TECHNICAL UNIVERSITY OF CRETE
SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

Indoor RF Localization with Algebraic Methods

by

Spyridon Peppas

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DIPLOMA OF
ELECTRICAL AND COMPUTER ENGINEERING

October 2021

THESIS COMMITTEE
Professor Aggelos Bletsas, Thesis Supervisor
Professor George N. Karystinos
Associate Professor Michail G. Lagoudakis

Abstract

This work investigates the problem of localizing batteryless, passive, ultra-low cost RFID tags, under the presence of non-white noise that multipath introduces inside buildings. Two recently proposed grid-based methods are utilized to tackle the aforementioned problem with radio signals. The first method is borrowed from the through-the-wall radar imaging community and this thesis adapts it to the case of monostatic interrogation of Gen2 tags, with commercial RFID readers. The method exploits the strength of the received signal (RSSI) and its phase, expressed in a linear system that incorporates the impact of known reflectors. The second one is a phase-based method, which puts forth a "differential mitigation" scheme based on maximum likelihood estimation (MLE).

For the first method, it is found that certain ambiguities need to be resolved, relevant to the wavelength, which can be mitigated based on the fundamental theory of compressive sensing. Specifically, it is observed that the so-called sensing matrix, can lead to a successful reconstruction by properly selecting the carrier frequency or spreading the measurements, with respect to the coherence of its columns or its blocks.

As for the second method, a selection of a phase measurement is needed; it is found that if more than one phase measurements are randomly selected, then a more accurate estimate of the target is achieved. According to simulations, the random selection of a subset of the phase measurements as a reference can provide up to 50% improvement, concerning root mean squared error.

This thesis includes both simulation and experimental results using a commercial Gen2 RFID reader at UHF, installed on a robotic platform. Experimental results with the available equipment show that, given static poles, mean absolute localization error in the order of 10 cm for indoor 2D localization is achieved. On the other hand, when the RFID reader is placed on a mobile robotic platform, it is shown that mean absolute localization error in the order of 19cm is possible. Finally, a proof-of-concept is offered, by the modeling of a single reflector, which leads to about 19% 3D localization error improvement, compared to the case where the reflector is ignored. Further examination of this finding perhaps opens a fruitful new research direction.

Acknowledgements

First and foremost, I would like to thank my supervisor *Prof. Aggelos Bletsas*, for exposing me to research and giving me the opportunity to work in his lab. He was always supporting my ideas and provided me more than the necessary support to complete this work. Moreover, I want to thank *Prof. Athanasios Liavas* who provided his experience and knowledge generously, his passion for studying advanced mathematical topics inspired me deeply.

I have enormous appreciation to my lab colleagues *George V.*, *Vaggelis G.*, *Konstantinos S.*, *Iosif V. Manos A.*, *George K.* and *Roza C.* who, in addition to being very good at their work, are above all excellent people. Special thanks to *Vaggelis G.* and *Manos A.* who supported me in this work. Overall, a thank you to all of you fellas for the encouragement I received in difficult times and your patience in everything we did. P.S If you want to go quickly, go alone. If you want to go far, go together.

I would like to thank all of my friends for the adorable moments I had with and for the psychological support that provided me in harsh moments.

Finally, I would like to thank my family for their unconditional support in all these years.

Table of Contents

List of Figures

List of Abbreviations

ADM Alternating Direction Method

CLF Cosine Likelihood Function

CS Compressive Sensing

EPC Electronic Product Code

LASSO Least Absolute Shrinkage and Selection Operator

LoS Line-of-Sight

MAE Mean Absolute localization Error

MLE Maximum Likelihood Estimation

NLF Naive Likelihood Function

NSP Null Space Property

PDF Probability Density Function

RFID Radio-Frequency IDentification

RIP Restricted Isometry Property

RMSE Root Mean Squared localization Error

RSSI Received Signal Strength Indication

RSWCLA Random Sampling Weighted Cosine Likelihood Algorithm

SNR Signal-to-Noise Ratio

SpaRSA Sparse Reconstruction by Separable Approximation

STD STandard Deviation

UHF Ultra High Frequency

USRP Universal Software Radio Peripheral

UUP Uniform Uncertainty Principle

WCLA Weighted Cosine Likelihood Algorithm

YALL1 Your ALgorithms for L1

Chapter 1

Introduction

1.1 Discussion

Nowadays, RFID tags have been widely deployed. It is not absurd to say that we are almost surrounded by RFID tags on a daily basis. For example, RFID tags are used in clothing stores, credit cards, ID cards, car keys, and more. The attraction of their interest is because they are very cost-effective and do not require an external power supply. Generally, an RFID system consists of a reader and multiple tags. Furthermore, special interest has been expressed in determining the location of the tags included in the system. The aforementioned system could be an application to a central library to sort books or identify their locations.

Certain research works have focused on the positioning of RFID tags with considerable success but with some overhead, as they require reference tags [1],[2],[3]. However, there are several challenges to be faced, especially in room interiors, where in addition to the backscattered tag signal, reflections are received from adjacent walls, which can degrade the tag's localization performance, if not properly modeled.

In this thesis, two grid-based methods are utilized to tackle the problem of tag localization with radio signals, which are independent of reference tags. The first utilizes RSSI and phase measurements, and solves a linear system that can accommodate possible reflections in a room, under some Compressive Sensing (CS) assumptions. These reflections are essentially expressed as signals that have characteristics that depend on the geometry of the space, which, if taken into account, can satisfactorily improve the target estimation. The second method and its variants exploit only phase measurement differences, which are capable of eliminating uncertainties that stem from reflections and thermal noise.

Simulations and experimental campaigns were conducted to evaluate the methods under light or harsh environmental conditions. A reader was installed on both a static pole and a robotic platform, through which measurements were captured at different power levels. It is found that the involved methods demonstrate some resilience to multipath conditions and the uncertainty that the robotic platform may induce, despite the under-modeling assumed in most campaigns. However, it was confirmed via simulations and experiments that modeling part of the multipath can certainly lead to an improvement in terms of mean error. Given a 2d topology, localization error under light multipath was found in the order of less than 2 cm in the best case. Moreover, by utilizing static poles under strong multipath, mean absolute localization error ranged between 9.4 cm - 26.83 cm. The error through the robotic platform increased, with the most favorable case giving an average error of 18.7 cm. Additionally for 3D localization, 30 tags were placed in a room without many obstacles, where mean absolute localization error ranged from 17.9 cm to 35.2 cm for transmission power regulated at 30

dBm. Finally, an around 19 % improvement was observed, by modeling a significant reflector at 30 dBm.

1.2 RFID System - Measurements

To begin with, an RFID reader is the brain of the RFID system and is necessary for any system to function. Readers, also called interrogators, are devices that transmit and receive radio waves in order to communicate with RFID tags.

As we mentioned in the previous section, the RFID system consists of one reader and numerous tags. The batteryless tags utilize the backscatter radio link for communications; through the link, tags modulate their information on the backscattered signals emitted from a reader. A backscatter tag close to the reader variates its impedance load on its antenna to modulate the reader's signal, in order to produce a message with ones and zeros.

Regarding the localization part, low-level user data, such as RSSI and phase, can be provided by most of the commercial RFID readers, which brings opportunities to realize fine-grained localization. RSSI is a measure of the strength of a signal received from a passive tag. One could abstractly say that RSSI is something proportional to the amplitude of the received signal and it is used to determine how close a tag is to a reader antenna. In general, the phase depends on the propagation channel and the modulating properties of both Tx/Rx antennas of the reader and the tag. Below a figure is depicted, which describes the communication function between an RFID reader and a tag.
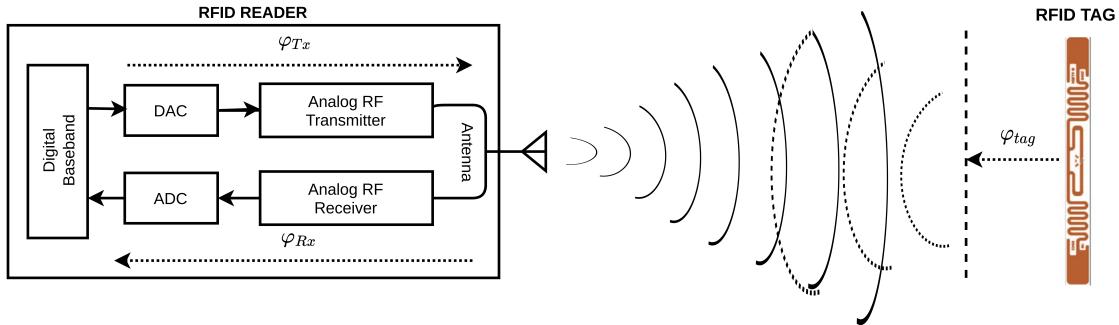
Figure 1: RFID System function, where φ_{Tx} and φ_{Rx} represent the phase shift introduced by the transmitter's and receiver's hardware circuit and cables, respectively. Moreover, φ_{tag} is the phase shift caused by tag's reflection characteristic.

9

Chapter 2

Compressive Sensing

2.1 Signal Model

In this section, a signal model is adopted from [4], which exploits the line-of-sight component and it can be extended in order to accommodate all the possible indirect paths (reflectors). Measurements can be expressed as a linear system, which can be solved through a group sparse optimization technique. The method utilizes both phase and RSSI measurements.

2D Model

Assume that N transceivers are arranged as a line array, with element positions $x_n, n = 0, \ldots, N-1$ in the interior of a room. Let the frequencies f_m, $m = 0, \ldots, M-1$, which are uniformly spaced over the desired bandwidth $f_{M-1} - f_0$.

The region of interest is divided into a regular grid (square shaped) of $N_x \times N_y$ points representing the number of points in crossrange and downrange, respectively. Let σ_p be the complex reflectivity of the p-th spatial grid point or the p-th target, where $p = 0, 1, \ldots, N_x N_y - 1$. Note that the absence of a target at a particular grid point is simply represented by a zero value for the corresponding target reflectivity. More precisely,

$$\sigma_p = \begin{cases} 0, & \text{absence of a target} \\ a, & \text{presence of a target } \{a \in \mathbb{C} \mid a \neq 0\}. \end{cases}$$

Assuming monostatic operation, the target [1] response can be expressed as

$$y[m, n] = \sum_{p=0}^{N_x N_y - 1} \sigma_p \, e^{-j2\pi f_m \tau_{pn}}, \tag{1}$$

where τ_{pn} denotes the round-trip propagation delay between the p-th target and the n-th transceiver.

Given one carrier frequency $f_c = f_m$, $M = 1 \Rightarrow m = 0$ and one target, each measurement in 1 can be found in 4.1. It is worth noting that section 4.1 fully describes the backscattered signal from a tag along with a single possible reflection.

3D Model

The only difference between the 2d model and the 3d model, is that instead of dividing the region of interest into $N_x \times N_y$ points, due to the increased dimension extending on the

---

[1]In our context when we refer to a target we imply the RFID tag that has been illuminated by a reader.

z axis, we divide the space into $N_x \times N_y \times N_z$ points (cube shaped grid in three dimensions). Subsequently, 1 is written as

$$y[m,n] = \sum_{p=0}^{N_x N_y N_z - 1} \sigma_p \, e^{-j2\pi f_m \tau_{pn}}. \tag{2}$$

2.2   Linear System

The measured data vector $\boldsymbol{y} \in \mathbb{C}^{MN \times 1}$ is obtained by stacking the measurements $y[m,n]$ into a single column vector

$$\boldsymbol{y} = [y[0,0], \ldots, y[M-1,0], \ldots, y[M-1, N-1]]^\top. \tag{3}$$

The complex reflectivities σ_p can be vectorized as

$$\begin{aligned} \boldsymbol{s} &= \left[\sigma_0, \sigma_1, \ldots, \sigma_{N_x N_y - 1}\right]^\top \quad, \ \text{2d model} \\ \boldsymbol{s} &= \left[\sigma_0, \sigma_1, \ldots, \sigma_{N_x N_y N_z - 1}\right]^\top \quad, \ \text{3d model.} \end{aligned} \tag{4}$$

The sensing matrix $\boldsymbol{\Phi}$ contains the phase terms of the target model,

$$\begin{aligned} [\boldsymbol{\Phi}]_{ip} = \exp\left(-j2\pi f_m \tau_{pn}\right), \quad m = i(\mathrm{mod}\,M), \ n = \lfloor i/M \rfloor, \\ i = 0, 1, \ldots, MN - 1. \end{aligned} \tag{5}$$

