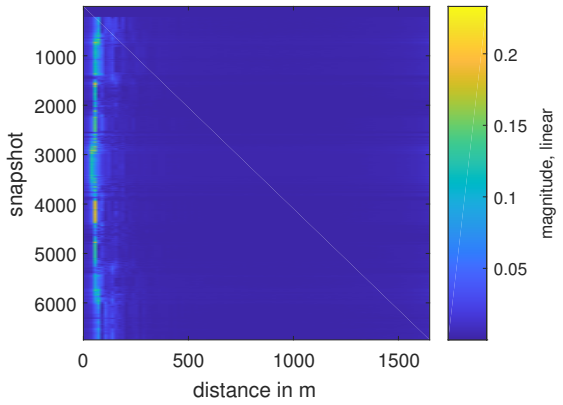
(d) CIR of antenna 4, layer 0



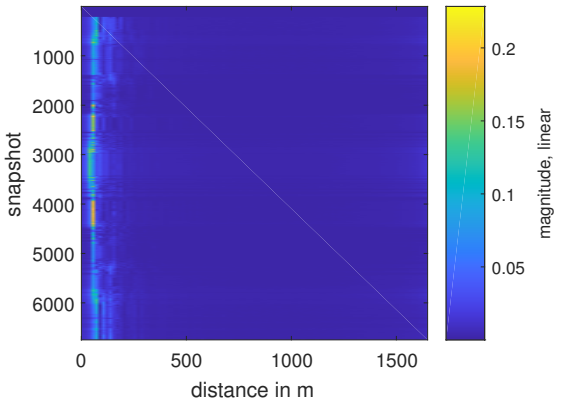
(e) CIR of antenna 5, layer 0



(f) CIR of antenna 6, layer 0



(g) CIR of antenna 7, layer 0



(h) CIR of antenna 8, layer 0

Figure 4.27: CIRs of all 8 antennas and layer 0 for measurement 1

Figures 4.26 show the TFs of measurment 3, for all 8 antennas and one layer/MS. The first approximately 300 snapshots showed no estimation of the TF, as the MS was not powered on at these times. This is due to cable-synchronization requiring that the BS is powered on first, resulting from the software from NI. Here again, like in measurement 2, channel fading can be seen clearly for several times, and the TF was less well defined when the MS was in movement. Also, compared to measurement 2 there are only about half as many snapshots, due to the measurement 2 being longer in time because a longer path was chosen. Figures 4.27 show the CIRs of measurement 3, corresponding to the TFs. At first glance, they seem to be comparable to those resulting from measurement 2, but they are slightly more confined in distance than Figures 4.19.

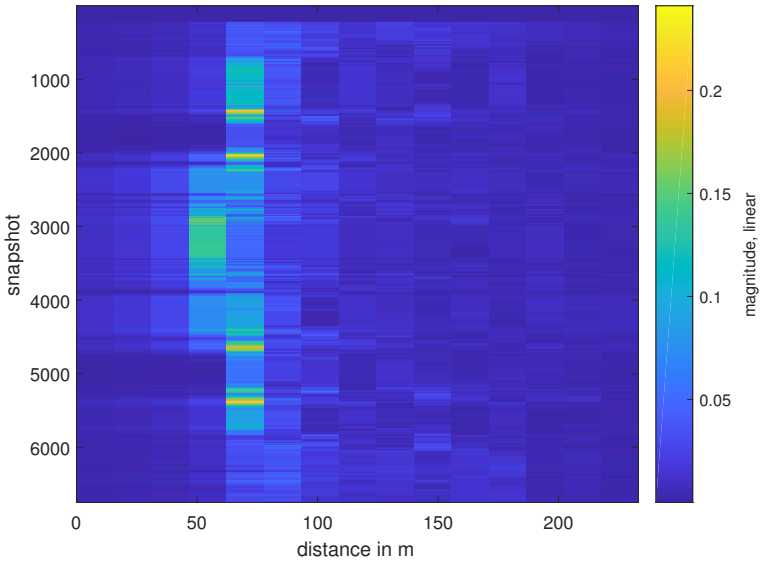


Figure 4.28: CIR of antenna 1, layer 0, measurement 3, detail view

Figure 4.28 shows an excerpt of Figure 4.27, comparable to those from measurement 1 and 2. Again, some mirroring can be seen, as in measurement 2, but the main peaks of measurement 3 are more defined.

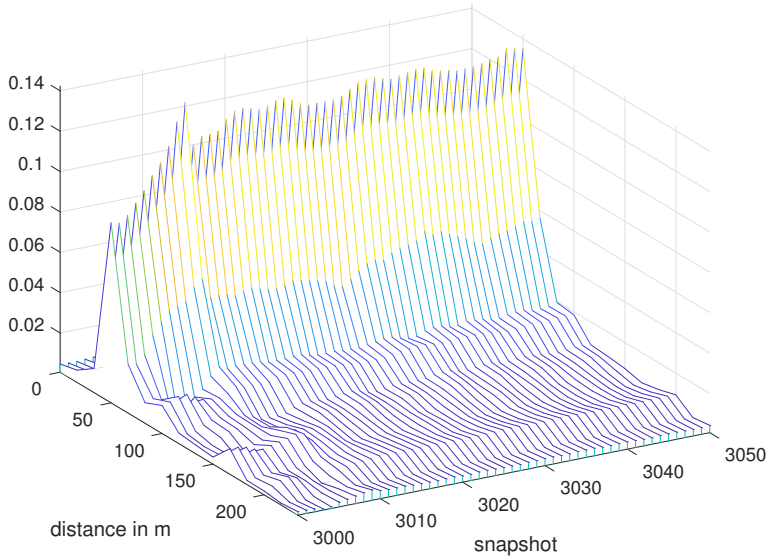


Figure 4.29: CIR of antenna 1, layer 0, measurement 3, detail view, waterfall plot

Compared to Figure 4.21, Figure 4.29 shows much less variation in the peaks of the CIRs, stemming from a much better synchronization. In further results, one can see that these smaller variations lead to much more reliable results.

4.5.2 Amplitude of the TFs

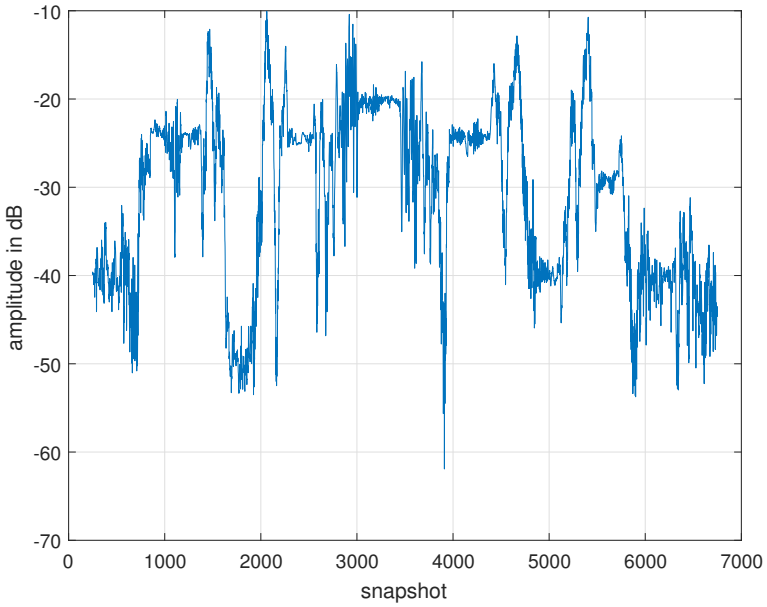
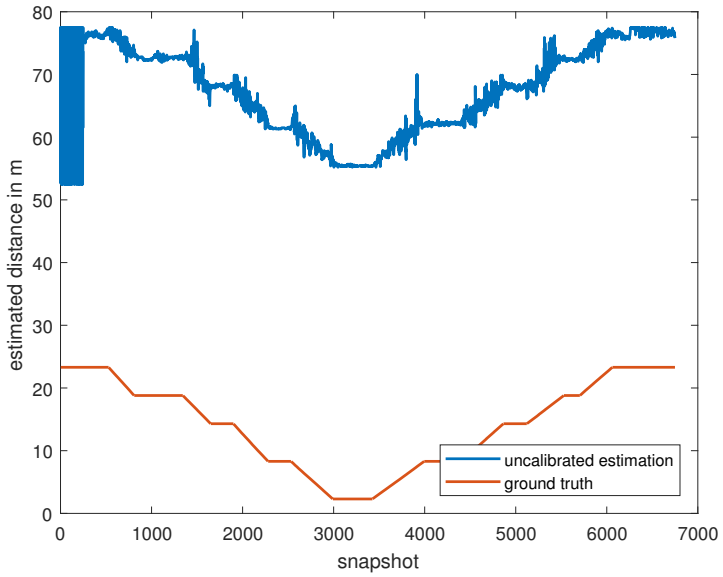


Figure 4.30: Mean amplitude from TFs of antenna 1, layer 0, measurement 3

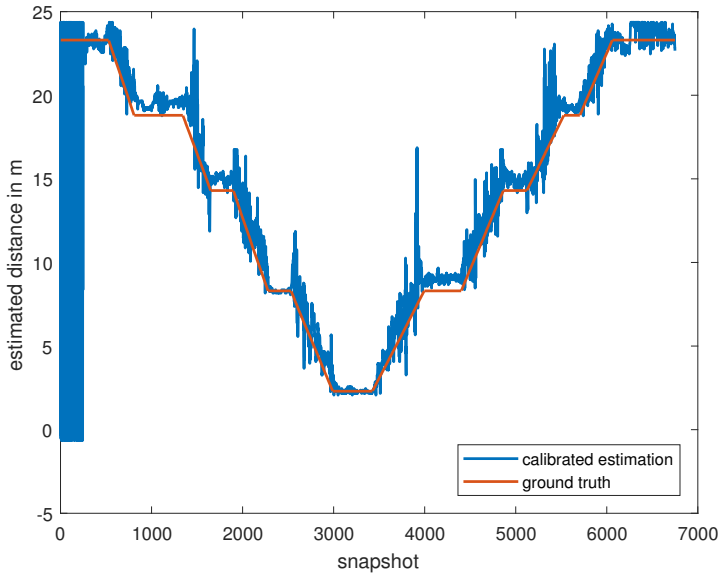
Figure 4.30 shows the mean amplitude of the TFs, similar to the two previous measurements.

4.5.3 Estimates

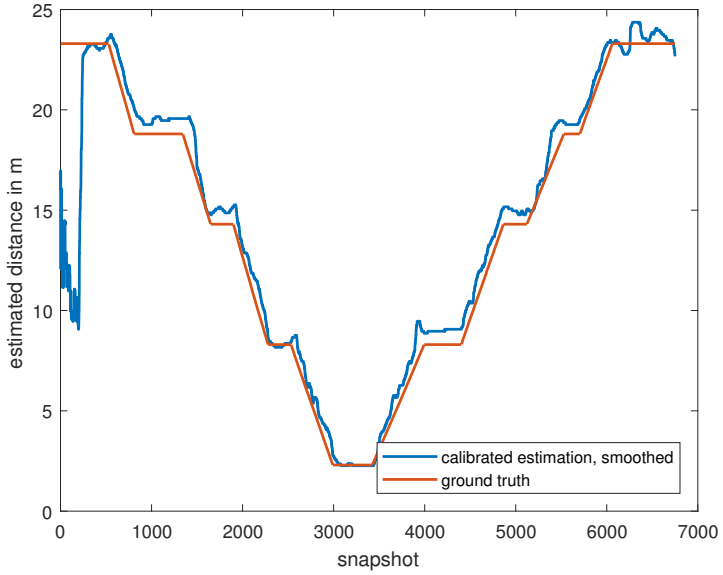
The estimation done via grid search led to the following estimates:



(a) Distance estimation of measurement 3, uncalibrated

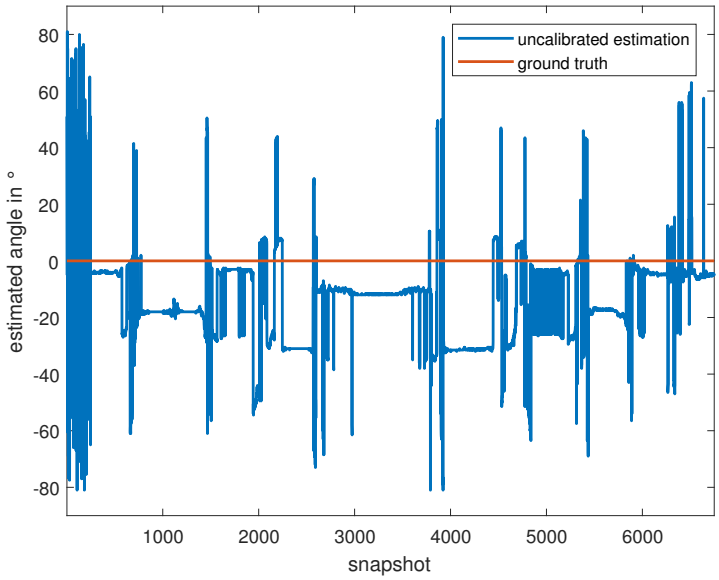


(b) Distance estimation of measurement 3, calibrated

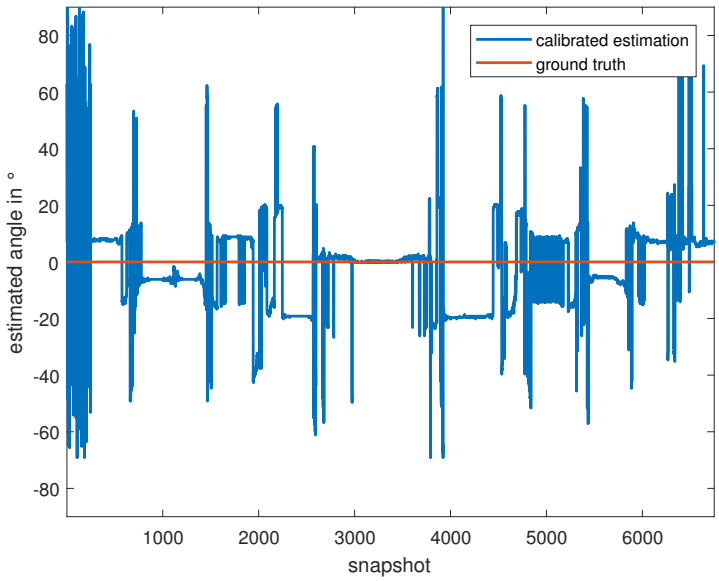


(c) Distance estimation of measurement 3, calibrated and smoothed

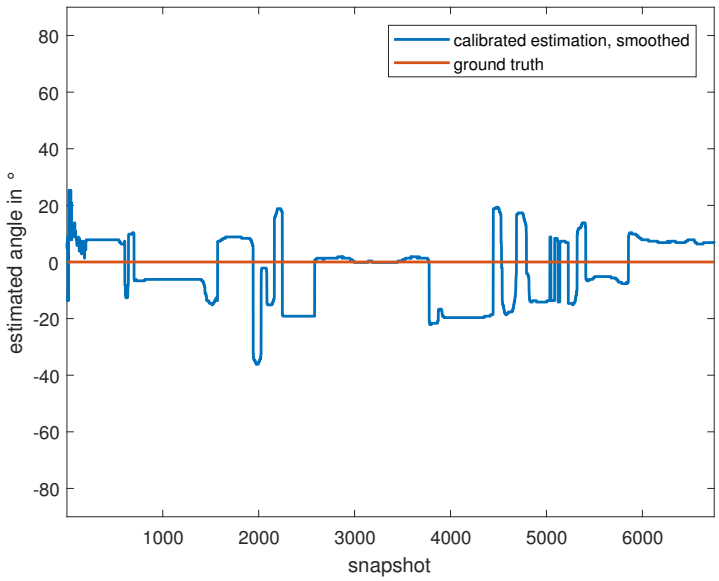
Figure 4.31: Distance estimates, uncalibrated, calibrated and smoothed over ground truth



(a) Angle estimation of measurement 3, uncalibrated



(b) Angle estimation of measurement 3, calibrated



(c) Angle estimation of measurement 3, calibrated and smoothed

Figure 4.32: Angle estimates, uncalibrated, calibrated and smoothed over ground truth

As one can clearly see in Figure 4.31(a), calibration is still a necessity with cable-synchronization. Also, the first few snapshots did not yield any useful result for estimation, as the BS was not yet powered on. More interestingly, in Figure 4.31(b) one can clearly see that the estimation is much more reliable and less noisy with cable-synchronization. The random jumps between two samples of the CIR, as in the previous measurements, are no longer present. Also, the estimation itself is less noisy. With Figure 4.31(c), with smoothing applied in the same manner as in measurements 1 and 2, the results seem to be very accurate. Most estimates seem to be within 1m of the ground truth, and especially for nearer measurements, the estimation is very accurate. Also, here the measurement was working well mostly when the MS was in movement, as the additional noise from the Doppler-shift did still allow for a good enough SNR.

Figure 4.32(a) shows the uncalibrated angle estimation for measurement 3. As can be seen, the estimation of the angle did not profit from better synchronization in the same way as the distance estimation, but did still improve a little bit compared to measurement 2, mostly when the MS was in movement. With calibration applied in Figure 4.32(b), the angle estimation shows an accuracy comparable to measurement 2, with errors within about 20° of the ground truth. Figure 4.32(c) is looking rather similar to Figure 4.24(c) from measurement 2. It also shows almost mirrored estimations around the position nearest to the BS. The explanation for the estimation not showing the ground truth is the same as in measurement 2: It is possible that reflections were overshadowing the LoS components.

5 Statistical Evaluation of Estimates

5.1 Introduction

In this chapter, the results are evaluated by calculating the Cramer-Rao lower bound of the estimates, a statistical lower bound of estimator performance. These results are then discussed and compared for the three measurements.

5.2 Cramer-Rao lower bound

Calculation of the CRLB was done via adapting equations from [12, p.528] and [13, (33)]. This led to following equations, where J_ϕ and J_τ are the position information of the AoA and ToA respectively:

$$D_\lambda^2(\phi) = \frac{\left(N^2-1\right)}{24}\frac{1+\cos^2(2\phi+\pi)}{2} \tag{5.1}$$

$$J_\phi \approx 8\pi^2 D_\lambda^2\left(\phi\right) N \cdot \mathrm{SNR} \tag{5.2}$$

$$J_\tau = 8\pi^2\beta^2 N \cdot \mathrm{SNR} \tag{5.3}$$

$D_\lambda^2(\phi)$ is the quadratic aperture for a $\lambda/2$ ULA, depending on the AoA. β^2 is the mean squared bandwidth, defined as $\beta^2 = \|\dot{\mathbf{s}}\|^2 / \left(4\pi^2\|\mathbf{s}\|^2\right) = \int_f f^2|S(f)|^2 df$ for a normalized pulse $\|\mathbf{s}\|^2 T_s = 1$. Here, $\dot{\mathbf{s}}$ is the sampled derivative of of the pulse [13]. N is the number of antennas, and SNR is the signal-to-noise ratio (SNR) described in the next section. This leads to the following angulation error bound (AEB) and ranging error bound (REB):

$$\mathrm{var}(\hat{\phi}) \geq \mathrm{AEB} = \sqrt{J_\phi^{-1}} \tag{5.4}$$

$$\mathrm{var}(\hat{\tau}) \geq \mathrm{REB} = \sqrt{J_\tau^{-1}} \tag{5.5}$$

The CRLB is the statistical lower bound on the variance for any unbiased estimator [6, p.480]. The AEB is the CRLB for the angle estimation, and the REB is the CRLB for the range/distance estimation. These results follow from the Fisher information matrix

$$\mathbf{J}_{\tau,\phi} = \mathbb{E}_{\mathbf{r};\tau,\phi}\left[\left(\frac{\partial}{\partial(\tau,\phi)}\ln((\tilde{p}(\boldsymbol{r};\tau,\phi))\right)\left(\frac{\partial}{\partial(\tau,\phi)}\ln(\tilde{p}(\boldsymbol{r};\tau,\phi))\right)^T\right] \tag{5.6}$$

which is comparable to the Fisher information matrix in [13, (7)].

5.3 Estimation of SNR

For the calculation of the CRLB seen in the previous section, an estimate for the SNR had to be found. The complex amplitude estimated in Chapter 3, with (3.13) was used for this purpose. The signal amplitude can be assumed to be approximately the mean of $\alpha_n[i]$ over several snapshots i and the noise can be assumed to be approximately the variance of $\alpha_n[i]$. Thus, the following equations can be used to obtain the SNR, with $L+1$ being the number of snapshots used for estimation:

$$\mu_{\alpha,n}[i] = \frac{1}{L+1}\sum_{a=i}^{i+L} |\alpha_n[a]| \tag{5.7}$$

$$\sigma^2_{\alpha,n}[i] = \frac{\sum_{a=i}^{i+L} |\alpha_n[a]|^2 - (\sum_{a=i}^{i+L} |\alpha_n[a]|)^2/(L+1)}{L} \tag{5.8}$$

$$\mathrm{SNR}_n[i] \approx \frac{\mu_{\alpha,n}[i]^2}{\sigma^2_{\alpha,n}[i]} \tag{5.9}$$

Equation (5.7) is for the calculation of the mean of $\alpha_n[i]$ for snapshots from i to $i+L$, (5.8) calculates the Variance of $\alpha_n[i]$ for given samples, with the normalization factor L being due to Bessel's correction for an unbiased estimate. Lastly, the SNR is calculated with (5.9). It should be noted, that this is an approximation, which could yield false values if there was interference at the times $\alpha_n[i]$ was estimated. To get an estimate for the SNR of all used antennas, the mean of $\mathrm{SNR}_n[i]$ is used:

$$\mathrm{SNR}[i] = \frac{1}{N}\sum_{n=1}^{N} \mathrm{SNR}_n[i] \tag{5.10}$$

5.4 Evaluation of Measurement 1

5.4.1 Estimated SNR

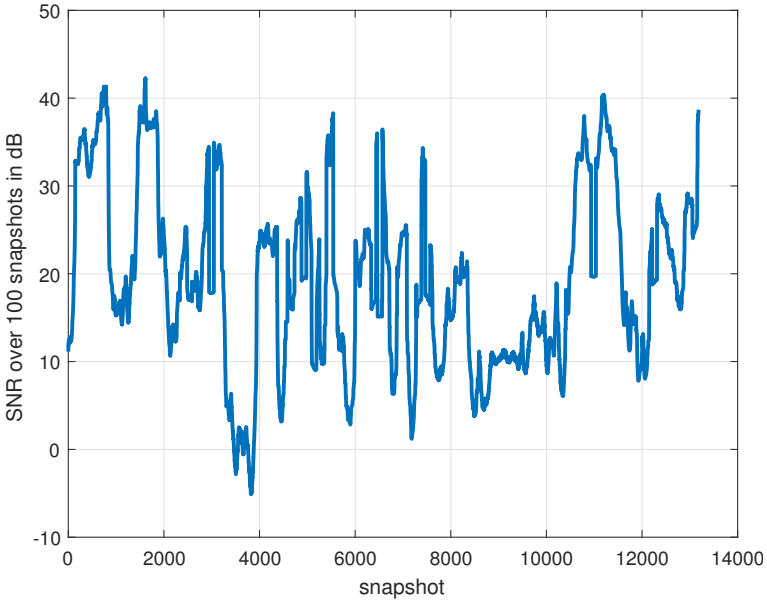


Figure 5.1: SNR of measurement 1 with $L = 100$

Figure 5.1 shows the SNR over measurement 1. As can be seen, the SNR was mostly in the range of $0 - 40$dB. At snapshots of approximately 3500 to 4000 the estimation of parameters was not converging, which leads to an estimated SNR of approximately 0dB.

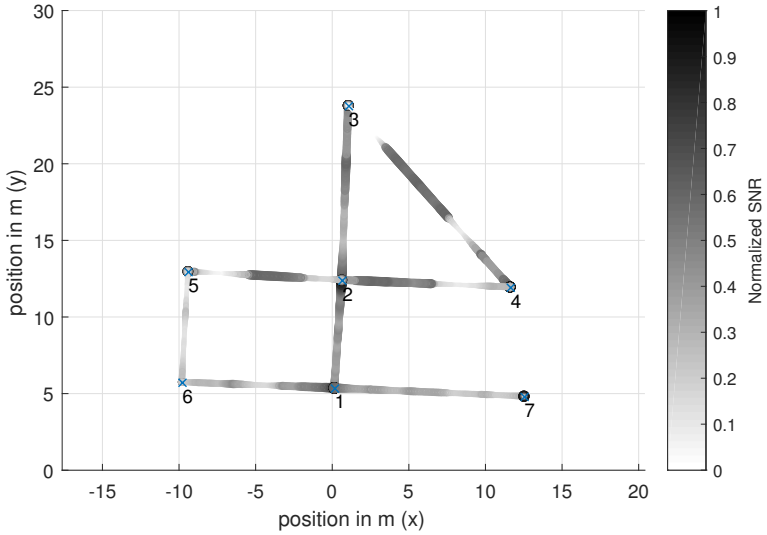


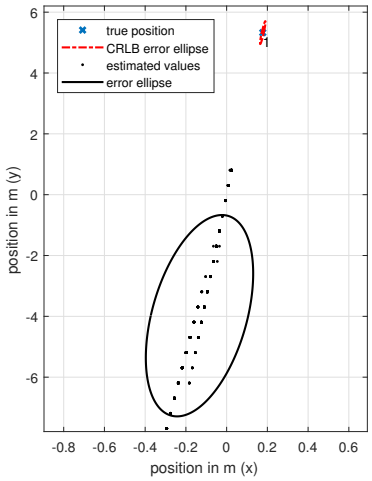
Figure 5.2: SNR of measurement 1 overlayed over the path, bigger marker width and darker color denotes better SNR. The SNR in dB has been normalized to [0,1]; 1 being the maximum SNR.

Figure 5.2 shows the SNR along the actual path, with bigger marker width and darker markers denoting a better SNR. Here it can be seen that some areas along the path seemed to work rather well, but others seemed to show interferences because of channel fading or shadowing between antennas.