Now, using the previous equations, Eq. 1 , 2 can be expressed as the linear system

$$\boldsymbol{y} = \boldsymbol{\Phi}\boldsymbol{s}. \tag{6}$$

In the case of 2d, components of 6 can be written more analytically as

$$
\boldsymbol{y} =
\begin{bmatrix}
y[0,0] \\
y[1,0] \\
y[2,0] \\
\vdots \\
y[M-1,0] \\
y[0,1] \\
y[1,1] \\
\vdots \\
y[M-1,N-1]
\end{bmatrix}, \quad
\boldsymbol{s} =
\begin{bmatrix}
\sigma_0 \\
\sigma_1 \\
\sigma_2 \\
\vdots \\
\sigma_{N_x N_y - 1}
\end{bmatrix},
$$

$$
\boldsymbol{\Phi} =
\begin{bmatrix}
\exp\left(-j2\pi f_0 \tau_{0,0}\right) & \exp\left(-j2\pi f_0 \tau_{1,0}\right) & \cdots & \exp\left(-j2\pi f_0 \tau_{N_x N_y - 1,0}\right) \\
\exp\left(-j2\pi f_1 \tau_{0,0}\right) & \exp\left(-j2\pi f_1 \tau_{1,0}\right) & \ddots & \exp\left(-j2\pi f_1 \tau_{N_x N_y - 1,0}\right) \\
\vdots & \vdots & & \vdots \\
\exp\left(-j2\pi f_{M-1} \tau_{0,0}\right) & \exp\left(-j2\pi f_{M-1} \tau_{1,0}\right) & & \exp\left(-j2\pi f_{M-1} \tau_{N_x N_y - 1,0}\right) \\
\exp\left(-j2\pi f_0 \tau_{0,1}\right) & \exp\left(-j2\pi f_0 \tau_{1,1}\right) & & \exp\left(-j2\pi f_0 \tau_{N_x N_y - 1,1}\right) \\
\exp\left(-j2\pi f_1 \tau_{0,1}\right) & \exp\left(-j2\pi f_1 \tau_{1,1}\right) & \cdots & \exp\left(-j2\pi f_1 \tau_{N_x N_y - 1,1}\right) \\
\vdots & \vdots & & \vdots \\
\exp\left(-j2\pi f_{M-1} \tau_{0,N-1}\right) & \exp\left(-j2\pi f_{M-1} \tau_{1,N-1}\right) & & \exp\left(-j2\pi f_{M-1} \tau_{N_x N_y - 1,N-1}\right)
\end{bmatrix}.
\tag{7}
$$

In the case of only one frequency, 7 reduces to

$$
\boldsymbol{y} =
\begin{bmatrix}
y[0,0] \\
y[0,1] \\
\vdots \\
y[0,N-1]
\end{bmatrix}, \quad
\boldsymbol{s} =
\begin{bmatrix}
\sigma_0 \\
\sigma_1 \\
\sigma_2 \\
\vdots \\
\sigma_{N_x N_y - 1}
\end{bmatrix},
$$

$$
\boldsymbol{\Phi} =
\begin{bmatrix}
\exp\left(-j2\pi f_0 \tau_{0,0}\right) & \exp\left(-j2\pi f_0 \tau_{1,0}\right) & \cdots & \exp\left(-j2\pi f_0 \tau_{N_x N_y - 1,0}\right) \\
\exp\left(-j2\pi f_0 \tau_{0,1}\right) & \exp\left(-j2\pi f_0 \tau_{1,1}\right) & & \exp\left(-j2\pi f_0 \tau_{N_x N_y - 1,1}\right) \\
\vdots & \vdots & & \vdots \\
\exp\left(-j2\pi f_0 \tau_{0,N-1}\right) & \exp\left(-j2\pi f_0 \tau_{1,N-1}\right) & & \exp\left(-j2\pi f_0 \tau_{N_x N_y - 1,N-1}\right)
\end{bmatrix}.
\tag{8}
$$

Extending in 3d components, we have

$$
\boldsymbol{y} =
\begin{bmatrix}
y[0,0] \\
y[1,0] \\
y[2,0] \\
\vdots \\
y[M-1,0] \\
y[0,1] \\
y[1,1] \\
\vdots \\
y[M-1,N-1]
\end{bmatrix}, \quad
\boldsymbol{s} =
\begin{bmatrix}
\sigma_0 \\
\sigma_1 \\
\sigma_2 \\
\vdots \\
\sigma_{N_x N_y N_z - 1,}
\end{bmatrix},
$$

$$
\boldsymbol{\Phi} =
\begin{bmatrix}
\exp\left(-j2\pi f_0 \tau_{0,0}\right) & \exp\left(-j2\pi f_0 \tau_{1,0}\right) & \cdots & \exp\left(-j2\pi f_0 \tau_{N_x N_y N_z - 1,0}\right) \\
\exp\left(-j2\pi f_1 \tau_{0,0}\right) & \exp\left(-j2\pi f_1 \tau_{1,0}\right) & \ddots & \exp\left(-j2\pi f_1 \tau_{N_x N_y N_z - 1,0}\right) \\
\vdots & \vdots & & \vdots \\
\exp\left(-j2\pi f_{M-1} \tau_{0,0}\right) & \exp\left(-j2\pi f_{M-1} \tau_{1,0}\right) & & \exp\left(-j2\pi f_{M-1} \tau_{N_x N_y N_z - 1,0}\right) \\
\exp\left(-j2\pi f_0 \tau_{0,1}\right) & \exp\left(-j2\pi f_0 \tau_{1,1}\right) & & \exp\left(-j2\pi f_0 \tau_{N_x N_y N_z - 1,1}\right) \\
\exp\left(-j2\pi f_1 \tau_{0,1}\right) & \exp\left(-j2\pi f_1 \tau_{1,1}\right) & \cdots & \exp\left(-j2\pi f_1 \tau_{N_x N_y N_z - 1,1}\right) \\
\vdots & \vdots & & \vdots \\
\exp\left(-j2\pi f_{M-1} \tau_{0,N-1}\right) & \exp\left(-j2\pi f_{M-1} \tau_{1,N-1}\right) & & \exp\left(-j2\pi f_{M-1} \tau_{N_x N_y N_z - 1,N-1}\right)
\end{bmatrix}.
\tag{9}
$$

2.3 First Order Reflections

Image of a point using rotations - 2D Model

Think about the scattering scenario 1 depicted in Fig. 2. The p-th point in grid is located at $\mathbf{w}_p = [x_p, y_p]^\top$, and the interior wall (sidewall of a room) is parallel to $y-$axis and located at $x = x_w$. Multipath propagation consists of the propagation from the antenna to the grid point along the path P_1, and then from the grid point through a reflection at the interior wall along P_2. Due to the laws of reflection [5] (angle of incidence equals angle of reflection), the location of the virtual target is found by reflecting the original target at the wall. In particular, the virtual point is located at $\mathbf{w}_p' = [2x_w - x_p, y_p]^\top$ and the delay associated with the path P_2 is the same as that of the path P_3 from the virtual point to the antenna. From the receiver's perspective, the two paths are equivalent in terms of delay and angle of incidence.
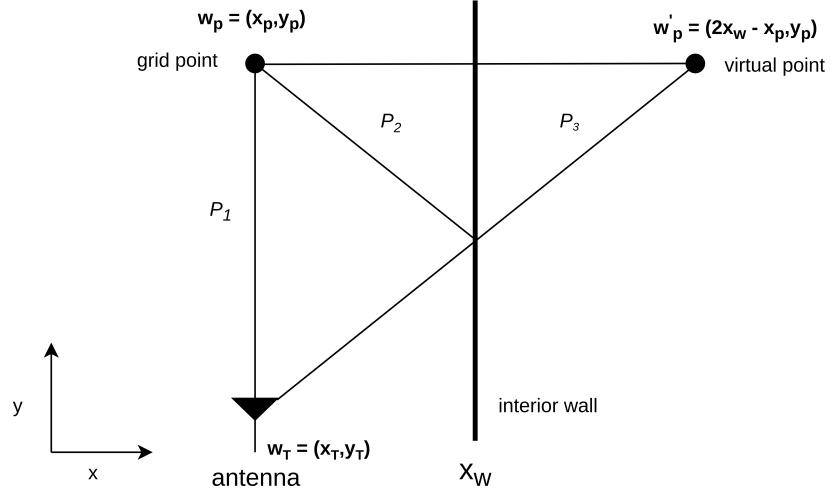
Figure 2: Multipath through reflection at an internal wall parallel to y axis.

Below, we depict the case where the wall is not parallel to $y-$axis.
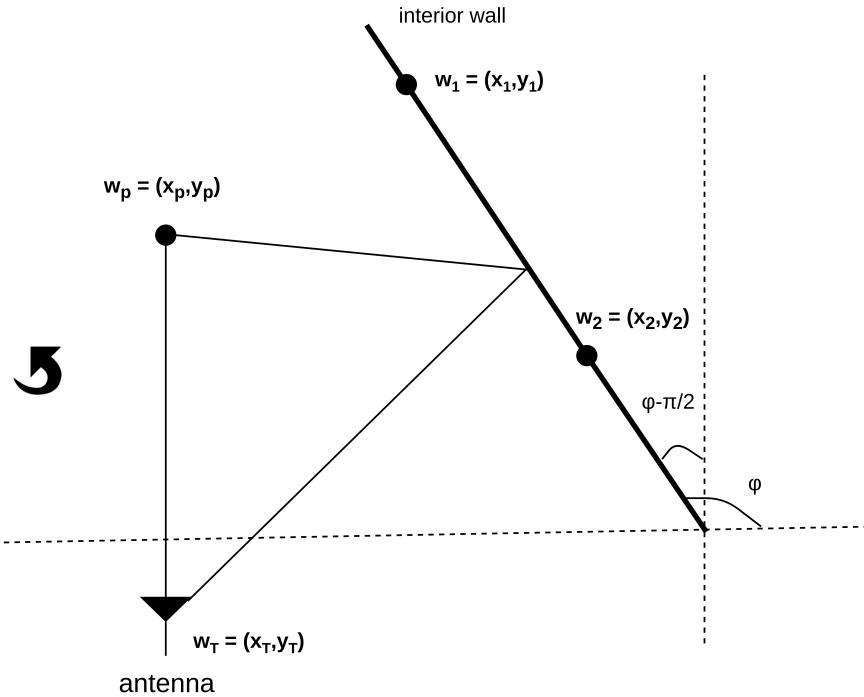
Figure 3: Multipath through reflection at an internal wall non parallel to y axis.

Consider the scattering scenario illustrated in Fig. 3. The goal is to recalculate the virtual point coordinates as we did in scenario 1. An easy way is to reduce scenario 2 to scenario 1 by rotating all of the coordinate system so that the interior wall is parallel to the y-axis. In linear algebra, a rotation matrix is a transformation matrix that is used to

perform a rotation in an Euclidean space. For example, matrix

$$\mathbf{A}(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \tag{10}$$

rotates points in the xy-plane counterclockwise, through an angle θ with respect to the x-axis. If we rotate all the points with respect to the angle $-(\phi - \pi/2)$ (counterclockwise) then the wall will be parallel to the y-axis. The angle ϕ is given by

$$\phi = \tan^{-1}\left(\frac{y_2 - y_1}{x_2 - x_1}\right), \tag{11}$$

where (x_1, y_1) and (x_2, y_2) are the coordinates of two known wall points.

Before rotation, it is important to translate all the points, according to some reference point. We select the antenna as a reference point, with coordinates $\mathbf{w}_T = [x_T, y_T]^\top$. Thus, the coordinates of the p-th grid point after translation are $\tilde{\mathbf{w}}_p = [x_p - x_T, y_p - y_T]^\top$; that also holds for the wall points $\tilde{\mathbf{w}}_1 = [x_1 - x_T, y_1 - y_T]^\top$ and $\tilde{\mathbf{w}}_2 = [x_2 - x_T, y_2 - y_T]^\top$, respectively. To calculate the angle ϕ, it makes no sense to do translation, because the angles at the Cartesian coordinate system after translation are not altered. For example, equation 12 is identical to 11.

$$\begin{aligned} \phi &= \tan^{-1}\left(\frac{y_2 - y_T - (y_1 - y_T)}{x_2 - x_T - (x_1 - x_T)}\right) \\ &= \tan^{-1}\left(\frac{y_2 - y_T - y_1 + y_T}{x_2 - x_T - x_1 + x_T}\right) \\ &= \tan^{-1}\left(\frac{y_2 - y_1}{x_2 - x_1}\right). \end{aligned} \tag{12}$$

Now, we apply rotation to the points, $\tilde{\mathbf{w}}_p, \tilde{\mathbf{w}}_1, \tilde{\mathbf{w}}_2$ as follows, assuming that $\mu = -\left(\phi - \dfrac{\pi}{2}\right)$.

$$\begin{aligned} \tilde{\mathbf{w}}'_p = \mathbf{A}(\mu)\tilde{\mathbf{w}}_p &= \begin{bmatrix} \cos(\mu) & -\sin(\mu) \\ \sin(\mu) & \cos(\mu) \end{bmatrix} \begin{bmatrix} x_p - x_T \\ y_p - y_T \end{bmatrix} \\ &= \begin{bmatrix} (x_p - x_T)\cos(\mu) - (y_p - y_T)\sin(\mu) \\ (x_p - x_T)\sin(\mu) + (y_p - y_T)\cos(\mu) \end{bmatrix} = \begin{bmatrix} x'_p \\ y'_p \end{bmatrix}, \end{aligned} \tag{13}$$

$$\begin{aligned} \tilde{\mathbf{w}}'_1 = \mathbf{A}(\mu)\tilde{\mathbf{w}}_1 &= \begin{bmatrix} \cos(\mu) & -\sin(\mu) \\ \sin(\mu) & \cos(\mu) \end{bmatrix} \begin{bmatrix} x_1 - x_T \\ y_1 - y_T \end{bmatrix} \\ &= \begin{bmatrix} (x_1 - x_T)\cos(\mu) - (y_1 - y_T)\sin(\mu) \\ (x_1 - x_T)\sin(\mu) + (y_1 - y_T)\cos(\mu) \end{bmatrix} = \begin{bmatrix} x'_1 \\ y'_1 \end{bmatrix}, \end{aligned} \tag{14}$$

$$\begin{aligned} \tilde{\mathbf{w}}'_2 = \mathbf{A}(\mu)\tilde{\mathbf{w}}_2 &= \begin{bmatrix} \cos(\mu) & -\sin(\mu) \\ \sin(\mu) & \cos(\mu) \end{bmatrix} \begin{bmatrix} x_2 - x_T \\ y_2 - y_T \end{bmatrix} \\ &= \begin{bmatrix} (x_2 - x_T)\cos(\mu) - (y_2 - y_T)\sin(\mu) \\ (x_2 - x_T)\sin(\mu) + (y_2 - y_T)\cos(\mu) \end{bmatrix} = \begin{bmatrix} x'_2 \\ y'_2 \end{bmatrix}. \end{aligned} \tag{15}$$

At this point, we are able to calculate the virtual point, on the translated and rotated plane. The virtual point of $\tilde{\mathbf{w}}'_p$ is

$$
\begin{aligned}
\tilde{\mathbf{w}}'_v &= \begin{bmatrix} 2x'_1 - x'_p \\ y'_p \end{bmatrix} \\
&= \begin{bmatrix} 2(x_1 - x_T)\cos(\mu) - 2(y_1 - y_T)\sin(\mu) - (x_p - x_T)\cos(\mu) + (y_p - y_T)\sin(\mu) \\ (x_p - x_T)\sin(\mu) + (y_p - y_T)\cos(\mu) \end{bmatrix}.
\end{aligned}
\tag{16}
$$

Once we have found the virtual point, we need to return it to the original coordinate system, applying the reverse process (Rotation - Translation). Hence, the virtual point in the original coordinate system is

$$
\mathbf{w}_v = \mathbf{A}(-\mu)\tilde{\mathbf{w}}'_v + \mathbf{w}_T.
\tag{17}
$$

Below, two examples are presented for finding the virtual point with the proposed method.
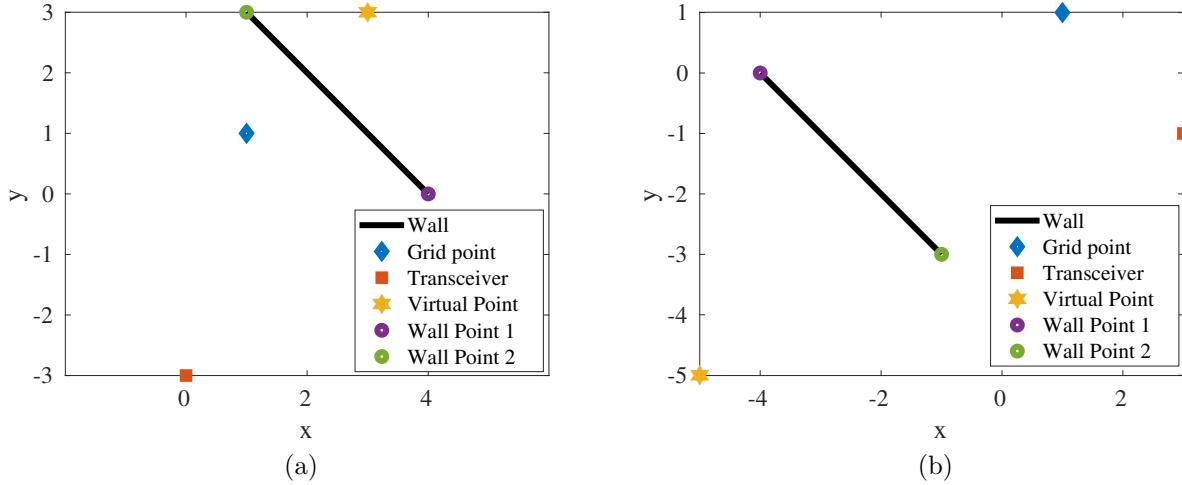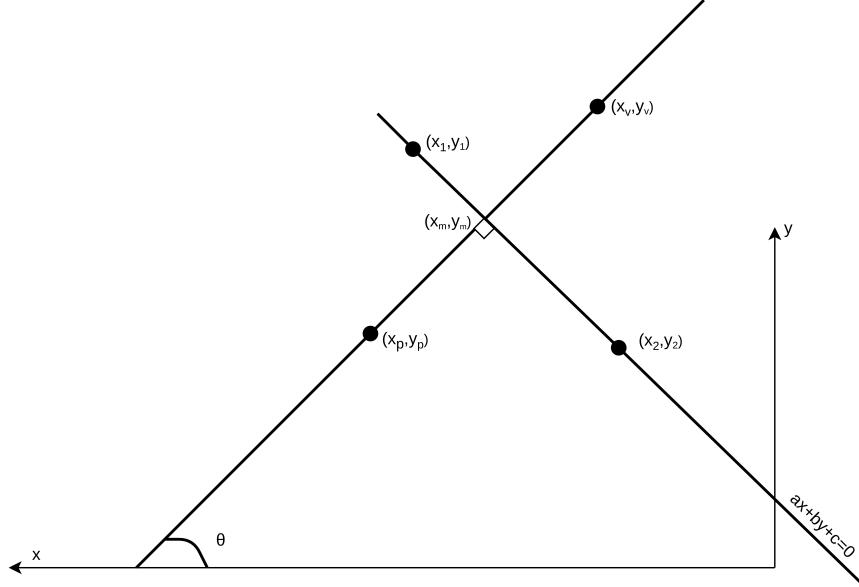
Figure 4: Two examples of finding a virtual point using the rotation method (2d).

Image of a point with respect to a line - 2D Model

For the sake of completeness, we re-calculated the virtual point for the general case with a classical geometric method to see whether its results coincide with the ones of the rotation method. Therefore, let two wall points $w_1 = (x_1, y_1)$, $w_2 = (x_2, y_2)$ and a p-th grid point $w_p = (x_p, y_p)$.

16

The equation of the line defined by the two points w_1 and w_2 is

$$y - y_1 = \lambda (x - x_1), \tag{18}$$

where λ is the slope of the line and is calculated as follows

$$\lambda = \frac{y_2 - y_1}{x_2 - x_1}. \tag{19}$$

Firstly, we convert 18 into the form of $ax + by + c = 0$,

$$\underbrace{-\lambda}_{a} x + \underbrace{1}_{b} y + \underbrace{\lambda x_1 - y_1}_{c} = 0. \tag{20}$$

Hence, $a = -\lambda$, $b = 1$ and $c = \lambda x_1 - y_1$. Also, the image of point w_p in the line mirror $ax + by + c = 0$ is the point $w_v = (x_v, y_v)$ such that $w_p w_v$ line segment is perpendicular to $ax + by + c = 0$ and the mid-point of the line segment $w_m = \left(\frac{x_p + x_v}{2}, \frac{y_p + y_v}{2}\right)$ lies on the aforementioned line. Moreover, the slope of the line $ax + by + c = 0$ is $-\frac{a}{b}$. Thus, the slope of the line segment $w_p w_v$ is $\frac{b}{a}$. Therefore, if $w_p w_v$ makes an angle θ with x-axis, then it holds that

$$\tan(\theta) = \frac{b}{a} \Rightarrow (i)\ \sin(\theta) = \frac{b}{\sqrt{a^2 + b^2}} \text{ and } (ii)\ \cos(\theta) = \frac{a}{\sqrt{a^2 + b^2}}. \tag{21}$$

17

Proof. Firstly, we express $\cos(\theta)$ as a function of $\sin(\theta)$ and vice versa

$$\tan(\theta) = \frac{\sin(\theta)}{\cos(\theta)}$$

$$\Leftrightarrow \quad \frac{\sin(\theta)}{\cos(\theta)} = \frac{b}{a}$$

$$\Leftrightarrow \quad \sin(\theta) = \frac{b}{a}\cos(\theta)$$

$$\Leftrightarrow \quad \cos(\theta) = \frac{a}{b}\sin(\theta). \tag{22}$$

Let the trigonometric identity

$$\sin^2(\theta) + \cos^2(\theta) = 1. \tag{23}$$

By combining 22 and 23, we check if our claim (i) is true

$$\sin^2(\theta) + \cos^2(\theta) = 1 \Leftrightarrow$$

$$\sin^2(\theta) + \left(\frac{a}{b}\sin(\theta)\right)^2 = 1 \Leftrightarrow$$

$$\sin^2(\theta) + \frac{a^2}{b^2}\sin^2(\theta) = 1 \Leftrightarrow$$

$$\sin^2(\theta)\left(1 + \frac{a^2}{b^2}\right) = 1 \Leftrightarrow \tag{24}$$

$$\sin^2(\theta)\left(\frac{a^2 + b^2}{b^2}\right) = 1 \Leftrightarrow$$

$$\sin^2(\theta) = \left(\frac{b^2}{a^2 + b^2}\right).$$

Note that the line segment $w_p w_v$, in all scenarios, creates an angle with the x-axis in the range $[0, \pi]$ (angle in radians), thus $0 \le \theta \le \pi$. This implies that $0 \le \sin(\theta) \le 1$. Therefore,

$$\sin(\theta) = \left(\frac{b}{\sqrt{a^2 + b^2}}\right). \tag{25}$$

Similarly, from 22 and 25 we get

$$\cos(\theta) = \frac{a}{b}\left(\frac{b}{\sqrt{a^2 + b^2}}\right) = \left(\frac{a}{\sqrt{a^2 + b^2}}\right). \tag{26}$$

The equation of line $w_p w_v$ in distance form (parametric form) is

$$\frac{x - x_p}{\cos(\theta)} = \frac{y - y_p}{\sin(\theta)}. \tag{27}$$

18

Let r be the length of the line segment $w_p w_m$. Then, the coordinates of w_m are given by

$$\frac{x - x_p}{\cos(\theta)} = \frac{y - y_p}{\sin(\theta)} = r. \tag{28}$$

Hence, the coordinates of w_m are $(x_p + r\cos(\theta), y_p + r\sin(\theta))$. Since, w_m lies on $ax + by + c = 0$, it holds that

$$a(x_p + r\cos(\theta)) + b(y_p + r\sin(\theta)) + c = 0$$
$$\Rightarrow \quad r = -\frac{ax_p + by_p + c}{a\cos(\theta) + b\sin(\theta)} \tag{29}$$
$$\Rightarrow \quad r = -\frac{ax_p + by_p + c}{\sqrt{a^2 + b^2}}.$$

The coordinates of w_v are given by

$$\begin{cases} \dfrac{x - x_p}{\cos(\theta)} = \dfrac{y - y_p}{\sin(\theta)} = 2r \\ \dfrac{x - x_p}{a} = \dfrac{y - y_p}{b} = -2\dfrac{ax_p + by_p + c}{\sqrt{a^2 + b^2}} \end{cases} \Rightarrow \begin{cases} x_v = -\dfrac{2a(ax_p + by_p + c)}{a^2 + b^2} + x_p \\ y_v = -\dfrac{2b(ax_p + by_p + c)}{a^2 + b^2} + y_p. \end{cases} \tag{30}$$

$\blacksquare$

Below, two examples are listed for finding the virtual point with the aforementioned method.
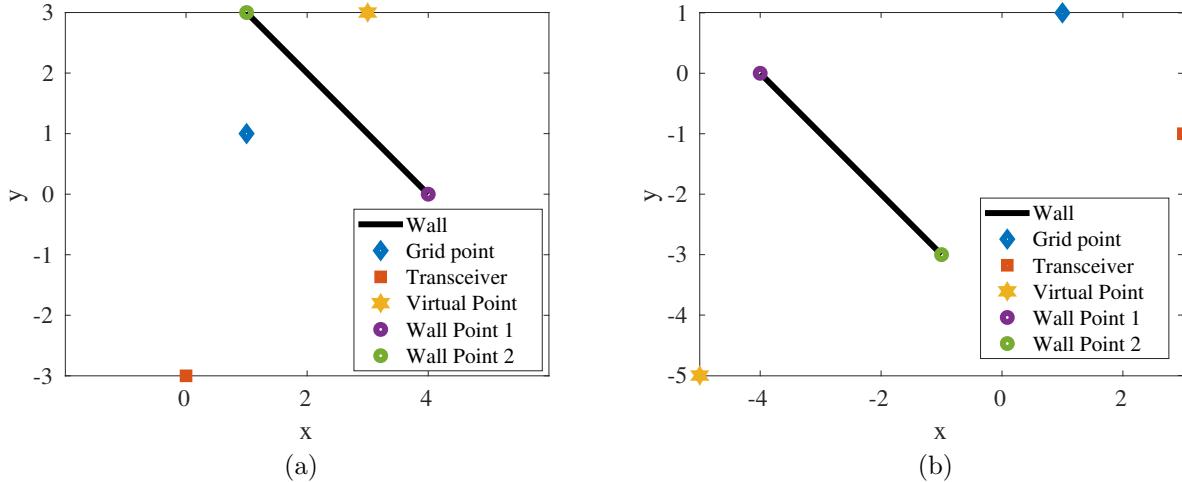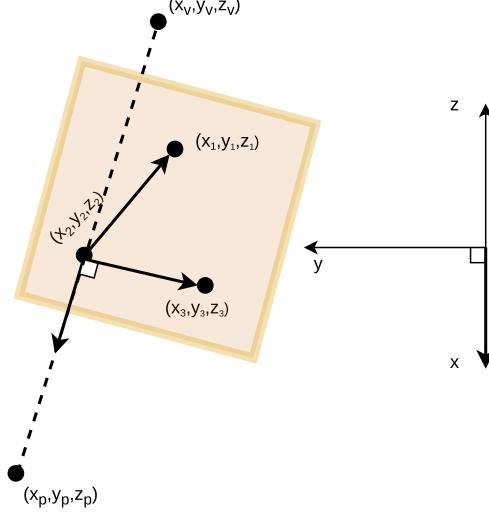
Figure 5: Two examples of finding a virtual point using the method that takes advantage of angles and perpendicularity between the wall and the straight section that joins the target to its image (2d).

Figures 4 and 5 represent the same scenarios, depicting the same virtual point. Therefore the two methods are equivalent.

3D Model

It is important for us to generalize the finding of the virtual point in three dimensions so that we can describe the delays in more realistic conditions. Thus, let three wall points $\mathbf{w}_1 = [x_1, y_1, z_1]^\top, \mathbf{w}_2 = [x_2, y_2, z_2]^\top, \mathbf{w}_3 = [x_3, y_3, z_3]^\top$ and a p-th grid point $\mathbf{w}_p = [x_p, y_p, z_p]^\top$.