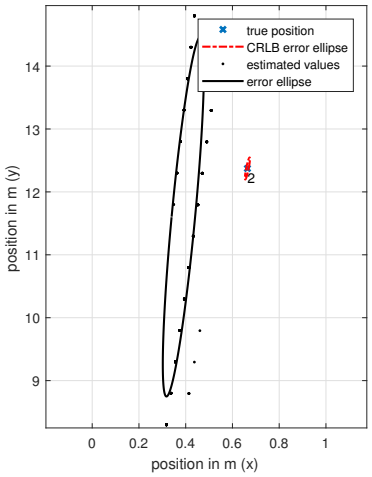
5.4.2 CRLB Ellipses

Figure 5.3(a-i) shows the error ellipses for the estimated values and the corresponding CRLB ellipses, calculated using (5.4) and (5.5). As one can see, the error ellipses for some subfigures (e.g. for position 1 and 7) show a large deviation from the CRLB error ellipses. This is due to the synchronization problems already discussed in Chapter 4. Other positions (i.e. position 9) show a big margin in the angle estimation between the CRLB error ellipse and the error ellipse from the estimated values. This is due to the estimator finding two local minima in the angle, and thus the error ellipse showing the deviation between these two angle estimates. Otherwise, the estimation performance was overall not as good as the CRLB indicates, but mostly these problems in this measurement arose from the erroneous synchronization.

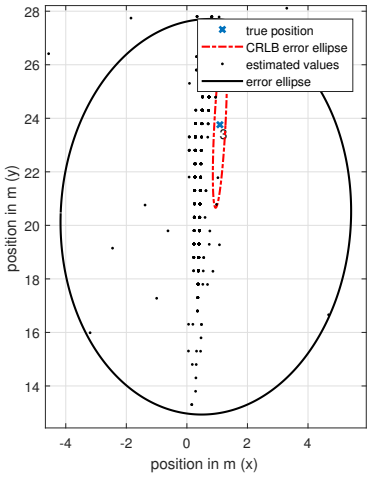
Note that the axis scaling is not by any means equal in these plots, which is to illustrate smaller angle deviations easier. Also, the *positions* here correspond to the sequence in which the points were gone through in the measurement, and the numbers seen in the plots correspond to the numbers seen in Figure 5.2. This is because some positions were measured twice, as a result of crossing paths.



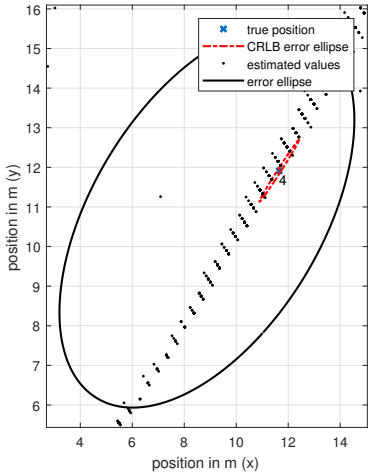
(a) CRLB and estimates of position 1 (moving towards 2)



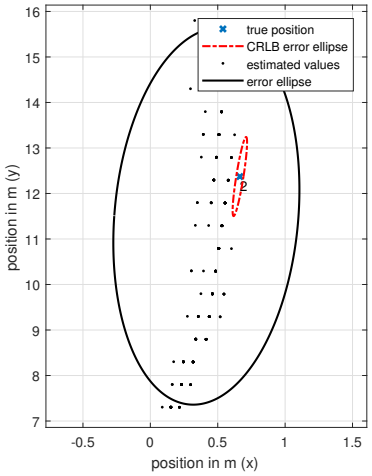
(b) CRLB and estimates of position 2 (moving towards 3)



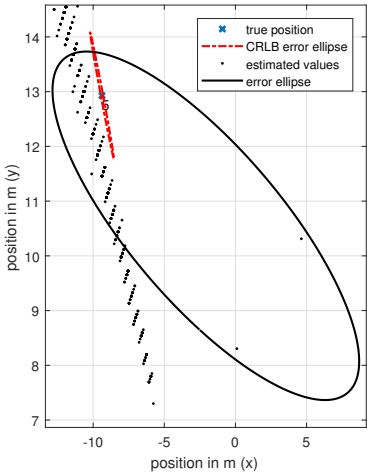
(c) CRLB and estimates of position 3



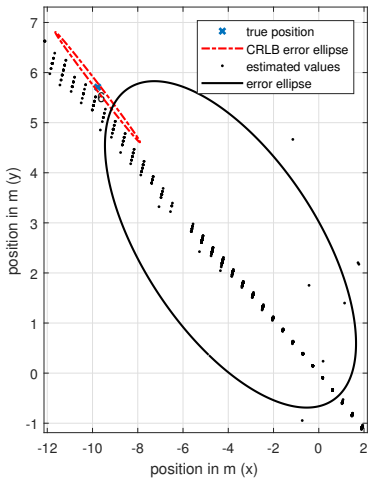
(d) CRLB and estimates of position 4 (moving towards 2)



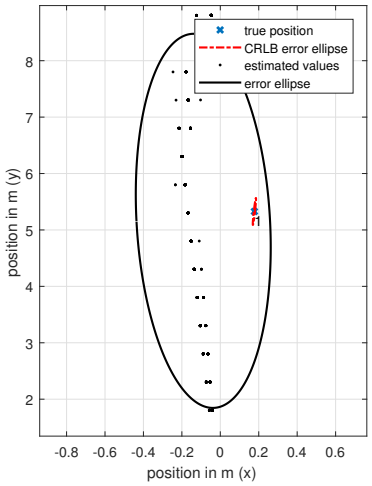
(e) CRLB and estimates of position 2 (moving towards 5)



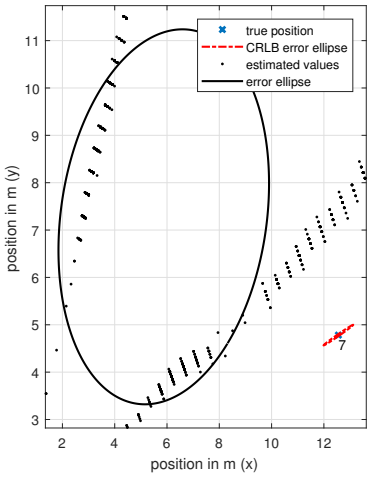
(f) CRLB and estimates of position 5



(g) CRLB and estimates of position 6 (moving towards 1)



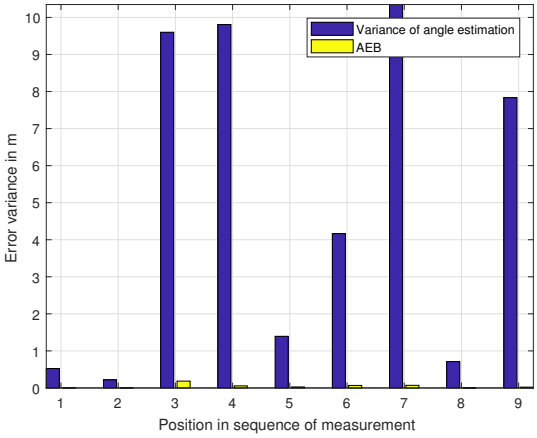
(h) CRLB and estimates of position 1 (moving towards 7)



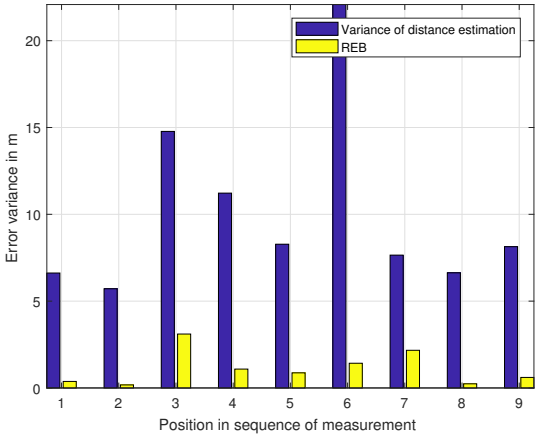
(i) CRLB and estimates of position 7

Figure 5.3: Error ellipses for the CRLB and actual estimates of measurement 1. x and y axes are not eqal.

5.4.3 Errorbars of CRLB and Estimation



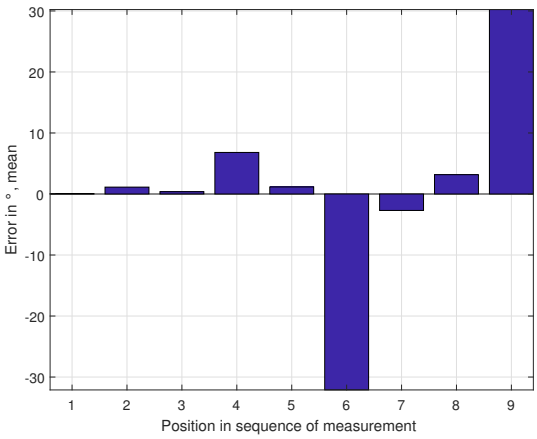
(a) Errorbars of AEB and variance of estimated angles



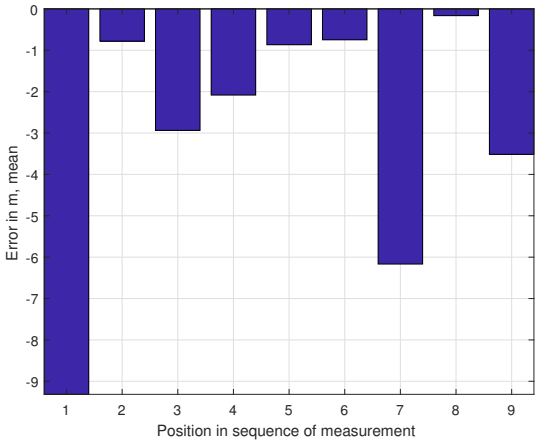
(b) Errorbars of REB and variance of estimated distances

Figure 5.4: Error variance for AEB and REB and distance and angle estimates, measurement 1

Figure 5.4(a) shows the AEB as errorbars from calculating the CRLB and the errorbars for the estimated values for the angle. As one can see, the estimation had its flaws and is not near the theoretical limit, due to the fact that the algorithm seemed to find other local minima. Still, some measurements, where SNR was good and the algorithm converged to the right minimum, show promising results. Figure 5.4(b) shows the REB as errorbars from calculating the CRLB and the errorbars for the estimated values for the distance. These values are also not very reliable, as can be seen at the large error bars. Again, these large errors come mostly from the erroneous synchronization.



(a) Mean error of estimated angles



(b) Mean error of estimated distances

Figure 5.5: Mean for angle and distance error, measurement 1

Figure 5.5(a) shows the mean error of the angle estimation, Figure 5.5(b) shows the mean error of distance estimation. Explanations for the errors here are the same as for the variance errors.

5.5 Evaluation of Measurement 2

5.5.1 Estimated SNR

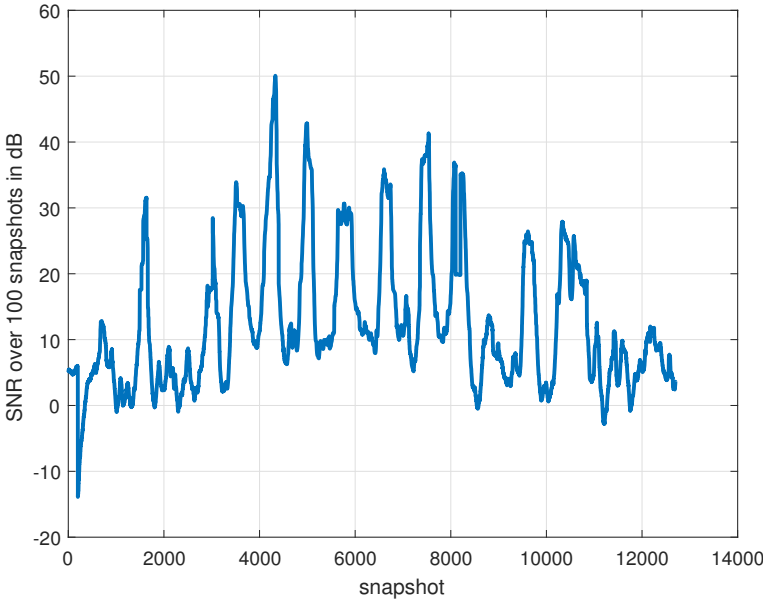


Figure 5.6: SNR of measurement 2 with $L = 100$

As can be seen in Figure 5.6, the SNR for the indoor measurement without cable-synchronization was comparable to the SNR from measurement 1. The biggest difference is the harsher dampening when in movement, and the bigger dependence on distance, which can be attributed simply to the bigger distances when measuring.

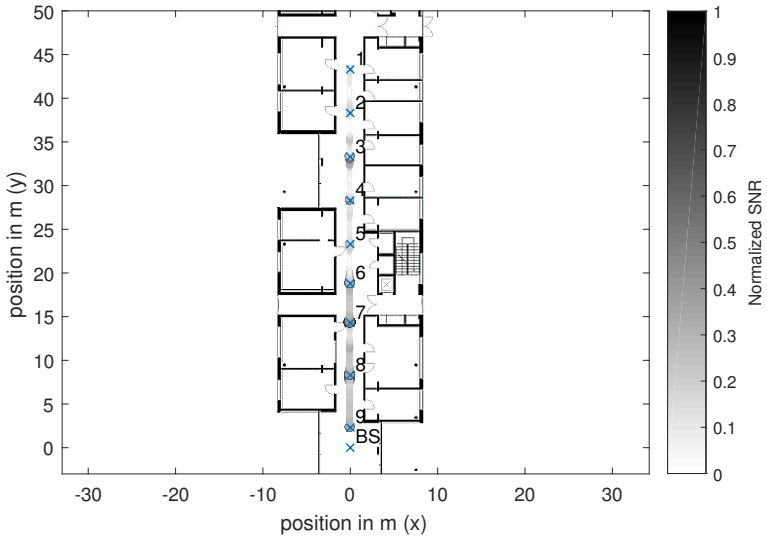
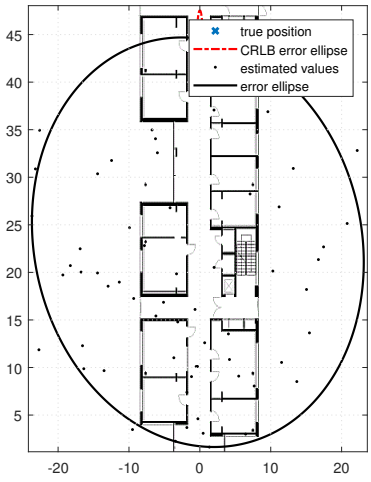


Figure 5.7: SNR of measurement 2 overlayed over the path, bigger marker width and darker color denotes better SNR. The SNR in dB has been normalized to [0,1]; 1 being the maximum SNR.