Assume two vectors, which define a plane in 3d,

$$\begin{aligned}
\mathbf{v}_1 &= [x_2 - x_1, y_2 - y_1, z_2 - z_1]^\top, \\
\mathbf{v}_2 &= [x_3 - x_1, y_3 - y_1, z_3 - z_1]^\top.
\end{aligned} \tag{31}$$

Our goal is to find a normal vector perpendicular to the plane defined by these vectors. In view of that, we proceed with the calculation of their cross product, utilizing the following "determinant" formula

$$\begin{aligned}
\mathbf{n} = \mathbf{v}_1 \times \mathbf{v}_2 &= \begin{vmatrix} \hat{\mathbf{i}} & \hat{\mathbf{j}} & \hat{\mathbf{k}} \\ \mathbf{v}_{1i} & \mathbf{v}_{1j} & \mathbf{v}_{1k} \\ \mathbf{v}_{2i} & \mathbf{v}_{2j} & \mathbf{v}_{2k} \end{vmatrix} \\
&= \begin{vmatrix} \mathbf{v}_{1j} & \mathbf{v}_{1k} \\ \mathbf{v}_{2j} & \mathbf{v}_{2k} \end{vmatrix} \hat{\mathbf{i}} - \begin{vmatrix} \mathbf{v}_{1i} & \mathbf{v}_{1k} \\ \mathbf{v}_{2i} & \mathbf{v}_{2k} \end{vmatrix} \hat{\mathbf{j}} + \begin{vmatrix} \mathbf{v}_{1i} & \mathbf{v}_{1j} \\ \mathbf{v}_{2i} & \mathbf{v}_{2j} \end{vmatrix} \hat{\mathbf{k}} \\
&= \left(\mathbf{v}_{1j}\mathbf{v}_{2k} - \mathbf{v}_{1k}\mathbf{v}_{2j} \right) \hat{\mathbf{i}} - \left(\mathbf{v}_{1i}\mathbf{v}_{2k} - \mathbf{v}_{1k}\mathbf{v}_{2i} \right) \hat{\mathbf{j}} + \left(\mathbf{v}_{1i}\mathbf{v}_{2j} - \mathbf{v}_{1j}\mathbf{v}_{2i} \right) \hat{\mathbf{k}} \\
&= \left(\mathbf{v}_{1j}\mathbf{v}_{2k} - \mathbf{v}_{1k}\mathbf{v}_{2j} \right) \hat{\mathbf{i}} + \left(\mathbf{v}_{1k}\mathbf{v}_{2i} - \mathbf{v}_{1i}\mathbf{v}_{2k} \right) \hat{\mathbf{j}} + \left(\mathbf{v}_{1i}\mathbf{v}_{2j} - \mathbf{v}_{1j}\mathbf{v}_{2i} \right) \hat{\mathbf{k}}.
\end{aligned} \tag{32}$$

Thus, the normal vector is

$$\mathbf{n} = \langle \alpha, \beta, \gamma \rangle, \tag{33}$$

where

$$\begin{aligned}
\alpha &= (y_2 - y_1)(z_3 - z_1) - (y_3 - y_1)(z_2 - z_1), \\
\beta &= (x_3 - x_1)(z_2 - z_1) - (x_2 - x_1)(z_3 - z_1), \\
\gamma &= (x_2 - x_1)(y_3 - y_1) - (y_2 - y_1)(y_3 - y_1).
\end{aligned}$$

The plane equation defined by the aforementioned normal vector and is satisfied by the point $[x, y, z]$ is given by

$$(x - x_1)\alpha + (y - y_1)\beta + (z - z_1)\gamma = 0. \tag{34}$$

We can express the coordinates of the point $[x, y, z]$ that satisfies 34 in a parametric form

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} x_p \\ y_p \\ z_p \end{bmatrix} + t \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix} = \begin{bmatrix} x_p + \alpha t \\ y_p + \beta t \\ z_p + \gamma t \end{bmatrix}. \tag{35}$$

By substituting the coordinates $[x, y, z]$ to the plane equation, we get

$$\begin{aligned} (x - x_1)\alpha + (y - y_1)\beta + (z - z_1)\gamma = 0 &\Leftrightarrow \\ \alpha x + \beta y + \gamma z = \alpha x_1 + \beta y_1 + \gamma z_1 &\Leftrightarrow \\ \alpha(x_p + \alpha t) + \beta(y_p + \beta t) + \gamma(z_p + \gamma t) = \alpha x_1 + \beta y_1 + \gamma z_1 &\Leftrightarrow \\ \alpha x_p + \alpha^2 t + \beta y_p + \beta^2 t + \gamma z_p + \gamma^2 t = \alpha x_1 + b\beta y_1 + \gamma z_1 &\Leftrightarrow \\ t = \frac{\alpha(x_1 - x_p) + \beta(y_1 - y_p) + \gamma(z_1 - z_p)}{\alpha^2 + \beta^2 + \gamma^2}. \end{aligned} \tag{36}$$

To find the virtual point, we have to move in the direction of the normal vector perpendicular to the plane by a distance of $2t$. Hence, the coordinates of the virtual point reflected by the plane are

$$\begin{bmatrix} x_v \\ y_v \\ z_v \end{bmatrix} = \begin{bmatrix} x_p \\ y_p \\ z_p \end{bmatrix} + 2t \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix} = \begin{bmatrix} x_p + 2\alpha t \\ y_p + 2\beta t \\ z_p + 2\gamma t \end{bmatrix}. \tag{37}$$

Below, two examples for finding the virtual point are depicted, using the aforementioned cross-product method.
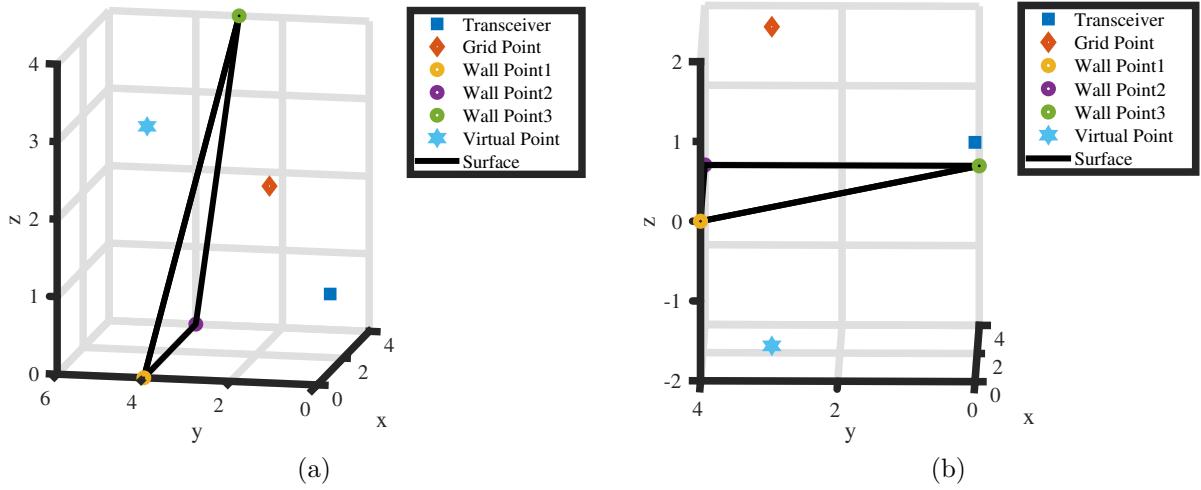
Figure 6: Two examples of finding a virtual point using the cross-product method considering 3 known points that belong to a surface. (a) The surface represents a wall in front of the target, (b) the surface represents the floor in a room (3d).

2.4   Multipath Mitigation

In our application, multipath propagation corresponds to indirect paths which involve reflections at one or more interior walls, by which the signal may reach the target. The dominant multipath component corresponds to a "bistatic" scattering scenario. Namely, the transmitted signal propagates directly to a scatterer and the scattered wave travels back to the transceiver after being reflected by an interior wall, or the opposite.

The observations that are made in the previous paragraph concerning the multipath have to be integrated into the model 6 . Let a maximum of $K-1$ multipath propagation paths and one direct path. For each considered propagation path k, a separate target reflectivity vector $\boldsymbol{s}^{(k)}$ is introduced. However, it is assumed that the target reflectivity corresponding to each path is constant across the array elements. Therefore, 6 can be modified to

$$
\boldsymbol{y} = \sum_{k=0}^{K-1} \boldsymbol{\Phi}^{(k)} \boldsymbol{s}^{(k)} = \boldsymbol{\Phi}^{(0)} \boldsymbol{s}^{(0)} + \boldsymbol{\Phi}^{(1)} \boldsymbol{s}^{(1)} + \cdots + \boldsymbol{\Phi}^{(K-1)} \boldsymbol{s}^{(K-1)}.
\tag{38}
$$

The received multipath components are expressed as the summation of the target reflectivity vectors $\boldsymbol{s}^{(k)}$ multiplied by the respective matrix $\boldsymbol{\Phi}^{(k)}$. The matrices include the modified delays as described in 2.3, resulting in phase terms analogous to 5.

2.5   Group Sparse Reconstruction with known walls

The backscattered signal is distorted differently when it is reflected directly from the target compared to when it is reflected from the target through an interior wall. Hence,

22

we cannot assume prior knowledge of the exact relationship between the various target reflectivity vectors $\boldsymbol{s}^{(k)}$. However, we know that all the sub-vectors are representations of the same ground-truth. If the i-th element of the sub-vector $\boldsymbol{s}^{(l)}$ is non-zero, this should also be the case for the i-th element of the vector $\boldsymbol{s}^{(m)}$, $\{l \neq m \mid l, m \in \{0, \ldots, K-1\}\}$. This observation calls for a group sparse or structured sparse reconstruction, which is described below.

All unknown vectors in 38 can be stacked to form a tall vector

$$\tilde{\boldsymbol{s}} = \left[\left(\boldsymbol{s}^{(0)}\right)^{\top} \quad \left(\boldsymbol{s}^{(1)}\right)^{\top} \quad \ldots \quad \left(\boldsymbol{s}^{(K-1)}\right)^{\top} \right]^{\top}. \tag{39}$$

The measurement vector can then be expressed as

$$\boldsymbol{y} = \tilde{\boldsymbol{\Phi}}\tilde{\boldsymbol{s}}, \quad \tilde{\boldsymbol{s}} \in \mathbb{C}^{N_x N_y K \times 1}, \tag{40}$$

where the combined measurement matrix is obtained by concatenating the individual matrices into one fat matrix

$$\tilde{\boldsymbol{\Phi}} = \begin{bmatrix} \boldsymbol{\Phi}^{(0)} & \boldsymbol{\Phi}^{(1)} & \ldots & \boldsymbol{\Phi}^{(K-1)} \end{bmatrix}, \quad \tilde{\boldsymbol{\Phi}} \in \mathbb{C}^{N \times N_x N_y K}. \tag{41}$$

We proceed to reconstruct the unknown stacked reflectivity vector $\tilde{\boldsymbol{s}}$ from the data record $\boldsymbol{y}$ under model 40. It has been shown that a group sparse reconstruction can be obtained by a mixed $\ell_1 - \ell_2$ norm [6],[7]

$$\hat{\tilde{\boldsymbol{s}}} = \arg\min_{\tilde{s}} \frac{1}{2}\|\boldsymbol{y} - \tilde{\boldsymbol{\Phi}}\tilde{\boldsymbol{s}}\|_2^2 + \lambda\|\tilde{\boldsymbol{s}}\|_{2,1}, \tag{42}$$

where

$$\|\tilde{\boldsymbol{s}}\|_{2,1} := \sum_{p=0}^{N_x N_y - 1} \left\| \left[s_p^{(0)}, s_p^{(1)}, \ldots, s_p^{(K-1)} \right]^{\top} \right\|_2 = \sum_{p=0}^{N_x N_y - 1} \sqrt{\sum_{k=0}^{K-1} s_p^{(k)} \left(s_p^{(k)} \right)^*}, \quad \text{2d model,}$$

$$\|\tilde{\boldsymbol{s}}\|_{2,1} := \sum_{p=0}^{N_x N_y N_z - 1} \left\| \left[s_p^{(0)}, s_p^{(1)}, \ldots, s_p^{(K-1)} \right]^{\top} \right\|_2 = \sum_{p=0}^{N_x N_y N_z - 1} \sqrt{\sum_{k=0}^{K-1} s_p^{(k)} \left(s_p^{(k)} \right)^*}, \quad \text{3d model,}$$

$$\tag{43}$$

and $\lambda > 0$ is the so-called regularization parameter. According to [6] it is reasonable to choose the regularization parameter as

$$\lambda = \zeta \left\| \tilde{\boldsymbol{\Phi}}^{\dagger}\boldsymbol{y} \right\|_{\infty}, \quad \zeta \in (0, 1). \tag{44}$$

Moreover, it can be shown that if $\lambda \geq \|\tilde{\boldsymbol{\Phi}}^{\dagger}\boldsymbol{y}\|_{\infty}$, then the unique solution is the zero vector [8],[9]. A value of λ such that $\lambda \lesssim \|\tilde{\boldsymbol{\Phi}}^{\dagger}\boldsymbol{y}\|_{\infty}$ can be considered "large", while a value such

---

For the 3d model holds that $\tilde{\boldsymbol{s}} \in \mathbb{C}^{N_x N_y N_z K \times 1}$ and $\tilde{\boldsymbol{\Phi}} \in \mathbb{C}^{N \times N_x N_y N_z K}$, respectively.

that $\lambda \ll \|\tilde{\mathbf{\Phi}}^{\dagger}\boldsymbol{y}\|_{\infty}$ can be seen as small. Problem 42 can be solved using SpaRSA [6] or YALL1-Group [7], by imposing a group-sparse constraint on the solution.