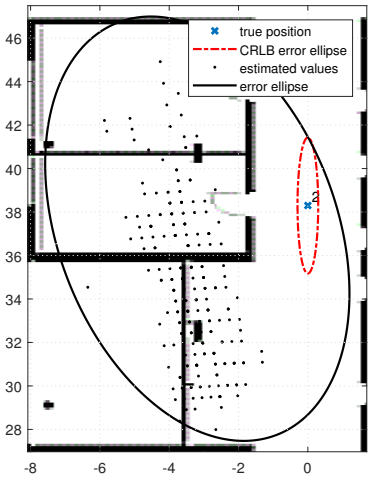
Plotting The SNR over the actual path, one can see the dependence of SNR on distance much better, and that the SNR was better when the MS was not in movement. Here again, darker color denotes a better SNR, lighter color denotes a smaller SNR.

5.5.2 CRLB Ellipses

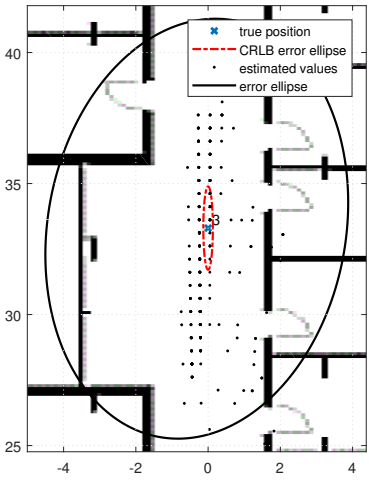
Within Figures 5.7(a-q) similar to measurement 1, one can again see the error ellipses for the estimated values and the corresponding CRLB ellipses. The same problems arose here, as with measurement 1, mostly insufficient synchronization led to bad estimates for the distance. The angle estimation showed another effect: Mostly the estimation seemed to work rather well, converging to a relatively small angular error. But most values do have a bias (e.g. in position 2, 5, 7, 8, 11, 12 ...). This could be due to reflections from the walls being the dominant pulse, not the LoS component. As the hallway is narrow, and reflections from walls are not easily discernible in distance from LoS components, this seems like a plausible explanation.



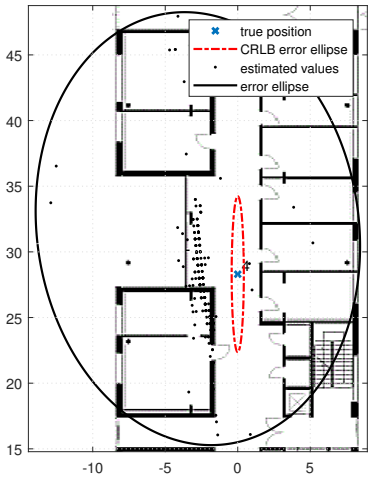
(a) CRLB and estimates of position 1



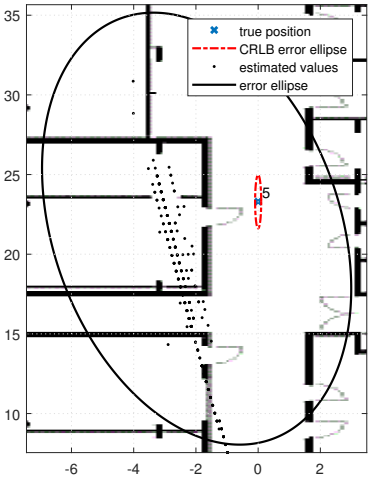
(b) CRLB and estimates of position 2



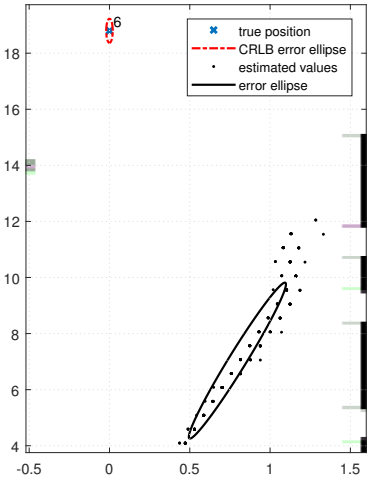
(c) CRLB and estimates of position 3



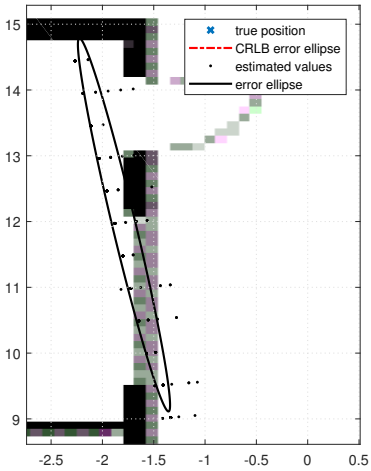
(d) CRLB and estimates of position 4



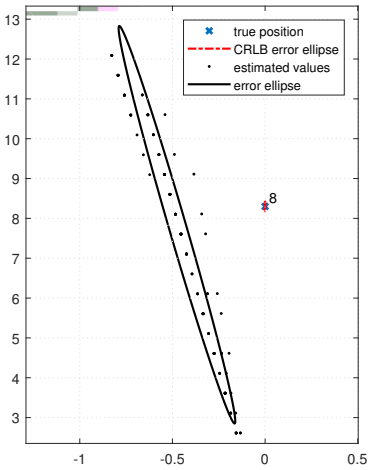
(e) CRLB and estimates of position 5



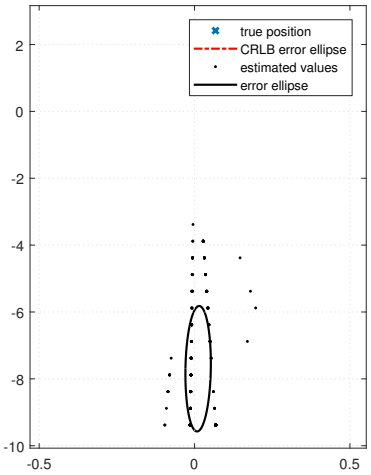
(f) CRLB and estimates of position 6



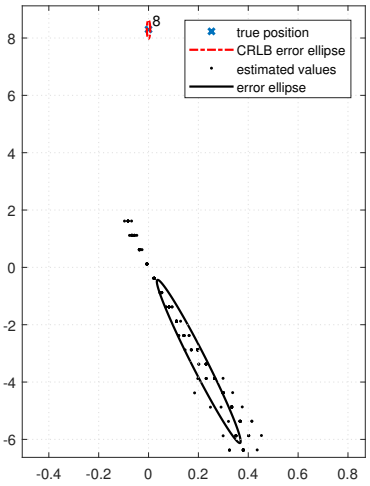
(g) CRLB and estimates of position 7



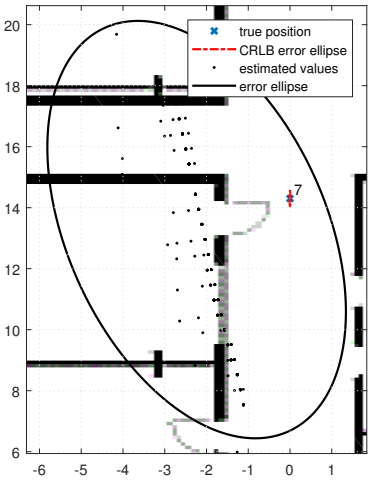
(h) CRLB and estimates of position 8



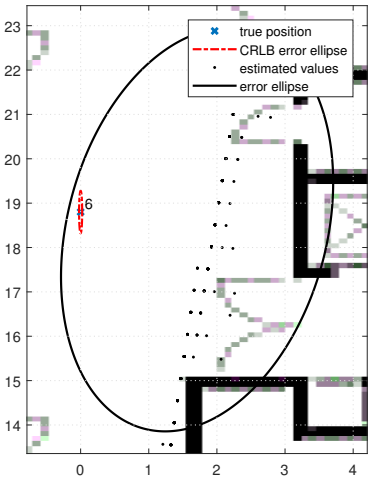
(i) CRLB and estimates of position 9



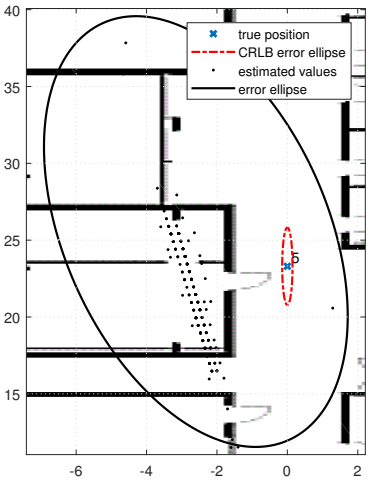
(j) CRLB and estimates of position 10



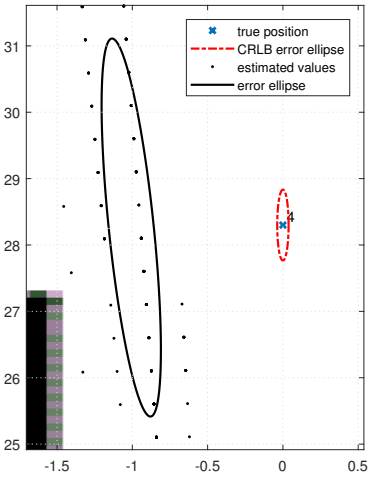
(k) CRLB and estimates of position 11



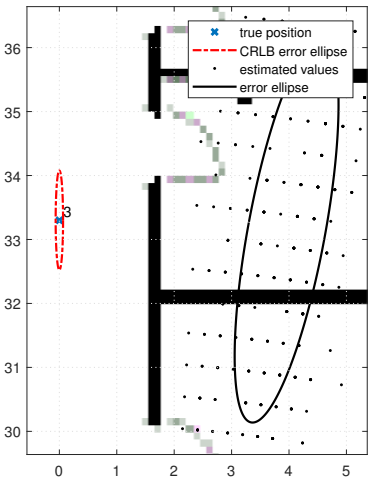
(l) CRLB and estimates of position 12



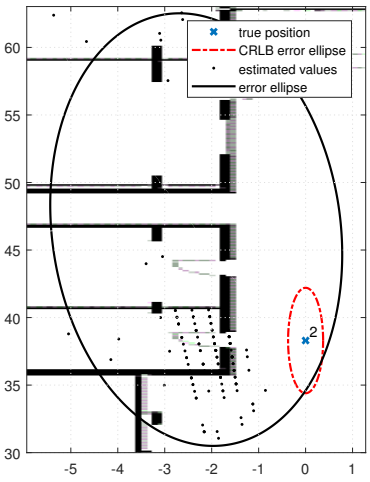
(m) CRLB and estimates of position 13



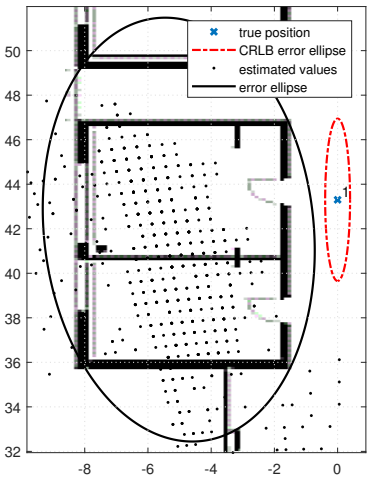
(n) CRLB and estimates of position 14



(o) CRLB and estimates of position 15



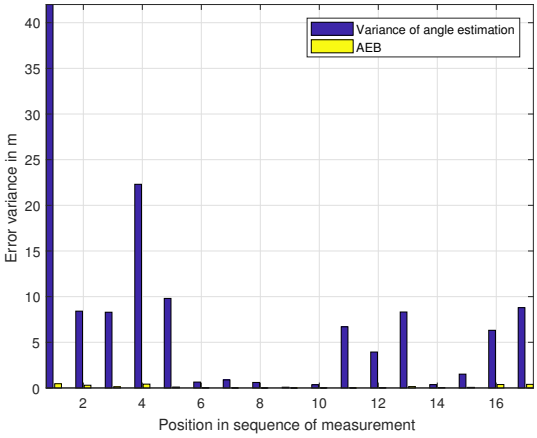
(p) CRLB and estimates of position 16



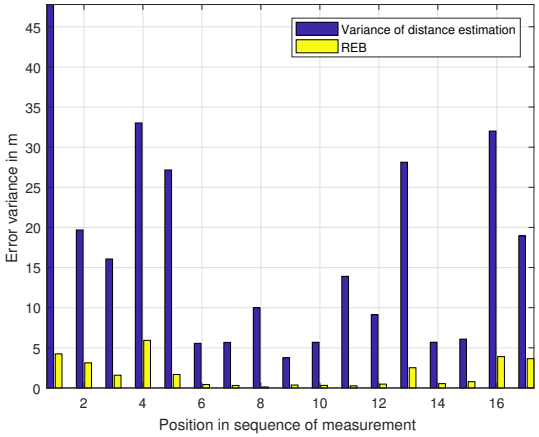
(q) CRLB and estimates of position 17

Figure 5.7: Error ellipses for the CRLB and actual estimates of measurement 2. x and y axes are not eqal.

5.5.3 Errorbars of CRLB and Estimation



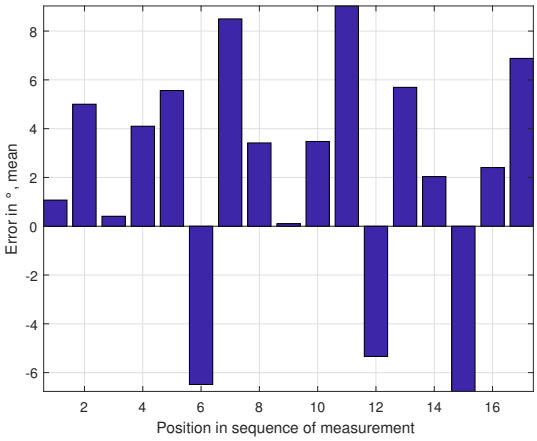
(a) Errorbars of AEB and variance of estimated angles



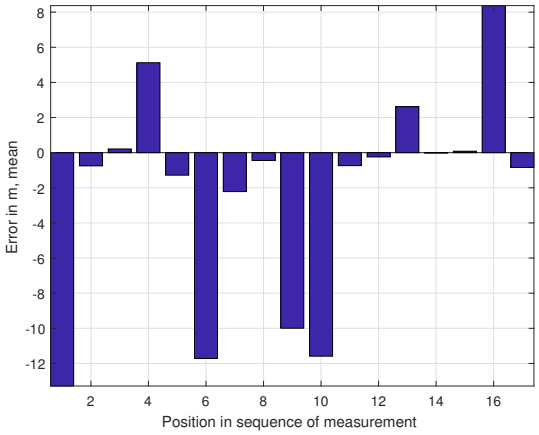
(b) Errorbars of REB and variance of estimated distances

Figure 5.8: Error variance for AEB and REB and distance and angle estimates, measurement 2

Figure 5.8(a) shows the error bars for the AEB and the estimated angles. Here, the estimation of reflections is seen once more (for example at position 6). Figure 5.8(b) again shows the error bars for the REB and the estimated distances. Here again, one can see the problems arising form erroneous synchronization: Large error ranges, and biases for some measurements. These results are comparable to the results from measurement 1.



(a) Mean error of estimated angles



(b) Mean error of estimated distances

Figure 5.9: Mean for angle and distance error, measurement 2

Figure 5.9(a) shows the bias of the angle estimation, the reflections can be seen in more detail. Figure 5.9(b) shows the bias of the distance estimation.

5.6 Evaluation of Measurement 3

5.6.1 Estimated SNR

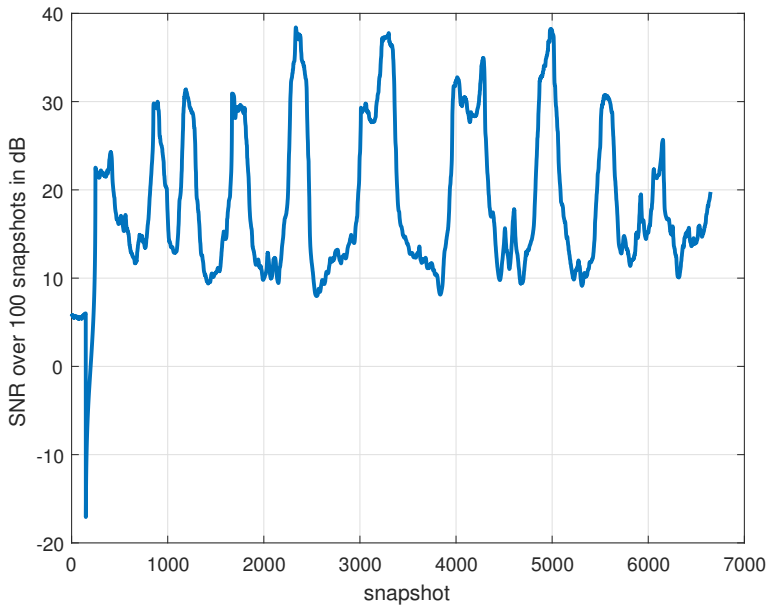


Figure 5.10: SNR of measurement 3 with $L = 100$

Figure 5.10 shows the estimated SNR for measurement 3. The first few samples are to be ignored, as the MS was not transmitting data yet. Otherwise, the SNR seems to be in general about 10dB better than the SNR of measurement 2, especially when the MS was in movement.

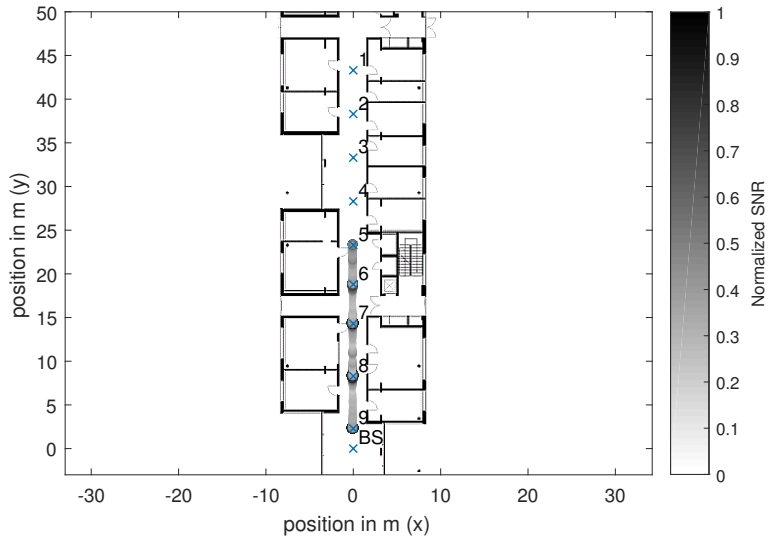
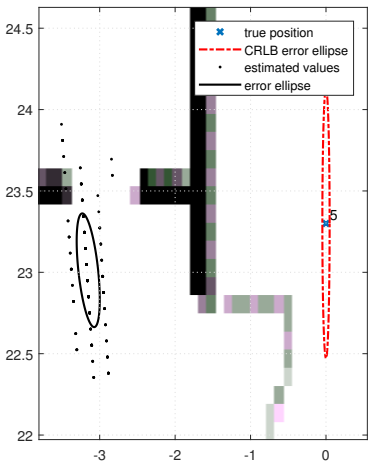


Figure 5.11: SNR of measurement 3 overlayed over the path, bigger marker width and darker color denotes better SNR. The SNR in dB has been normalized to [0,1]; 1 being the maximum SNR.

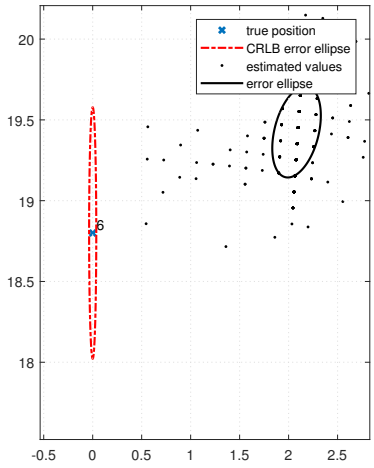
Figure 5.11 shows the SNR plotted over the the true path, again darker color denoting better SNR. The results show better SNR when standing still and lower SNR when moving, but the difference is less compared to measurement 2.

5.6.2 CRLB Ellipses

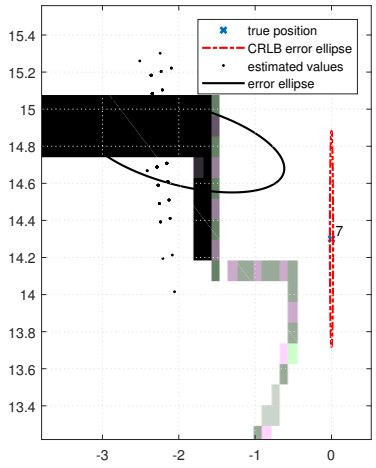
Figures 5.12(a-i) show the error ellipses for estimated values and the CRLB, comparable to measurement 2. A bias in angle estimation can be seen, for the same reasons as in measurement 2. But otherwise, the distance estimation seems to be much more reliable and showing useful results. Some estimates seem to even be better than the CRLB, but similar to an explanation done for measurement 2, this is clearly the result of a bad estimate for the SNR. Comparing Figure 5.12(b) with Figure 5.12(i) one can see that the latter has a bigger error ellipse, although data points suggest a better error ellipse. This is because for Figure 5.12(i), there are other clusters of data points, which can not be seen. Figure 5.12(b) is the other way around: There are multiple data points plotted over each other, causing a smaller error ellipse than suggested.



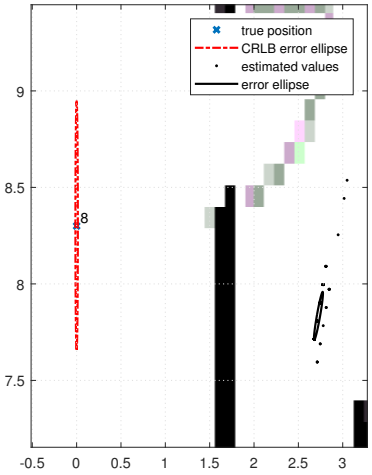
(a) CRLB and estimates of position 1



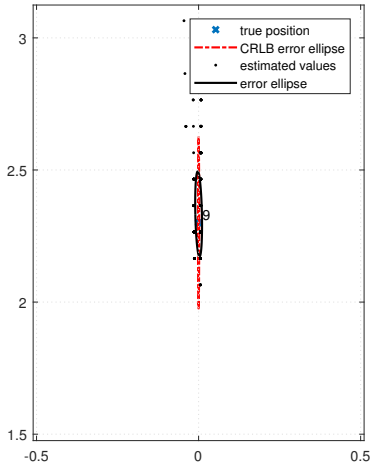
(b) CRLB and estimates of position 2



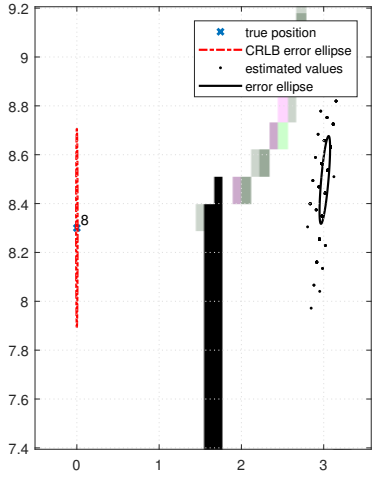
(c) CRLB and estimates of position 3



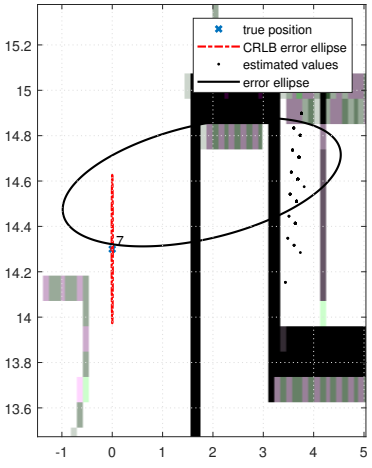
(d) CRLB and estimates of position 4



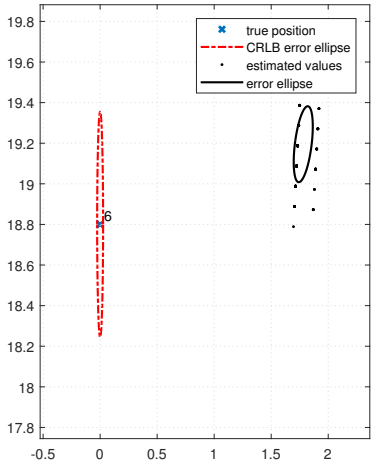
(e) CRLB and estimates of position 5



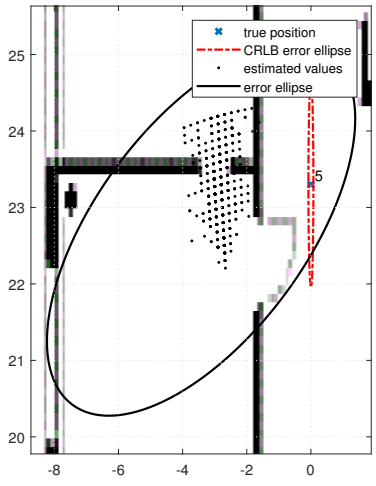
(f) CRLB and estimates of position 6



(g) CRLB and estimates of position 7



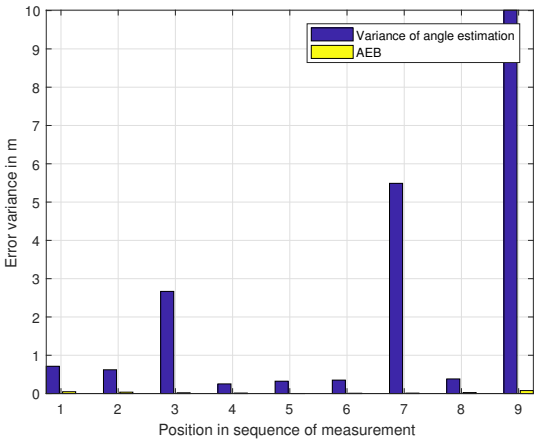
(h) CRLB and estimates of position 8



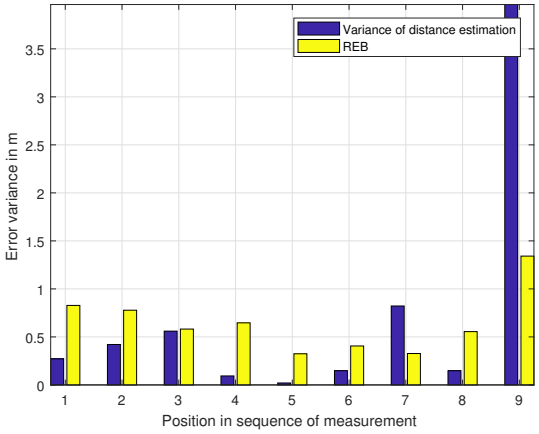
(i) CRLB and estimates of position 9

Figure 5.12: Error ellipses for the CRLB and actual estimates of measurement 3. x and y axes are not eqal.