Once a solution $\hat{\tilde{\boldsymbol{s}}}$ is obtained, the sub-vectors can be combined to form an overall vector with a number of elements equal to the number of points on the grid. In the absence of any knowledge of the phase of the individual reflectivities $s_p^{(k)}$, a reasonable choice is a non-coherent combination. Hence, we obtain the final vector by forming the Euclidean norm over each group. The elements of the composite vector $\hat{\boldsymbol{s}}$ are defined by

$$[\hat{\boldsymbol{s}}]_p = \left\| \left[\hat{s}_p^{(0)}, \hat{s}_p^{(1)}, \ldots, \hat{s}_p^{(K-1)} \right]^{\top} \right\|_2, \quad p = 0, \ldots, N_x N_y - 1^2. \tag{45}$$

The non-coherent combination of the sub-vectors does not provide SNR gain, as the spatially white noise will also be accumulated. However, we are able, to enhance the signal-to-clutter ratio. By employing the aforementioned scheme 45, we ensure that the power scattered by the target, received through the direct path and multipath, is combined at the true location. Little or no power is added to the remaining clutter.

Note that when $K = 1$ then problem 42 reduces to the classic complex LASSO problem

$$\begin{aligned} \hat{\tilde{\boldsymbol{s}}} &= \arg\min_{\tilde{s}} \frac{1}{2}\|\boldsymbol{y} - \tilde{\mathbf{\Phi}}\tilde{\boldsymbol{s}}\|_2^2 + \lambda\|\tilde{\boldsymbol{s}}\|_1 \\ &= \arg\min_{\boldsymbol{s}^{(0)}} \frac{1}{2}\|\boldsymbol{y} - \mathbf{\Phi}^{(0)}\boldsymbol{s}^{(0)}\|_2^2 + \lambda\|\boldsymbol{s}^{(0)}\|_1, \end{aligned} \tag{46}$$

which can also be solved by SpaRSA or YALL1-Basic. Notice that SpaRSA uses a sequence of sub-problems that can be solved efficiently and converges to the overall optimum solution. YALL1-Group is based on an ADM, splitting the problem into two alternatingly solved subproblems. Both methods can be applied to large-scale problems. However, only in [6] a proof of convergence for this type of problems is given.

2.6   Sufficient Conditions

Intuitively, specific assumptions for a sensing matrix $\tilde{\mathbf{\Phi}}$ must hold such that a signal can be reconstructed. If an important part of the signal lies in the null space of the matrix, there is no way to reconstructing it. Several conditions and properties for the sensing matrix have been presented such as the RIP [10],[11], mutual coherence [12],[13],[14], NSP [12] and UUP [15]. The aforementioned conditions are able to provide a unique reconstruction. Below, only RIP, mutual coherence and block-coherence are presented.

---

[2]For the 3d model p takes values in $\{0, \ldots, N_x N_y N_z - 1\}$.

2.6.1 Mutual Coherence

For a given sensing matrix $\tilde{\boldsymbol{\Phi}} \in \mathbb{C}^{N \times P}$, the coherence is defined in [13] as

$$\mu(\tilde{\boldsymbol{\Phi}}) = \max_{i,j \neq i} \frac{\left| \tilde{\phi}_i^\dagger \tilde{\phi}_j \right|}{\left\| \tilde{\phi}_i \right\|_2 \left\| \tilde{\phi}_j \right\|_2}, \tag{47}$$

where $\tilde{\phi}_i$ denotes the i-th column of $\tilde{\boldsymbol{\Phi}}$. Hence, the coherence of a matrix is the maximum correlation between any two of its columns. The mutual coherence should be as low as possible to obtain a faithful reconstruction. In other words, if any two columns are too similar to each other, their effect on the vector $\boldsymbol{y}$ cannot be discerned leading to reconstruction failure.

In the case of group sparse signals, the group pattern should be taken into consideration. Therefore, the definition of mutual coherence has been extended to the block-coherence [16]. The block-coherence is a measure of the correlation between blocks of columns in $\tilde{\boldsymbol{\Phi}}$, rather than individual columns as before. For simplicity, it is assumed that the groups occur in blocks of size d in the sparse vector. The block-coherence of a matrix $\tilde{\boldsymbol{\Phi}}$ is defined in [16] as

$$\mu_{\mathrm{B}}(\tilde{\boldsymbol{\Phi}}) = \max_{i,j \neq i} \frac{1}{d} \rho \left(\boldsymbol{\Phi}^{(i)\dagger} \boldsymbol{\Phi}^{(j)} \right),$$
$$\tilde{\boldsymbol{\Phi}} = \left[\underbrace{\tilde{\phi}_1 \ldots \tilde{\phi}_d}_{\boldsymbol{\Phi}^{(0)}} \quad \underbrace{\tilde{\phi}_{d+1} \ldots \tilde{\phi}_{2d}}_{\boldsymbol{\Phi}^{(1)}} \quad \cdots \quad \underbrace{\tilde{\phi}_{P-d+1} \ldots \tilde{\phi}_P}_{\boldsymbol{\Phi}^{(K-1)}} \right], \tag{48}$$

where $\rho(\cdot)$ is the spectral norm and $\boldsymbol{\Phi}^{(i)}$ is the normalized sub-matrix of $\tilde{\boldsymbol{\Phi}}$ that contains all columns belonging to the i-th block. Normalization of a matrix means that all columns are scaled to have unit Euclidean norm. Note that for $d = 1$ the block-coherence is equal to the regular coherence, furthermore in can be proved that $0 \leq \mu_{\mathrm{B}}(\tilde{\boldsymbol{\Phi}}) \leq 1$ [17].

The advantage of mutual coherence and block-coherence is the ease of computation. For a given matrix, 47 or 48 can be calculated numerically where the cost is linear in the number of columns or blocks, respectively. Hence, performance bounds for a practical system can be found. However, the theoretical results in terms of the RIP are more universal than for the coherence.

To examine the effect of different carrier frequencies on $\tilde{\boldsymbol{\Phi}}$, we constructed a toy example where the signal phase depends on the topology (such as in the section 4.1), but the $\boldsymbol{s}^{(k)}, k = 0, \ldots, K-1$ sub-vectors are constructed to have the same support. We assume a 2d environment where we put a target at location $(x_{\mathrm{tag}}, y_{\mathrm{tag}}) = (1.5, 2.7)$ and also we consider that there is a parallel wall at known location $y_w = 4$; hence $K = 2$ propagation paths. The transceivers $(N = 100)$ are arranged in a line array with zero $y-$coordinates and $x-$coordinates that extend from -0.95 to 0.9310, with element spacing 0.019. A cell array of $N_x \times N_y$ points is created, which represents the grid search space. Let j be the

---

Consider that for the 2 dimensional model it is valid that $P = KN_xN_y$, while for the 3 dimensional model $P = KN_xN_yN_z$, where K is defined as the number of propagation paths.

index of the target in the grid search space. In order to create the reflectivities of the simulated signal, we construct the two zero vectors $\boldsymbol{s}^{(0)}$ (for the direct path) and $\boldsymbol{s}^{(1)}$ (for the indirect path) and we replace the $j-$th element with the values 1 and 0.6, respectively. We suppose that the signal has no noise in order to examine the effect of frequencies on the matrix $\tilde{\boldsymbol{\Phi}}$. Therefore, our data vector $\boldsymbol{y}$ is constructed by the multiplication of the known stacked vector $\tilde{\boldsymbol{s}} = \left[\left(\boldsymbol{s}^{(0)} \right)^{\top} \left(\boldsymbol{s}^{(1)} \right)^{\top} \right]^{\top}$ and $\tilde{\boldsymbol{\Phi}}$. Below, Fig. 7 is depicted, which shows how the block-coherence metric behaves at different carrier frequencies without the presence of noise.
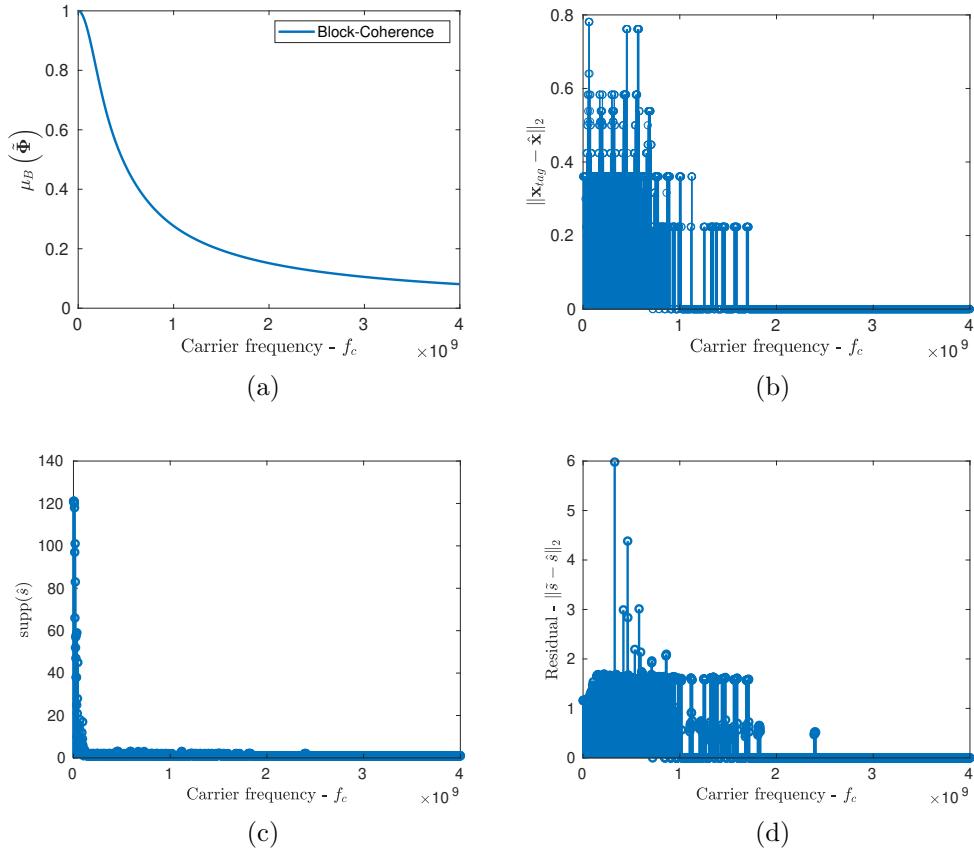
Figure 7: Theoretical evaluation considering different metrics. (a) Contains the block coherence of the matrix $\tilde{\boldsymbol{\Phi}}$ vs various carrier frequencies, (b) represents the target estimation error by changing the carrier frequency, (c) captures the support of the estimated composite vector $\hat{\boldsymbol{s}}$ for different frequencies, and (d) depicts the residual defined as the euclidean norm between the estimated composite vector and the true composite vector for each carrier frequency.

It is easy to observe that at carrier frequencies greater than 1.8 GHz, the method accurately estimates the target. The efficiency of the method in this frequency range seems to

26

be well explained by the fact that for high frequencies the block-coherence of matrix $\tilde{\boldsymbol{\Phi}}$ is quite small.

Equally important is whether the distance between the antennas from measurement to measurement affects the matrix $\tilde{\boldsymbol{\Phi}}$. Thus, we experimented by setting the carrier frequency to $f_c = 868$ MHz but changing the distance between adjacent antennas.

By referring to Fig. 8, we observe that for distances between neighboring antennas greater than 5 cm the efficiency of the method is the best possible. Again, we see that this is achieved when the block-coherence is small enough. This raises the necessity for a bound which will link the efficiency of the method with the carrier frequency and the block-coherence of the matrix $\tilde{\boldsymbol{\Phi}}$. Namely, we are looking for some inequality, which, when satisfied, can safely guarantee the proper reconstruction of the vector $\tilde{\boldsymbol{s}}$. However, this is a rather challenging part that is assigned to future work and which is capable of opening new horizons in the connection between narrowband systems and RFID tags.
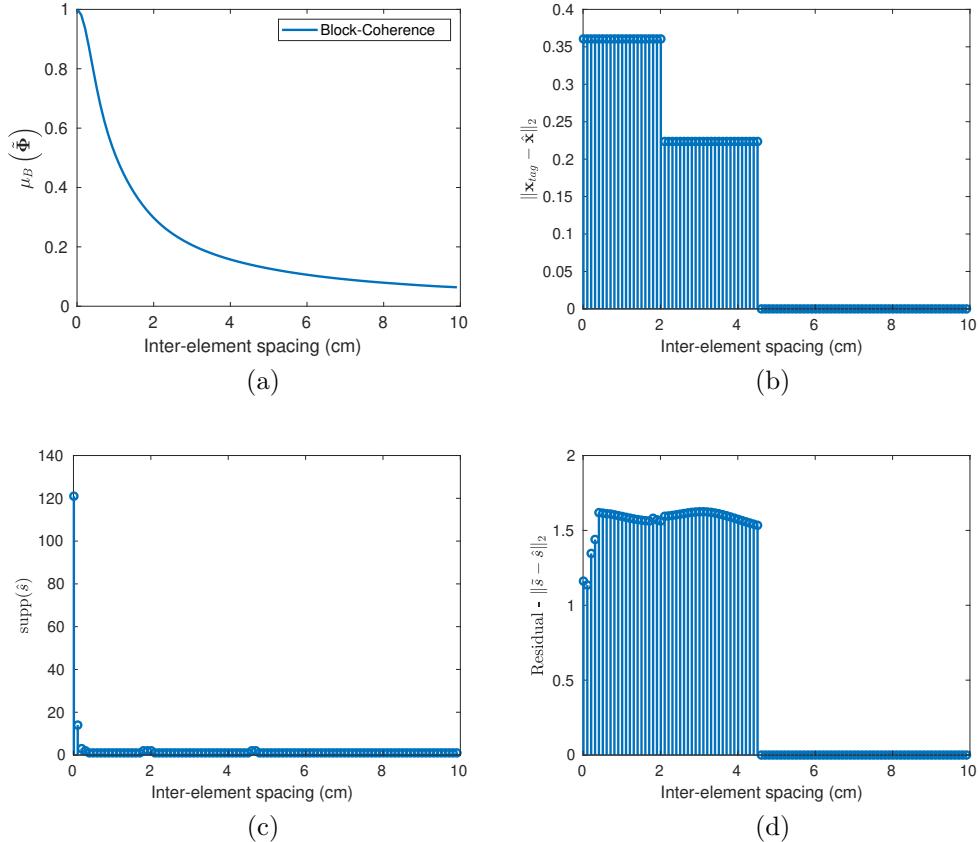
Figure 8: Theoretical evaluation considering different metrics. Note that (a), (b), (c), (d) are exactly the same as Fig. 7 with the only difference that what varies concerning x-axis is the inter-element spacing given the carrier frequency $f_c = 868$ MHz.

2.6.2 Restricted Isometry Property

The RIP for matrix $\tilde{\boldsymbol{\Phi}} \in \mathbb{C}^{N \times P}$ is defined with respect to a given sparsity level ϵ and isometry constant $\delta_\epsilon > 0$. If $\tilde{\boldsymbol{\Phi}}$ obeys the RIP, the inequality

$$(1 - \delta_\epsilon) \, \|\tilde{\boldsymbol{s}}\|_2^2 \leq \|\tilde{\boldsymbol{\Phi}}\tilde{\boldsymbol{s}}\|_2^2 \leq (1 + \delta_\epsilon) \, \|\tilde{\boldsymbol{s}}\|_2^2 \tag{49}$$

holds for any ϵ-sparse vector $\tilde{\boldsymbol{s}}$ [10],[11]. For instance, consider that the RIP holds with small isometry constants. That indicates that the Euclidean norm of any ϵ-sparse vector is merely altered by the small constant $1 \pm \delta_\epsilon$ when multiplied with $\tilde{\boldsymbol{\Phi}}$. Thus, the energy of the signal is maintained. Alternatively, any set of ϵ columns in $\tilde{\boldsymbol{\Phi}}$ is approximately linearly independent. Hence, for a ϵ-sparse signal, the matrix behaves like an orthonormal matrix.

2.7 Compressive Sensing Localization Algorithm

Based on all the above, the algorithmic steps for 2d localization, are described in 1. A similar procedure is followed for positioning a target in the 3d space. We introduce also a method for the division of the initial grid search space. It makes sense to do this to reduce computational complexity. More specifically, we can start with a fairly large grid, but relatively sparse; then, find a solution to it, and immediately develop a new smaller but denser grid around the previous solution. Thus, we will gradually come up with a better solution.

Algorithm 1: CSLA - 2D (Compressive Sensing Localization Algorithm)

    Result: Estimated Coordinates $\hat{x}, \hat{y}$

1 Initialize tolerance between $10^{-2}, 10^{-8}$, $\zeta \in (0,1)$ and T ($T \in \mathbb{N}$ indicates how many times we want the grid to be divided, if $T = 1$, then the grid will never be divided)

2 Collect measurements in a vector $\boldsymbol{y}$

3 Select x_{start}, x_{end}, y_{start} and y_{end} points

4 Define the grid resolution d_x and d_y

5 $grid\_coords := \{\}$

6 $[grid\_coords] = \text{Create\_grid}(x_{\text{start}}, x_{\text{end}}, y_{\text{start}}, y_{\text{end}}, d_x, d_y)$

7 **for** $t = 1 : T$ **do**

8 $\tilde{\boldsymbol{\Phi}} := []$

9   **for** $k = 0 : K - 1$ **do**

10 $\tilde{\boldsymbol{\Phi}} = \begin{bmatrix} \tilde{\boldsymbol{\Phi}} & \boldsymbol{\Phi}^{(k)} \end{bmatrix}$

11   **end**

12 $\lambda = \zeta \left\| \tilde{\boldsymbol{\Phi}}^{\dagger} \boldsymbol{y} \right\|_{\infty}$, $\zeta \in (0,1)$

13 $[\tilde{\boldsymbol{s}}] = \text{SpaRSA}(\boldsymbol{y}, \tilde{\boldsymbol{\Phi}}, \lambda)$

14 $[\hat{\boldsymbol{s}}]_p = \left\| \begin{bmatrix} \hat{s}_p^{(0)}, \hat{s}_p^{(1)}, \ldots, \hat{s}_p^{(K-1)} \end{bmatrix}^{\top} \right\|_2$

15 $[\sim, idx] = \max(\hat{\boldsymbol{s}})$

16 $[\hat{x}, \hat{y}] = grid\_coords\{idx\}$

17 $[x_{\text{start}}, y_{\text{start}}, x_{\text{end}}, y_{\text{end}}, d_x, d_y, \text{dist}] = \text{Shrink\_Grid}\{\text{dist}, d_x, d_y, \hat{x}, \hat{y}\}$

18 **end**

19 **return** $\hat{x}, \hat{y}$

Algorithm 2: Shrink_Grid

    Result: $x_{\text{start}}, y_{\text{start}}, x_{\text{end}}, y_{\text{end}}, d_x, d_y, dist$

1 Define $dist$ at first place

2 Note that the variable $dist$ symbolizes the distance from the progressive solutions to the multiple grids

3 Initialize parameters $\alpha, \beta \in (0,1)$

4 $dist = dist \times \beta$

5 $d_x = \alpha \times d_x$

6 $d_y = \alpha \times d_y$

7 $x_{\text{start}} = \hat{x} - dist$

8 $y_{\text{start}} = \hat{y} - dist$

9 $x_{\text{end}} = \hat{x} + dist$

10 $y_{\text{end}} = \hat{y} + dist$

11 **return** $x_{\text{start}}, y_{\text{start}}, x_{\text{end}}, y_{\text{end}}, d_x, d_y, dist$

The parameters α, β in line 3 of the method Shrink_Grid regulate the resolution and the size of the grid, respectively. Finally, it is noted that these parameters take values in the interval (0,1), because we progressively seek to shrink the grid.

Chapter 3

Phase-based Variant Maximum Likelihood

In this section, a phase-based passive tag positioning method is adopted [18], based on MLE. The likelihood function and the weight are reconstructed to mitigate the phase uncertainty and improve the positioning performance in the presence of multipath. Moreover, an extension of the original idea is presented, in particular, a method that does random sampling on reference measurements is constructed and it is tested on both simulated and real data.

3.1 Signal Model

The communication between an UHF-RFID reader and a passive tag depends on the backscatter modulation resulting from the varying load impedance, which consists of the forward and the backscatter link. The forward link powers the passive RFID tags, while the backscatter link reads the message in the tags. Considering a narrowband RFID reader with transceiver (Tx/Rx) , the transmitted signal at time t is given as

$$m\left(t\right) = \mu_{Tx}\left(t\right)\cos\left(2\pi f_c t + \varphi_{Tx}\right), \tag{50}$$

where μ_{Tx} denotes the amplitude, f_c the carrier frequency, φ_{Tx} the initial phase shift (caused by the transmitter's circuits and antenna) at the transmitter. The backscattered signal after the round-trip propagation can be given by

$$x_{Rx}(t) = \mu_{Rx}(t)\cos\left(2\pi f_c\left(t - 2\tau_0\right) + \varphi_{Tx,Rx}\right)x_{ID}\left(t - \tau_0\right), \tag{51}$$

where μ_{Rx} is the received amplitude, $\tau_0 = \frac{d}{c}$ represents one-way time-of-flight between the reader and the tag, d the distance from the antenna to the tag, and c the speed of light. $\varphi_{Tx,Rx}$ represents the phase shift introduced by the transceiver's hardware circuit and cables. The signal x_{ID} contains the tag's unique information i.e. EPC. Hence, the complex baseband equivalent of the signal is

$$x_{BS}(t) = \mu_{BS}(t)\exp\left\{-j\left(4\pi f_c\tau_0 + \varphi_{Tx,Rx} + \varphi_{tag}\right)\right\}, \tag{52}$$

where μ_{BS} is the amplitude, and φ_{tag} is the phase shift caused by tag's reflection characteristic.

Assuming no multipath interference, the exported phase from 52 at the n-th observation can be given by

$$\phi[n] = 4\pi f_c\tau_0 + \underbrace{\varphi_{Tx,Rx} + \varphi_{\text{tag}}}_{\varphi_0} \overset{\tau_0 = \frac{d[n]}{\lambda f_c}}{=} 4\pi\frac{d[n]}{\lambda} + \varphi_0, \tag{53}$$

where $\lambda = c/f$ is the wavelength. By referring to 53, the phase depends on the distance between the transceiver and the tag as well as on the technical characteristics of the devices involved. Additionally, the Impinj Speedway R420 RFID reader that we used in the experiments does a modulo operation, which limits our phase measurements to the interval $[0, 2\pi)$, namely, $\phi_m = \mathrm{mod}(\phi, 2\pi)$.

3.2 Distribution

In this section we will study the distribution of the phase measurements through simulations; similar conclusions have been drawn for the distribution of the phase measurements and through experimental data in [18].
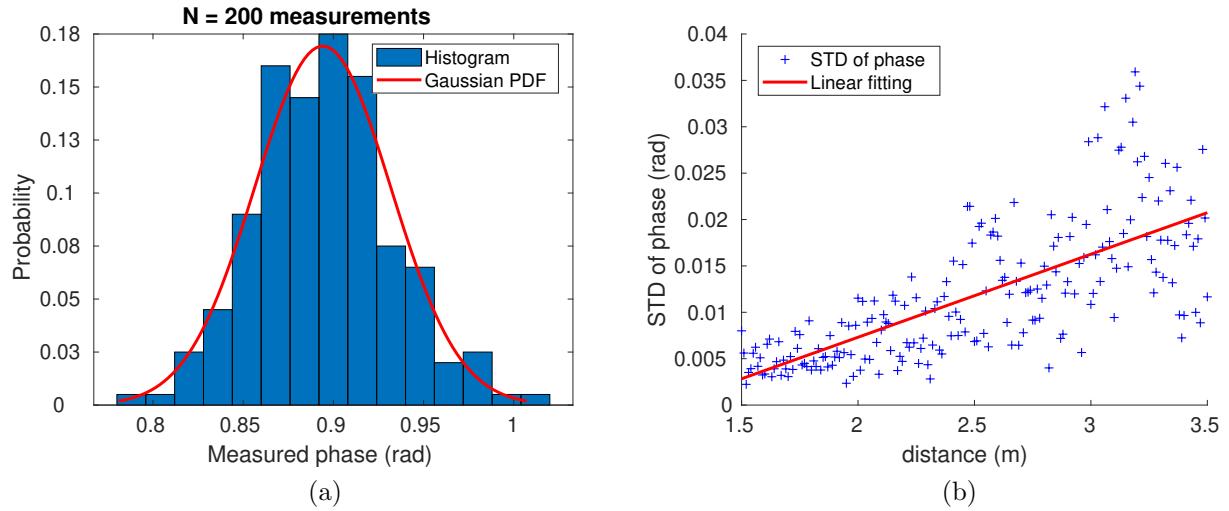
Figure 9: Measured phases: (a) the histogram of measured phase and Gaussian PDF fitting, (b) the STD of phase vs distance.

The measured phase is derived from the backscattered signal, which is always contaminated by the thermal noise and environmental clutter, leading to measurement errors. To examine the statistical distribution of the measured phase, we created simulations considering different channels, and distances. In order to produce the Fig. 9 (a) and Fig. 9 (b) we added on simulated data thermal noise $w(t)$ modeled by a complex, circularly symmetric, additive Gaussian noise process with the following power spectral density

$$S_{ww}(F) = \begin{cases} \frac{N_0}{2}, & |F| \leq BW \\ 0, & \text{else,} \end{cases} \tag{54}$$

where BW stands for receiver's bandwidth, $N_0 = kT$, k is the Boltzmann's constant and T is the receiver's temperature in Kelvin's scale. A single reflector is assumed as described

The parameters for the simulations were selected as $BW = 2$ MHz and $T = 300$ K.

in the section 4.1 and only one tag is utilized to extract information. In Fig. 9 (a), the normalized histogram of measured phases is depicted, along with the Gaussian PDF fitting curve. The histogram seems to have a good match with the Gaussian PDF, so it is justified to consider that the measured phases follow a Gaussian distribution as $\mathcal{N}\left(\mu, \sigma^2\right)$. Fig. 9 (b) shows the STD of the measured phases, it presents a slow increase concerning the distance. The STD of the measured phases can be approximated by $\sigma = 0.008d - 0.01$ rad using a linear fitting. Furthermore, we observe that the STD has negligible variations of 0.01; hence measured phases can be considered as a constant in a small range between 1.5 meters and 3 meters.