5.6.3 Errorbars of CRLB and Estimation



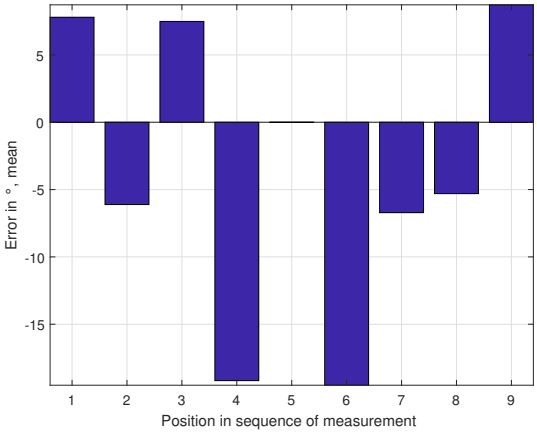
(a) Errorbars of AEB and variance of estimated angles



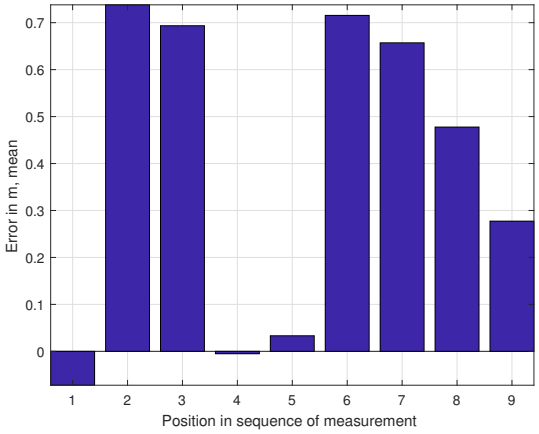
(b) Errorbars of REB and variance of estimated distances

Figure 5.13: Error variance for AEB and REB and distance and angle estimates, measurement 3

Figure 5.13(a) shows the AEBs and the error variance of the estimated angle. Results for this figure are comparable to the results from measurement 2, and show the same bias in estimation. This bias can be attributed to the same problem: Reflections being dominant over the LoS path. More interestingly, Figure 5.13(b), showing the REBs and the estimated distances, shows much improvement over measurement 2. This clearly shows that a better synchronization between BS and MS leads to a better performance of the algorithms. It should be noted, that although it seems that the REB is larger than the estimation itself, the calculated REB is relying on an estimation of the SNR (see (5.9)), and not on an actual measured value. For example destructive interference can lead to a CRLB which is not representing the real value.



(a) Mean error of estimated angles



(b) Mean error of estimated distances

Figure 5.14: Mean for angle and distance error, measurement 3

Figure 5.14(a) is comparable to the results of measurement 2, the angle estimates have the same problems as described for measurement 2. Figure 5.14(b) shows the bias of the estimation being much smaller for distance than the bias of measurement 2.

6 Conclusion and Further Outlook

6.1 Conclusion

This work shows that indoor positioning is possible using an OFDM based, 8x1 MIMO system developed for field-testing of massive MIMO systems. Still, there are several problems that need to be solved, the largest being an insufficient synchronization between BS and MS on the specific hardware used here. Furthermore, the algorithm used for estimation now only accounts for LoS components, and thus is neglecting information which could be extracted from the estimated TFs. Nonetheless, results especially from an indoor environment with complex geometries are very promising, as seen in the third measurement. The mean error for distance estimation was less than 40cm, and the mean error for angle estimation was less than $-4°$. The error variance for the distance estimation was less than 10cm, and the error variance for angle estimation was less than 22cm, translated to position estimation. These values hold only true for LoS conditions.

6.2 Outlook

Further research with hardware of this kind should focus primarily on a reliable OTA synchronization, which could be implemented by analyzing the errors measured here, and taking a deeper look at the programs from NI, or by implementing a time division duplex based synchronization scheme. Also, the algorithms used could be improved to model MPCs in a better way to improve general positioning. Furthermore, performance of the measured data could be evaluated for less antennas, for example using only 2 of the 8 antennas, and seeing if the results are still reliable enough.

Other research could focus on implementing the algorithms used here in real-time, and in smaller, more consumer-oriented hardware. As Wi-Fi 6 is already on the doorstep during completion of this work [4], and the standard specifies similar antenna configurations and bandwidths, one could try to implement such algorithms on a standard compliant Wi-Fi node. As Wi-Fi 6 does not allow for time division duplex, a workaround for this would have to be found, but the specification allows for example trigger frames, which could be used to achieve similar functionality.

Overall, as computing power and MIMO systems get increasingly cheaper, the research done here could hopefully lead to much better indoor positioning in the near future.

7 Appendix and Bibliography

7.1 Acronyms

AEB angulation error bound

AoA angle of arrival

AWGN additive white Gaussian noise

BS base station

CIR channel impulse response

CRLB Cramer-Rao lower bound

FPGA field programmable gate array

GNSS global navigation satellite system

IDFT inverse discrete Fourier transform

LMMSE linear minimum mean square error

LoS line-of-sight

MIMO multiple input multiple output

ML maximum-likelihood

MPC multi path component

MS mobile station

NI National Instruments

OFDM orthogonal frequency division multiplexing

OTA over the air

PAM pulse amplitude modulation

PCI-X peripheral component interconnect extended

PDF probability density function

REB ranging error bound

SDR software defined radio

SNR signal-to-noise ratio

SPSC Signal Processing and Speech Communications Laboratory

TF transfer function

ToA time of arrival

ULA uniform linear array

WSSUS wide-sense stationary and uncorrelated scattering

7.2 Still Frames of Measurement Videos

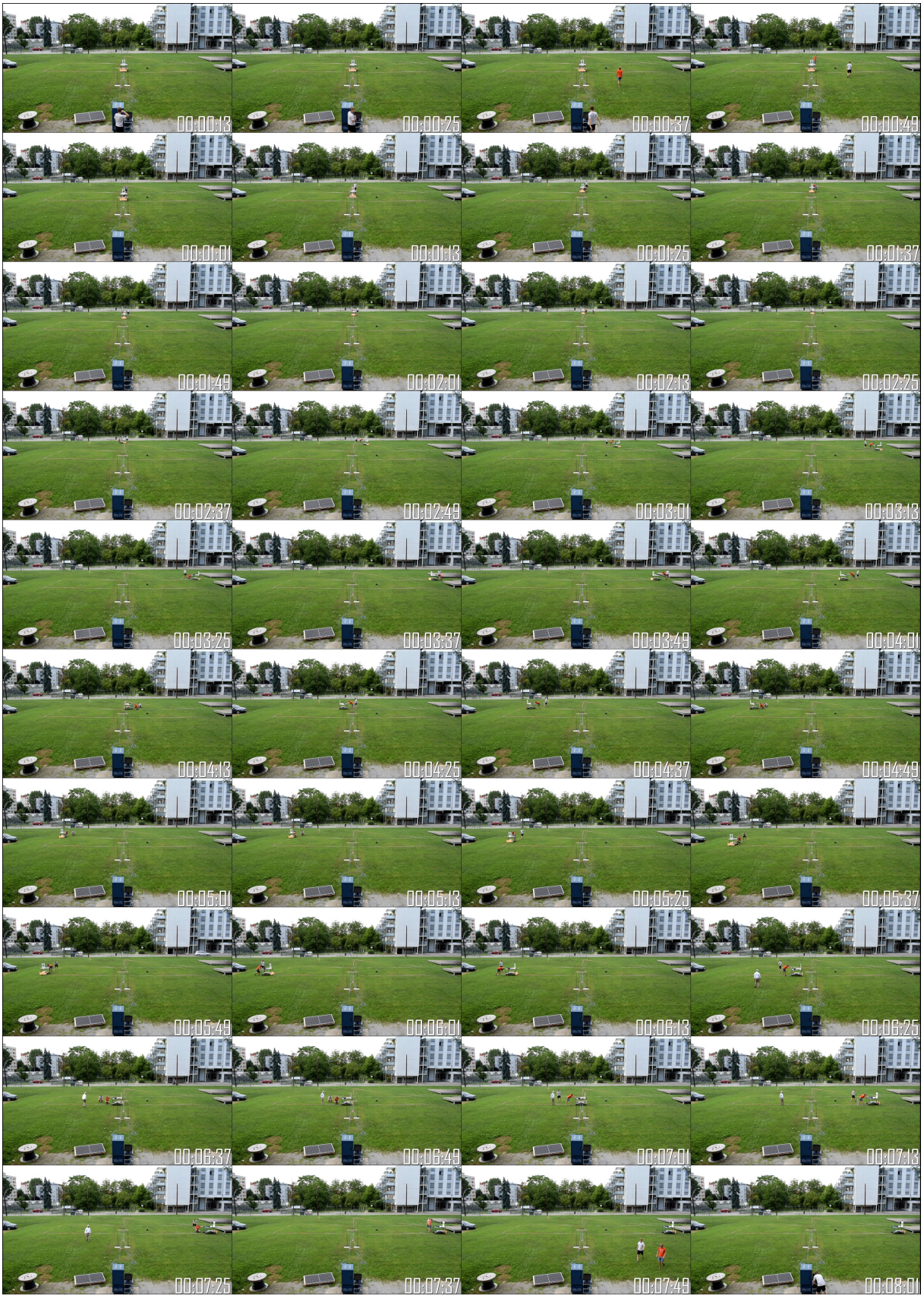


Figure 7.1: Still frames of measurement 1

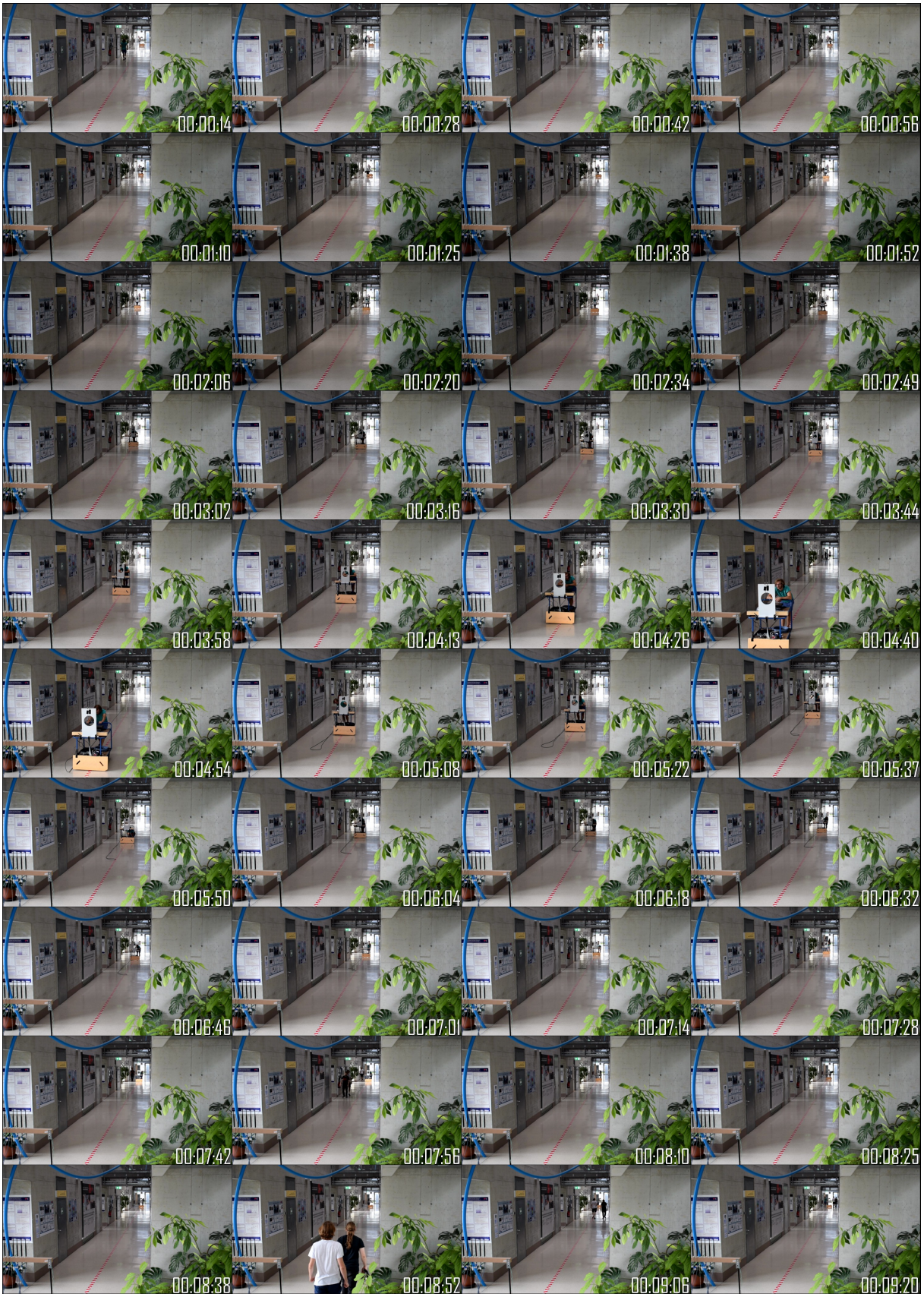


Figure 7.2: Still frames of measurement 2

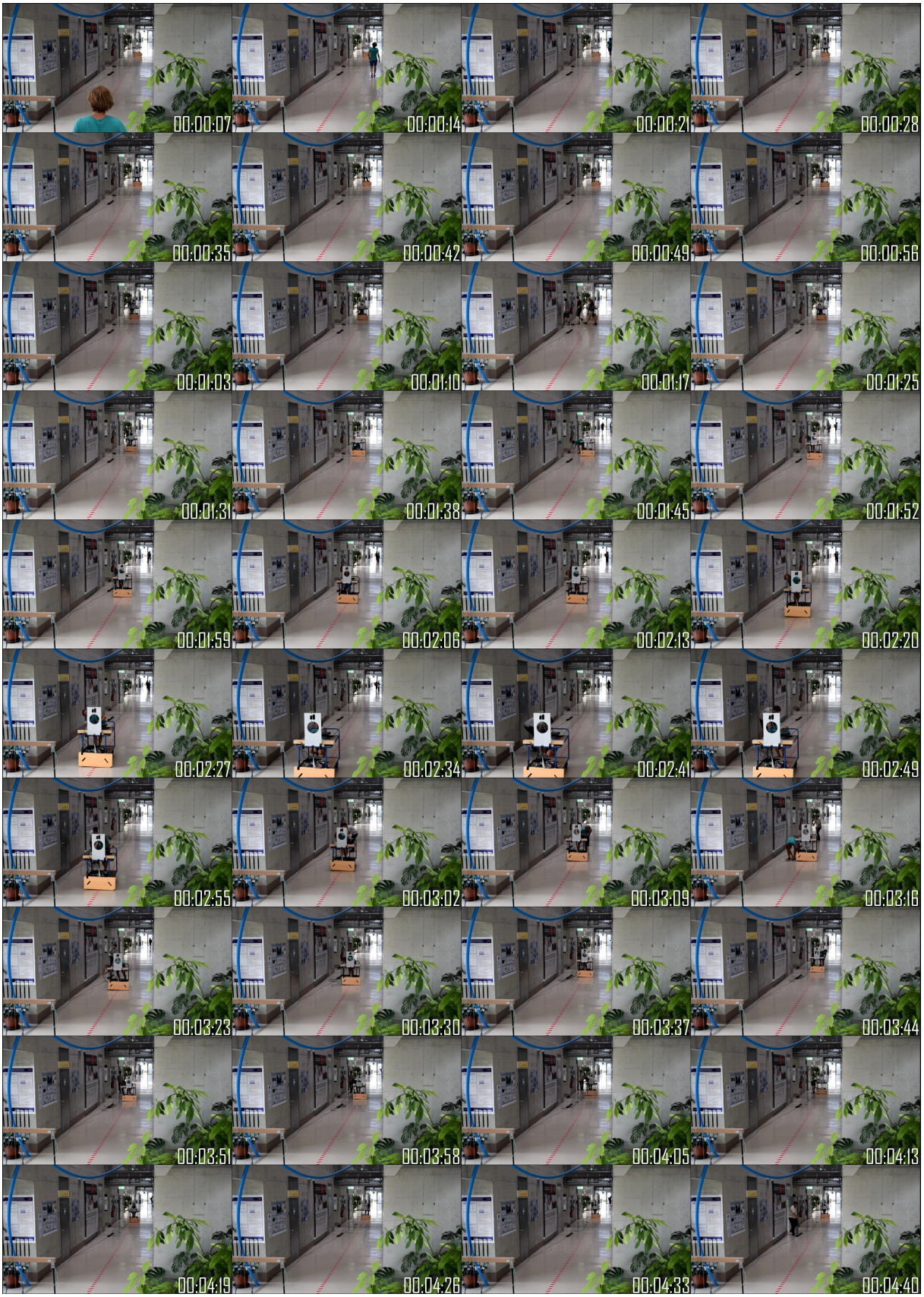


Figure 7.3: Still frames of measurement 3

7.3 Excerpts of the Code (Matlab)

```
function s = generate_pulse(md, tau_0, tau, normalize,f)
%function s = generate_pulse(md, tau_0, tau, normalize)
%
% Generates a pulse (RC, RRC) specified by md at position tau_0.
%The pulse is defined over the vector tau, with optional
%normalize is optional:
% 1: normalize to unit maximum amplitude
% 2: normalize to unit energy (discrete time)
% 3: normalize to unit energy (continuous time, acc. to tau vector)
%
% Example:
% md.type = 'RRC';
% md.Tp = 1e-9;
% md.beta = 0.6;
% tau = (-5*md.Tp:1e-12:5*md.Tp)';
% tau_0 = 1.23456e-12;
% s = generate_pulse(md, tau_0, tau, 0);
% figure; plot(tau, s);


% based on: Paul Meissner, Dec. 2012, update Feb. 2016 Stefan Hinteregger
% adapted:  Thomas Wilding, Jan. 2019, Andreas Fuchs, Nov. 2019

%% Initialize and preliminaries
if( ~(strcmp(md.type, 'RC') || strcmp(md.type, 'RRC') || strcmp(md.type, 'sinc')) )
  error('Only RC, RRC and sinc pulse supported!')
end

tau = tau(:);

if length(tau) > 1
  tau_res = tau(2) - tau(1);
else
  tau_res = 0;
end
%% RC
if strcmp(md.type, 'RC')
  t = tau-tau_0;
  i=find(t==0);
  % RC pulse
  a = sin(pi*t/md.Tp);
  b = cos(pi*md.beta*t/md.Tp);
  c = pi*t/md.Tp;
  d = 1-4*md.beta^2*t.^2/md.Tp^2;

  s = a.*b./(c.*d);
  s(abs(abs(t) - 0) < tau_res/1e3) = 1;
  s(abs(abs(t) - md.Tp/2/md.beta) < tau_res/1e3) = sinc(1/2/md.beta)*pi/4;
%% RRC
elseif strcmp(md.type, 'RRC')
  t = bsxfun(@minus,tau,tau_0);
  i=find(t==0);
  tau_res = tau(2)-tau(1);

  a = 4*md.beta*cos(pi*(1+md.beta)*t/md.Tp);
  b = sin(pi*(1-md.beta)*t/md.Tp);
  c = md.Tp./t;
  d = 1-16*md.beta^2*t.^2/md.Tp^2;

  s = 1/(pi*sqrt(md.Tp)) * (a+b.*c)./d;
  s(i) = 1/sqrt(md.Tp)*(1 - md.beta + 4*md.beta/pi);
  s(abs(abs(t) - md.Tp/4/md.beta) < tau_res/1e3) = ... %do not use multiples of 4*beta for Ts!
    md.beta/sqrt(2*md.Tp)*( (1+2/pi)*sin(pi/4/md.beta) + (1-2/pi)*cos(pi/4/md.beta) );
elseif strcmp(md.type, 'sinc')% tau is the frequency vector
  s = ifft(ones(size(f)).*exp(-2i*pi*f*tau_0))';
end
%% If desired, normalize
if(nargin > 3)
  if(normalize == 0)
    if strcmp(md.type, 'RC')
      normFactor = sqrt((1-md.beta/4)*md.Tp);
    else
      normFactor = 1;
    end
  elseif(normalize == 1)
    normFactor = max(abs(s));
  elseif(normalize == 2)
    normFactor = sqrt(sigEnergy(s));
  elseif(normalize == 3)
    normFactor = sqrt(sigEnergy(s, tau));
  end

  s = s./normFactor;
end
```

Listing 7.1: generate_pulse.m

```
function [lhf] = estimator_joint_ML_AOA_TOA(theta,r_w,time,md,array)
% (I)&(II): estimator_joint_ML_AOA_TOA(aoa,Cinv,r,alpha,time,md,array)
% (III): for fminsearch and ML estimator
% function for joint ML estimate of ToA and AoA. needs estimated alphas.
c = 299792458;
r_w = r_w(:);
%---- (III): joint estimates to use as function for fminsearch()
% estimate amplitudes

s = get_array_signals(array, theta(1), theta(2)/c, time, md);
s_w = s;

a_hat = (s_w'*s_w)^-1*s_w'*r_w; %estimate alpha at current tau
lhf = +(r_w - a_hat*s_w)'*(r_w - a_hat*s_w)*1e6; % + due to fminsearch

if theta(2) >= 200
    lhf = inf;
end
if theta(2) <= 0.8
    lhf = inf;
end
end
```

Listing 7.2: estimator_joint_ML_AOA_TOA.m

```
function [s_array] = getArraySignals(array, aoa_mpc, tau_mpc, time, md)
% outputs the array signal for a single pulse s in a stacked vector of size
% M*N. needs the correct array struct
%  M ... array elements
%  N ... signal length in samples
%  aoa_mpc ... angle of arrival (single value)
%  tau_mpc ... time of arrival at array center (single value)
%
%  s_array ... stacked array signals
%
% Thomas Wilding, January 2018
%                 July 2018
%                August 2018
switch ~isfield(time,'N_sig_samp')
  case isfield(time,'N')
    time.N_sig_samp = time.N;
end

NORMALIZE = 1;

s_array = zeros(time.N_sig_samp*array.M,1);

s = generate_pulse(md, tau_mpc, time.tau, NORMALIZE, md.f).*exp(-2i*pi*md.fc*tau_mpc);
tau_delay = get_array_delays(array,aoa_mpc);

for mm = 1:array.M
  s_array((mm-1)*time.N_sig_samp+1:mm*time.N_sig_samp,1) =
        shift_pulse(s,time.tau,tau_delay(mm),md);
end
```

Listing 7.3: get_array_signals.m

```
function [deltatau] = get_array_delays(array,aoa)
% get the delays for each array element corresponding to a certain aoa
% the element position vectors in array.pm should be column vectors
% Thomas Wilding, 2018/01
c = 299792458;

aoa = aoa(:)';

if size(array.pm,2) == array.M
  deltatau = -1/c*( (array.pm(1,:)-array.p(1))*cos(aoa - array.psi) + ...
    (array.pm(2,:)-array.p(2))*sin(aoa - array.psi) )';
elseif size(array.pm,2) ~= array.M
  deltatau = -1/c*( (array.pm(:,1)-array.p(1))*cos(aoa - array.psi) + ...
    (array.pm(:,2)-array.p(2))*sin(aoa - array.psi) )';
  warning('Check the if array delays are correct!')
end
```

Listing 7.4: get_array_delays.m

```
%% grid search
time_loop = 1;
min = inf;
for i = 1:1:num_samples
    tic
    y = squeeze(bs_ifft(:,LAYER+1,:,i));

    %normalize CIR for every Antenna to 1
    for k = 1:MAX_ANTENNA
        y(:,k) = y(:,k)/ max(abs(y(:,k)));
    end

    dist_est_grid = 40:5:80;
    alpha_est_grid = -80:2:80;
    alpha_est_grid = alpha_est_grid.*pi./180;

    clc;
    disp(['start of search ' num2str(i)]);
    disp([num2str(ceil(((i-1)/num_samples) * 100)) '% of grid search']);
    disp([num2str(time_loop * (num_samples - i)) ' seconds remaining (estimated)']);

    for k = 1:1:length(dist_est_grid)
        for m = 1:1:length(alpha_est_grid)
            minvar = estimator_joint_ml_aoa_toa([alpha_est_grid(m)
                dist_est_grid(k)],y,time,signal,array);
            if minvar < min
                min = minvar;
                minplace = [m, k];
                dist_min(i) = dist_est_grid(k);
                alpha_min(i) = alpha_est_grid(m);
                %disp(['New min for dist: ' num2str(dist_min(i))]);
                %disp(['New min for alpha: ' num2str(alpha_min(i))]);
            end
        end
    end
    min = inf;
    try
        alpha_min(i);
        dist_min(i);
    catch
        alpha_min(i) = 0;
        dist_min(i) = 50;
    end

    dist_est_grid = (dist_min(i)-2.5):1:(dist_min(i)+2.5);
    alpha_est_grid = (alpha_min(i)-(1*pi/180)):(0.5*pi/180):(alpha_min(i)+(1*pi/180));

    %second finer run

    for k = 1:1:length(dist_est_grid)
        for m = 1:1:length(alpha_est_grid)
            minvar = estimator_joint_ml_aoa_toa([alpha_est_grid(m)
                dist_est_grid(k)],y,time,signal,array);
            if minvar < min
                min = minvar;
                minplace = [m, k];
                dist_min(i) = dist_est_grid(k);
                alpha_min(i) = alpha_est_grid(m);
                %disp(['New min for dist: ' num2str(dist_min(i))]);
                %disp(['New min for alpha: ' num2str(alpha_min(i))]);
            end
        end
    end
    min = inf;
    try
        alpha_min(i);
        dist_min(i);
    catch
        alpha_min(i) = 0;
        dist_min(i) = 50;
    end

    %third run, only for distance
    dist_est_grid = 40:0.5:80;
    alpha_est_grid = alpha_min(i);

    for k = 1:1:length(dist_est_grid)
        for m = 1:1:length(alpha_est_grid)
            minvar = estimator_joint_ml_aoa_toa([alpha_est_grid(m)
                dist_est_grid(k)],y,time,signal,array);
            if minvar < min
                min = minvar;
                minplace = [m, k];
                dist_min(i) = dist_est_grid(k);
                alpha_min(i) = alpha_est_grid(m);
            end
        end
    end
    min = inf;
end
```