3.3 Maximum Likelihood Estimation

According to 3.2, the measured phases of the tag follow a Gaussian distribution, so it is reasonable to utilize MLE in order to solve the tag localization problem. Considering the N independent observations of the measured phases, the MLE is given by

$$\mathbf{P}_{tag} = \arg\max_{\mathbf{P}_{tag}} \prod_{n=0}^{N-1} \frac{1}{\sqrt{2\pi}\sigma[n]} e^{-\frac{\left(\phi_m[n] - \text{mod}\left(\frac{4\pi}{\lambda}d[n] + \varphi_0\right)\right)^2}{2\sigma^2[n]}}, \tag{55}$$

where $d = \|\mathbf{P}_{ant} - \mathbf{P}_{tag}\|_2$ is the distance from the antenna's (upon the reader) coordinates $\mathbf{P}_{ant}$ to the tag's coordinates $\mathbf{P}_{tag}$. Based on Fig. 9 (b), the STD can be considered as a constant. Hence, 55 can be rewritten as

$$
\begin{aligned}
\mathbf{P}_{tag} &= \arg\max_{\mathbf{P}_{tag}} \prod_{n=0}^{N-1} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{\left(\phi_m[n] - \text{mod}\left(\frac{4\pi}{\lambda}d[n] + \varphi_0\right)\right)^2}{2\sigma^2}} \\
&= \arg\max_{\mathbf{P}_{tag}} \sum_{n=0}^{N-1} \left[-\left(\phi_m[n] - \text{mod}\left(\frac{4\pi}{\lambda}d[n] + \varphi_0\right)\right)^2 \right].
\end{aligned}
\tag{56}
$$

For the same tag, phase shift φ_0 can be regarded as almost constant when the antenna moves along the tag (see [18] for the details on the feasibility of this assumption). Thus, a differential mitigation method is utilized to mitigate the uncertain phase shift φ_0. The first scheme of the differential mitigation is the misaligned subtraction, in which differences of adjacent measured phases are used in order to eliminate the impact of φ_0. Define $\Delta\phi_m^{[n,n-1]} = \phi_m[n] - \phi_m[n-1]$, and $\Delta\varphi_d^{[n,n-1]} = \text{mod}\left(\frac{4\pi}{\lambda}d[n] + \varphi_0\right) - \text{mod}\left(\frac{4\pi}{\lambda}d[n-1] + \varphi_0\right)$, so we have

$$\mathbf{P}_{tag} = \arg\max_{\mathbf{P}_{tag}} \sum_{n=1}^{N-1} \left[-\left(\Delta\phi_m^{[n,n-1]} - \Delta\varphi_d^{[n,n-1]}\right)^2 \right]. \tag{57}$$

It holds that (proof is presented in the appendix *)

$$\Delta\varphi_d^{[n,n-1]} = \begin{cases} \text{mod}\left(\frac{4\pi}{\lambda}\Delta d^{[n,n-1]}\right) - 2\pi, & \Delta\varphi_d^{[n,n-1]} \in (-2\pi, 0) \\ \text{mod}\left(\frac{4\pi}{\lambda}\Delta d^{[n,n-1]}\right), & \Delta\varphi_d^{[n,n-1]} \in [0, 2\pi). \end{cases} \tag{58}$$

32
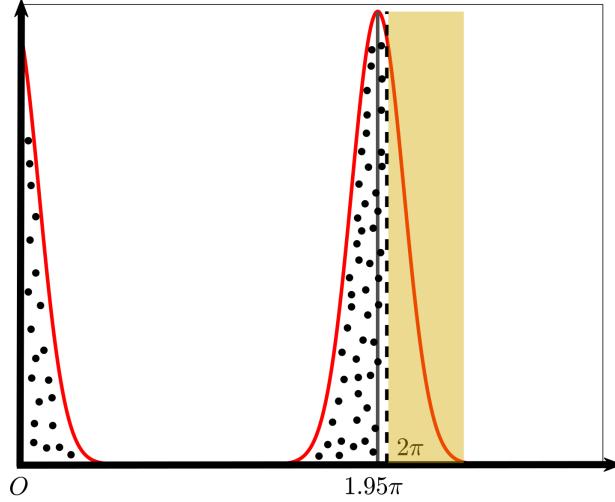
Figure 10: An example of phase jump when the actual phase is 1.95π rad.

The second scheme is selecting one of the measured phases as the reference $\phi_m[r]$, such as the phase at the first recorded position. Thus, the tag's position can be given as

$$\mathbf{P}_{tag} = \arg\max_{\mathbf{P}_{\text{tag}}} \sum_{n=0}^{N-1} \left[-\left(\Delta\phi_m^{[n,r]} - \Delta\varphi_d^{[n,r]} \right)^2 \right], \tag{59}$$

where $\Delta\phi_m^{[n,r]} = \phi_m[n] - \phi_m[r]$, and

$$\Delta\varphi_d^{[n,r]} = \begin{cases} \mod\left(\frac{4\pi}{\lambda}\Delta d^{[n,r]} \right) - 2\pi, & \Delta\varphi_d^{[n,r]} \in (-2\pi, 0) \\ \mod\left(\frac{4\pi}{\lambda}\Delta d^{[n,r]} \right), & \Delta\varphi_d^{[n,r]} \in [0, 2\pi). \end{cases} \tag{60}$$

According to 58 and 60, we observe that the phase uncertainty is mitigated. However, the judging condition in 58 and 60, namely $\Delta\varphi_d^{[n,n-1]} \gtrless 0$ or $\Delta\varphi_d^{[n,r]} \gtrless 0$, is undecidable.

3.4   Likelihood Function Reconstruction

In the previous section, we studied the difference of phase measurements, through which the phase uncertainty was reduced. However, a new problem arises, that of phase jumps as a result of the modulo-2π operations. With regard to Fig. 10, when the phase takes values in the borders of the interval $[0, 2\pi]$, the measured phase may jump to the value left of 0 or right of 2π. Hence, the likelihood function $f_{NLF}\left(\Delta\phi_m^{[n,r]} \mid d \right) = -\left(\Delta\phi_m^{[n,r]} - \Delta\varphi_d^{[n,r]} \right)^2$ in 59, which we could call as NLF, will cause large errors as a result of $\Delta\phi_m^{[n,r]}$ sharp jumping when the measured phase is at around 2π or 0 rad. For instance, when the actual measured phase $\phi_m[r] = 1.6\pi$ rad and $\phi_m[n] = 1.95\pi$ rad, the phase difference is $\Delta\phi_m^{[n,r]} = 0.35\pi$ rad. But

33

due to noise or other interference, $\phi_m[n]$ may jump to 0.03π rad, and then $\Delta\phi_m^{[n,r]} = -1.57\pi$ rad, which causes massive offset to the NLF.

To eliminate the discontinuities caused by the phase jump, a trigonometric function transformation is exploited. The cosine function is a good choice as it reduces sharp transitions from 0 to 2π or vice versa. Observe that 59 can be expressed as a function of the cosine, according to the second-order Taylor series. Thus, the maximum likelihood positioning estimation can be converted as

$$
\begin{aligned}
\mathbf{P}_{tag} &= \arg\max_{\mathbf{P}_{tag}} \sum_{n=0}^{N-1} \left[-\left(\Delta\phi_m^{[n,r]} - \Delta\varphi_d^{[n,r]} \right)^2 \right] \\
&\approx \arg\max_{\mathbf{P}_{tag}} \sum_{n=0}^{N-1} \left[2\cos\left(\Delta\phi_m^{[n,r]} - \Delta\varphi_d^{[n,r]} \right) - 2 \right] \\
&= \arg\max_{\mathbf{P}_{tag}} \sum_{n=0}^{N-1} \underbrace{\cos\left(\Delta\phi_m^{[n,r]} - \Delta\varphi_d^{[n,r]} \right)}_{f_{CLF}\left(\Delta\phi_m^{[n,r]} | d \right)},
\end{aligned}
\tag{61}
$$

where the reconstructed likelihood function in 61 is called CLF and marked as $f_{CLF}\left(\Delta\phi_m \mid d \right)$ (see * the proof of why the approximate equality holds). In addition, it is worth noting that the cosine transformation also eliminates the condition judgment (the decision) in 60, since the term -2π when $\Delta\varphi_d^{[n,r]} \in (-2\pi, 0)$ will not change the value of the likelihood function as a result of the cosine transformation. More specifically, recall the useful trigonometric identity $\cos(2\pi + x) = \cos(x)$; utilizing this property we get that

- If $\Delta\varphi_d^{[n,r]} \in [0, 2\pi)$, then

$$
\cos\left(\Delta\phi_m^{[n,r]} - \Delta\varphi_d^{[n,r]} \right) = \cos\left(\Delta\phi_m^{[n,r]} - \mathrm{mod}\left(\frac{4\pi}{\lambda} \Delta d^{[n,r]} \right) \right).
\tag{62}
$$

- If $\Delta\varphi_d^{[n,r]} \in (-2\pi, 0)$, then

$$
\begin{aligned}
\cos\left(\Delta\phi_m^{[n,r]} - \Delta\varphi_d^{[n,r]} \right) &= \cos\left(\Delta\phi_m^{[n,r]} - \left(\mathrm{mod}\left(\frac{4\pi}{\lambda} \Delta d^{[n,r]} \right) - 2\pi \right) \right) \\
&= \cos\left(\Delta\phi_m^{[n,r]} - \mathrm{mod}\left(\frac{4\pi}{\lambda} \Delta d^{[n,r]} \right) + 2\pi \right) \\
&= \cos\left(\Delta\phi_m^{[n,r]} - \mathrm{mod}\left(\frac{4\pi}{\lambda} \Delta d^{[n,r]} \right) \right),
\end{aligned}
\tag{63}
$$

which confirms the extermination of the $\Delta\varphi_d^{[n,r]} \gtrless 0$ condition.

In closing, the measurements must be evaluated for their quality. Namely, in real conditions RFID signal may contain contaminated components from the thermal noise, reflection, scattering, or other interference. Hence, the contaminated phase will contain an offset with the actual phase. To deal with the incident, a weighted MLE is utilized to augment the

likelihood function. The component with a smaller bias compared with the calculated phase will be assigned a larger weight. The weighted MLE can be given as

$$\mathbf{P}_{\text{tag}} = \arg\max_{\mathbf{P}_{\text{tag}}} \sum_{n=0}^{N-1} w[n] f_{CLF} \left(\Delta\phi_m^{[n,r]} \mid d \right), \tag{64}$$

where $w[n] \in [0,1]$ is constructed based on the fact that large values of $\left(\Delta\phi_m^{[n,r]} - \Delta\varphi_d^{[n,r]} \right)$ will be punished. Thus, the weights can be given by

$$w[n] = \left| f_{CLF} \left(\Delta\phi_m^{[n,r]} \mid d \right) \right|. \tag{65}$$

3.5 Algorithms

Below, the algorithms for determining the position of a tag in 2d are listed. The methods easily extend to 3d, simply by adding another z dimension to the grid search space. Note that Scheme 1 refers to the fact that the phase measurement differences are taken in pairs between adjacent antennas. As mentioned above, Scheme 2 is referring to selecting one of the measured phases as the reference, such as the phase at the first recorded position, and calculate all the differences between this reference measured phase and all the others.

Algorithms 3 and 4 do not take advantage of any property of a trigonometric function to deal with possible phase discontinuities, nor do they use any weight to mitigate the multipath. They try to directly maximize the likelihood of measurements' differences and that is why it is characterized as naive.

For algorithms 5 and 6, the cosine is used as an approximation through the Taylor series of the cost function and, through the absolute value of this function, weights are generated that are able to evaluate the quality of measurements.

Finally, algorithms 7 and 8 are an extension of algorithm 6. In particular, algorithm 7 exhaustively examines all possible measurements as a reference and concludes which of them maximizes the cost function. Algorithm 8 examines a subset, which is randomly selected.

Algorithm 3: Naive Likelihood Algorithm (NLA) - Scheme 1

Result: Estimated Coordinates $\hat{x}, \hat{y}$

1 Let ϕ_m the vector with the measured phases wrapped within $[0, 2\pi)$

2 P_{reader} are the coordinates of the reader for all the recorded measurements and N is the number of measurements

3 Select $x_{\text{start}}, x_{\text{end}}, y_{\text{start}}$ and y_{end} points.

4 Define the grid resolution d_x and d_y

5 $grid\_coords := \{\}$

6 $candidate\_values := []$

7 $[grid\_coords]=$Create_grid$(x_{\text{start}}, x_{\text{end}}, y_{\text{start}}, y_{\text{end}}, d_x, d_y)$

8 for P_{tag} *in* $grid\_coords$ **do**

9 $cost\_f = 0$

10   **for** $n = 2 : N$ **do**

11 $\Delta\phi_m^{[n,n-1]} = \phi_m[n] - \phi_m[n-1]$

12 $d[n-1] = \left\| P_{tag} - P_{reader}^{[n-1]} \right\|_2$

13 $d[n] = \left\| P_{tag} - P_{reader}^{[n]} \right\|_2$

14 $\Delta\varphi_d^{[n,n-1]} = \text{mod}\left(\frac{4\pi}{\lambda}d[n] + \varphi_0\right) - \text{mod}\left(\frac{4\pi}{\lambda}d[n-1] + \varphi_0\right)$

15 $\Delta d^{[n,n-1]} = d[n] - d[n-1]$

16 $\Delta\varphi_d^{[n,n-1]} = \begin{cases} \text{mod}\left(\frac{4\pi}{\lambda}\Delta d^{[n,n-1]}\right) - 2\pi, & \Delta\varphi_d^{[n,n-1]} \in (-2\pi, 0) \\ \text{mod}\left(\frac{4\pi}{\lambda}\Delta d^{[n,n-1]}\right), & \Delta\varphi_d^{[n,n-1]} \in [0, 2\pi) \end{cases}$

17 $cost\_f = cost\_f - \left(\Delta\phi_m^{[n,n-1]} - \Delta\varphi_d^{[n,n-1]}\right)^2$