Listing 7.5: Excerpt of estimation_main.m, grid search

7.4 Excerpts of the Code (Labview)

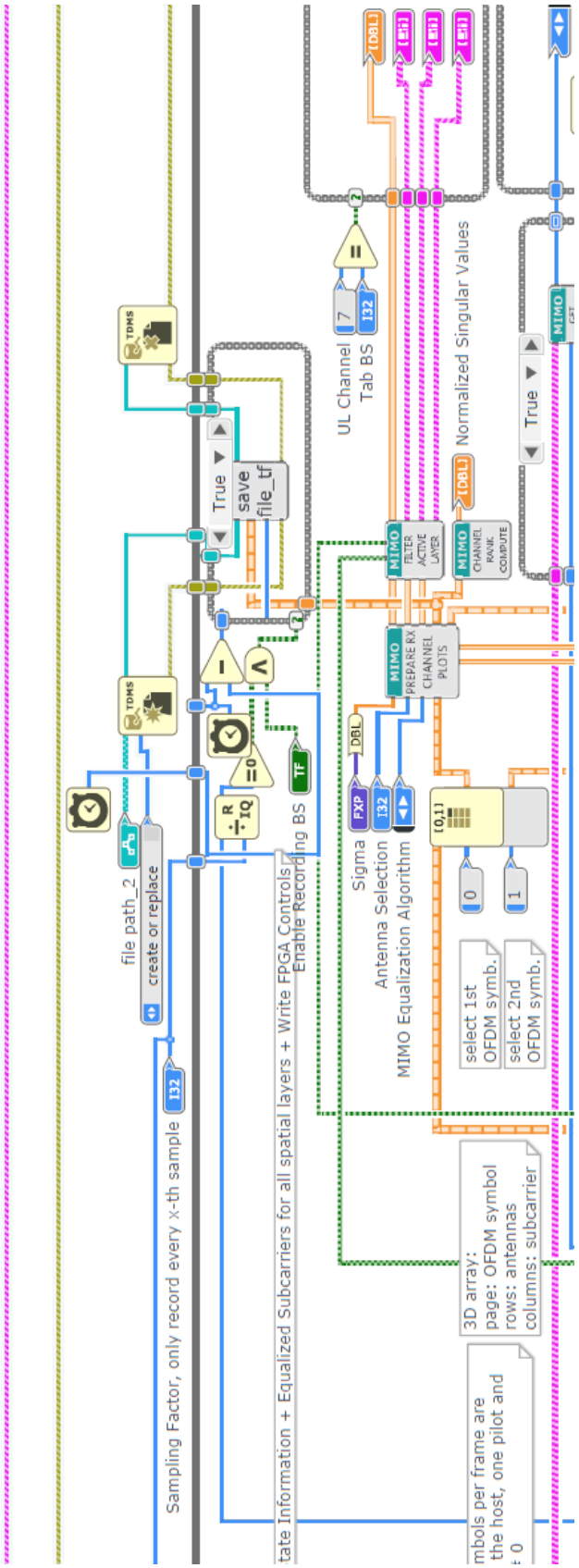


Figure 7.4: Base Station Host.gvi, excerpt

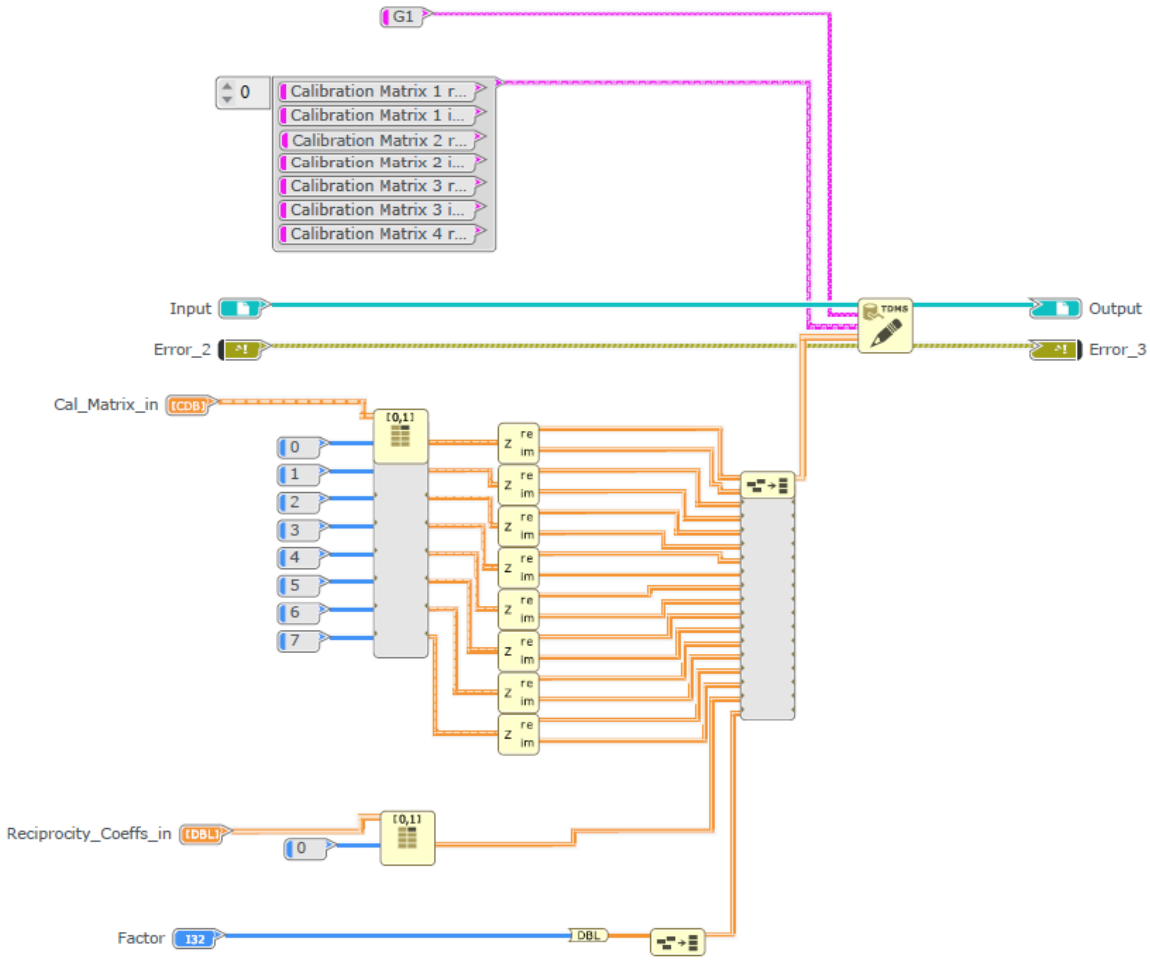


Figure 7.5: save file_cal.gvi

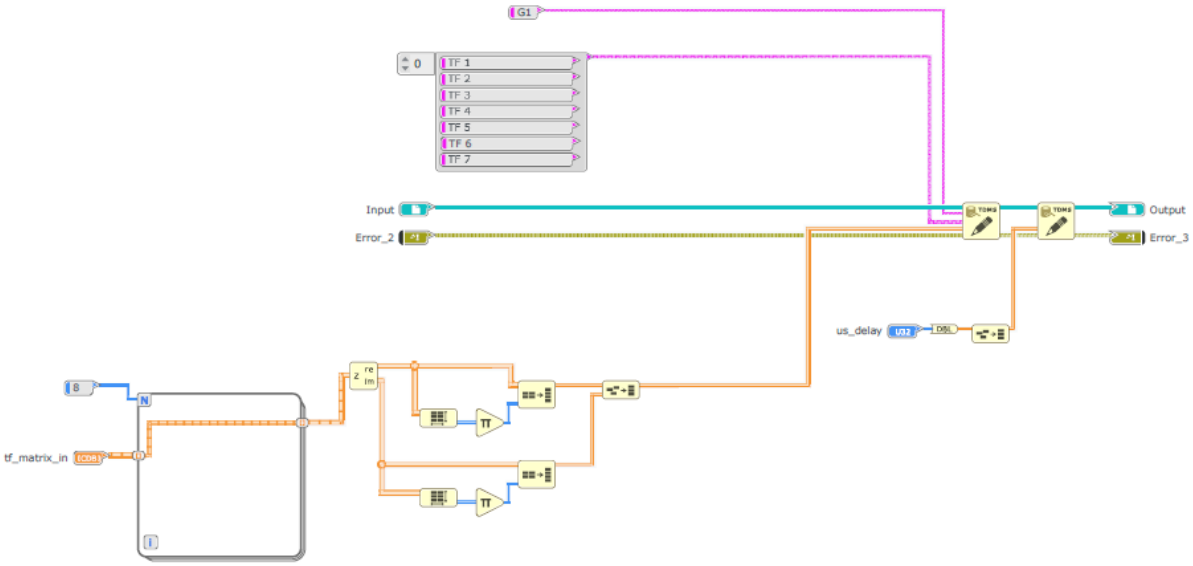


Figure 7.6: save file_tf.gvi

Bibliography

[1] R. Di Taranto, S. Muppirisetty, R. Raulefs, D. T. Slock, T. Svensson, and H. Wymeersch, "Location-aware communications for 5G networks: How location information can improve scalability, latency, and robustness of 5G," *IEEE Signal Processing Magazine, Volume:31, Issue: 6, October 15, 2014*, 10 2014. [Online]. Available: http://www.eurecom.fr/publication/4432

[2] N. Garcia, H. Wymeersch, E. G. Larsson, A. M. Haimovich, and M. Coulon, "Direct localization for massive mimo," *Trans. Sig. Proc.*, vol. 65, no. 10, pp. 2475–2487, May 2017. [Online]. Available: https://doi.org/10.1109/TSP.2017.2666779

[3] X. Li, E. Leitinger, M. Oskarsson, K. Åström, and F. Tufvesson, "Massive mimo-based localization and mapping exploiting phase information of multipath components," *arXiv preprint arXiv:1811.04494*, 2018.

[4] M. Turner, "Wi-Fi 6 Explained: The Next Generation of Wi-Fi," https://www.techspot.com/article/1769-wi-fi-6-explained/, 2019, accessed: 2019-12-09.

[5] A. F. Molisch, *Wireless Communications, Second Edition.* Wiley, 2011.

[6] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory.* Prentice Hall, 1993, vol. 1.

[7] M. Zhu, J. Vieira, Y. Kuang, K. Åström, A. F. Molisch, and F. Tufvesson, "Tracking and positioning using phase information from estimated multi-path components," in *Communication Workshop (ICCW), 2015 IEEE International Conference on.* IEEE, 2015, pp. 712–717.

[8] G. Golub and V. Pereyra, "The differentiation of pseudo-inverses and nonlinear least squares problems whose variables separate," *SIAM Journal on Numerical Analysis*, vol. 10, no. 2, pp. 413–432, 1973.

[9] J. Hokanson, "TDMS Reader," https://de.mathworks.com/matlabcentral/fileexchange/30023-tdms-reader, 2017, accessed: 2019-12-02.

[10] "Gebäudeplan TU Graz," https://online.tugraz.at/tug_online/ris.ris?pOrgNr=37&pActionFlag=A&pQuellGeogrBTypNr=5&pQuellGeogrBerNr=3640001&pZielGeogrBTypNr=4&pZielGeogrBerNr=&pRaumNr=5652&pShowEinzelraum=J&pRaumPKs=&pVirtGeogrBerPKs=&pCoordsX=&pCoordsY=&pDummySelectBox=3640001&pZoomFaktor=10&pAnsichtNr=1&pKombinationNr=2, 2019, accessed: 2019-12-10.

[11] W. Grabinski, B. Nauwelaers, and D. Schreurs, *Transistor Level Modeling for Analog/RF IC Design*, 01 2006.

[12] T. Wilding, S. Grebien, E. Leitinger, U. Mühlmann, and K. Witrisal, "Single-anchor, multipath-assisted indoor positioning with aliased antenna arrays," 2018.

[13] T. Wilding, S. Grebien, U. Mühlmann, and K. Witrisal, "Accuracy bounds for array-based positioning in dense multipath channels," *Sensors*, vol. 18, no. 12, p. 4249, 2018.