18   **end**

19 $candidate\_values = [candidate\_values \ \ cost\_f]$

20 end

21 $[\sim, idx] = \max\left(candidate\_values\right)$

22 $est\_coords = grid\_coords\{idx\}$

23 $[\hat{x}, \hat{y}] = est\_coords$

Algorithm 4: Naive Likelihood Algorithm (NLA) - Scheme 2

Result: Estimated Coordinates $\hat{x}, \hat{y}$

1   Let ϕ_m the vector with the measured phases wrapped within $[0, 2\pi)$

2   P_{reader} are the coordinates of the reader for all the recorded measurements and N is the number of measurements

3   Select $x_{\text{start}}, x_{\text{end}}, y_{\text{start}}$ and y_{end} points

4   Define the grid resolution d_x and d_y

5   Select $r \in \{1, \ldots, N\}$ index for the reference phase

6    $grid\_coords := \{\}$

7    $candidate\_values := []$

8    $[grid\_coords]=\text{Create\_grid}(x_{\text{start}}, x_{\text{end}}, y_{\text{start}}, y_{\text{end}}, d_x, d_y)$

9   **for** P_{tag} *in* $grid\_coords$ **do**

10      $cost\_f = 0$

11   **for** $n = 1 : N$, $n \neq r$ **do**

12        $\Delta\phi_m^{[n,r]} = \phi_m[n] - \phi_m[r]$

13        $d[r] = \left\| P_{tag} - P_{reader}^{[r]} \right\|_2$

14        $d[n] = \left\| P_{tag} - P_{reader}^{[n]} \right\|_2$

15        $\Delta\varphi_d^{[n,r]} = \text{mod}\left(\frac{4\pi}{\lambda}d[n] + \varphi_0\right) - \text{mod}\left(\frac{4\pi}{\lambda}d[r] + \varphi_0\right)$

16        $\Delta d^{[n,r]} = d[n] - d[r]$

17        $\Delta\varphi_d^{[n,r]} = \begin{cases} \text{mod}\left(\frac{4\pi}{\lambda}\Delta d^{[n,r]}\right) - 2\pi, & \Delta\varphi_d^{[n,r]} \in (-2\pi, 0) \\ \text{mod}\left(\frac{4\pi}{\lambda}\Delta d^{[n,r]}\right), & \Delta\varphi_d^{[n,r]} \in [0, 2\pi) \end{cases}$

18        $cost\_f = cost\_f - \left(\Delta\phi_m^{[n,r]} - \Delta\varphi_d^{[n,r]}\right)^2$

19   **end**

20      $candidate\_values = [candidate\_values \;\; cost\_f]$

21   **end**

22    $[\sim, idx] = \max(candidate\_values)$

23    $est\_coords = grid\_coords\{idx\}$

24    $[\hat{x}, \hat{y}] = est\_coords$

Algorithm 5: Weighted Cosine Likelihood Algorithm (WCLA) - Scheme 1

Result: Estimated Coordinates $\hat{x}, \hat{y}$

1 Let ϕ_m the vector with the measured phases wrapped within $[0, 2\pi)$

2 P_{reader} are the coordinates of the reader for all the recorded measurements and N is the number of measurements

3 Select $x_{\text{start}}, x_{\text{end}}, y_{\text{start}}$ and y_{end} points

4 Define the grid resolution d_x and d_y

5 $grid\_coords := \{\}$

6 $candidate\_values := []$

7 $[grid\_coords] = \text{Create\_grid}(x_{\text{start}}, x_{\text{end}}, y_{\text{start}}, y_{\text{end}}, d_x, d_y)$

8 **for** P_{tag} *in grid_coords* **do**

9 $cost\_f = 0$

10   **for** $n = 2 : N$ **do**

11 $\Delta\phi_m^{[n,n-1]} = \phi_m[n] - \phi_m[n-1]$

12 $d[n-1] = \left\| P_{tag} - P_{reader}^{[n-1]} \right\|_2$

13 $d[n] = \left\| P_{tag} - P_{reader}^{[n]} \right\|_2$

14 $\Delta d^{[n,n-1]} = d[n] - d[n-1]$

15 $w[n] = \left| \cos\left(\Delta\phi_m^{[n,n-1]} - \frac{4\pi}{\lambda}\Delta d^{[n,n-1]} \right) \right|$

16 $cost\_f = cost\_f + w[n] \cos\left(\Delta\phi_m^{[n,n-1]} - \frac{4\pi}{\lambda}\Delta d^{[n,n-1]} \right)$

17   **end**

18 $candidate\_values = [candidate\_values \ \ cost\_f]$

19 **end**

20 $[\sim, idx] = \max(candidate\_values)$

21 $est\_coords = grid\_coords\{idx\}$

22 $[\hat{x}, \hat{y}] = est\_coords$

Algorithm 6: Weighted Cosine Likelihood Algorithm (WCLA) - Scheme 2

Result: Estimated Coordinates $\hat{x}, \hat{y}$

1 Let ϕ_m the vector with the measured phases wrapped within $[0, 2\pi)$

2 P_{reader} are the coordinates of the reader for all the recorded measurements and N is the number of measurements

3 Select $x_{\text{start}}, x_{\text{end}}, y_{\text{start}}$ and y_{end} points

4 Define the grid resolution d_x and d_y

5 Select $r \in \{1, \ldots, N\}$ index for the reference phase

6 $grid\_coords := \{\}$

7 $candidate\_values := []$

8 $[grid\_coords] = \text{Create\_grid}(x_{\text{start}}, x_{\text{end}}, y_{\text{start}}, y_{\text{end}}, d_x, d_y)$

9 **for** P_{tag} *in grid_coords* **do**

10 $\quad$ $cost\_f = 0$

11 $\quad$ **for** $n = 1 : N$, $n \neq r$ **do**

12 $\quad\quad$ $\Delta\phi_m^{[n,r]} = \phi_m[n] - \phi_m[r]$

13 $\quad\quad$ $d[r] = \left\| P_{tag} - P_{reader}^{[r]} \right\|_2$

14 $\quad\quad$ $d[n] = \left\| P_{tag} - P_{reader}^{[n]} \right\|_2$

15 $\quad\quad$ $\Delta d^{[n,r]} = d[n] - d[r]$

16 $\quad\quad$ $w[n] = \left| \cos\left(\Delta\phi_m^{[n,r]} - \frac{4\pi}{\lambda}\Delta d^{[n,r]}\right) \right|$

17 $\quad\quad$ $cost\_f = cost\_f + w[n] \cos\left(\Delta\phi_m^{[n,r]} - \frac{4\pi}{\lambda}\Delta d^{[n,r]}\right)$

18 $\quad$ **end**

19 $\quad$ $candidate\_values = [candidate\_values \ \ cost\_f]$

20 **end**

21 $[\sim, idx] = \max(candidate\_values)$

22 $est\_coords = grid\_coords\{idx\}$

23 $[\hat{x}, \hat{y}] = est\_coords$

Algorithm 7: Deterministic Sampling - Weighted Cosine Likelihood Algorithm (DSWCLA) - Proposed Approach-1

Result: Estimated Coordinates $\hat{x}, \hat{y}$

1 Let ϕ_m the vector with the measured phases wrapped within $[0, 2\pi)$.

2 P_{reader} are the coordinates of the reader for all the recorded measurements and N is the number of measurements.

3 Select $x_{\text{start}}, x_{\text{end}}, y_{\text{start}}$ and y_{end} points

4 Define the grid resolution d_x and d_y

5 $ref\_points = 1 : N$

6 $values := []$

7 $indices := []$

8 $grid\_coords := \{\}$

9 $[grid\_coords] = \text{Create\_grid}(x_{\text{start}}, x_{\text{end}}, y_{\text{start}}, y_{\text{end}}, d_x, d_y)$

10 **for** ref_idx in ref_points **do**

11 $\quad r = ref\_idx$

12 $\quad candidate\_values := []$

13 $\quad$ **for** P_{tag} in $grid\_coords$ **do**

14 $\quad\quad cost\_f = 0$

15 $\quad\quad$ **for** $n = 1 : N$, $n \neq r$ **do**

16 $\quad\quad\quad \Delta\phi_m^{[n,r]} = \phi_m[n] - \phi_m[r]$

17 $\quad\quad\quad d[r] = \left\| P_{tag} - P_{reader}^{[r]} \right\|_2$

18 $\quad\quad\quad d[n] = \left\| P_{tag} - P_{reader}^{[n]} \right\|_2$

19 $\quad\quad\quad \Delta d^{[n,r]} = d[n] - d[r]$

20 $\quad\quad\quad w[n] = \left| \cos\left(\Delta\phi_m^{[n,r]} - \frac{4\pi}{\lambda} \Delta d^{[n,r]} \right) \right|$

21 $\quad\quad\quad cost\_f = cost\_f + w[n] \cos\left(\Delta\phi_m^{[n,r]} - \frac{4\pi}{\lambda} \Delta d^{[n,r]} \right)$

22 $\quad\quad$ **end**

23 $\quad\quad candidate\_values = [candidate\_values \ \ cost\_f];$

24 $\quad$ **end**

25 $\quad [val, idx] = \max(candidate\_values)$

26 $\quad values := [values \ val]$

27 $\quad indices := [indices \ idx]$

28 **end**

29 $[\sim, idx] = \max(values)$

30 $est\_coords = grid\_coords \{indices(idx)\}$

31 $[\hat{x}, \hat{y}] = est\_coords$

Algorithm 8: Random Sampling - Weighted Cosine Likelihood Algorithm (RSWCLA) - Proposed Approach-2

Result: Estimated Coordinates $\hat{x}, \hat{y}$

1 Let ϕ_m the vector with the measured phases wrapped within $[0, 2\pi)$

2 P_{reader} are the coordinates of the reader for all the recorded measurements and N is the number of measurements

3 Select $x_{\text{start}}, x_{\text{end}}, y_{\text{start}}$ and y_{end} points

4 Define the grid resolution d_x and d_y

5 $ref\_points \subset 1 : N$ % Select randomly $\lceil \frac{N}{l} \rceil$ unique elements from $1 : N$, $1 \leq l \leq N$

6 $values := []$

7 $indices := []$

8 $grid\_coords := \{\}$

9 $[grid\_coords]=\text{Create\_grid}(x_{\text{start}}, x_{\text{end}}, y_{\text{start}}, y_{\text{end}}, d_x, d_y)$

10 for ref_idx in ref_points **do**

11 $\quad r = ref\_idx$

12 $\quad candidate\_values := []$

13 $\quad$ **for** P_{tag} in $grid\_coords$ **do**

14 $\quad\quad cost\_f = 0$

15 $\quad\quad$ **for** $n = 1 : N$, $n \neq r$ **do**

16 $\quad\quad\quad \Delta\phi_m^{[n,r]} = \phi_m[n] - \phi_m[r]$

17 $\quad\quad\quad d[r] = \left\| P_{tag} - P_{reader}^{[r]} \right\|_2$

18 $\quad\quad\quad d[n] = \left\| P_{tag} - P_{reader}^{[n]} \right\|_2$

19 $\quad\quad\quad \Delta d^{[n,r]} = d[n] - d[r]$

20 $\quad\quad\quad w[n] = \left| \cos\left(\Delta\phi_m^{[n,r]} - \frac{4\pi}{\lambda} \Delta d^{[n,r]} \right) \right|$

21 $\quad\quad\quad cost\_f = cost\_f + w[n] \cos\left(\Delta\phi_m^{[n,r]} - \frac{4\pi}{\lambda} \Delta d^{[n,r]} \right)$

22 $\quad\quad$ **end**

23 $\quad\quad candidate\_values = [candidate\_values \ \ cost\_f]$

24 $\quad$ **end**

25 $\quad [val, idx] = \max\left(candidate\_values \right)$

26 $\quad values = [values \ val]$

27 $\quad indices = [indices \ idx]$

28 end

29 $[\sim, idx] = \max\left(values \right)$

30 $est\_coords = grid\_coords\left\{ indices\left(idx \right) \right\}$

31 $[\hat{x}, \hat{y}] = est\_coords$

It is worth noting that if $l = N$ then algorithm 8 is identical to the algorithm 6.

Chapter 4

Simulations

In this section, we tested the performance of our algorithms under some assumptions as well as the effect of frequency on the constructed phases. The performance of the algorithms was measured with metrics that capture the mean error after many experiments. Below we explain, which metrics were selected.

Let $\mathbf{x}$ be the actual coordinates of a tag in either a 2d or 3d environment and $\hat{\mathbf{x}}$ the estimation of the coordinates resulting from the algorithms. Therefore the error is defined as $e := \|\mathbf{x} - \hat{\mathbf{x}}\|_2$. In our experiments two metrics where utilized MAE and RMSE, in particular MAE was set as $\mathcal{E}\{e\}$ and RMSE as $\sqrt{\mathcal{E}\{e^2\}}$. The same metrics for evaluating algorithms were used in the experimental section. It is worth noting that due to the triangular inequality, we expect that $\mathcal{E}\{e\} \le \sqrt{\mathcal{E}\{e^2\}}$ to be valid. The aforementioned metrics were also used in the experimental section.

Lastly, for all of the simulations, the parameter r in algorithm 4 was chosen to be 1. This implies that the first measurement is chosen as reference. Additionally, the regularization parameter λ in algorithm 1 was set as $\lambda = 0.95\|\tilde{\mathbf{\Phi}}^\dagger \mathbf{y}\|_\infty$, and $T = 1$ assuming no grid division. Finally, for the algorithm 8, l parameter was chosen as $l = 4$.

4.1   Measurement Model

In the simulations, the received signal in the presence of one tag and one carrier frequency f_c modeled as follows. For the direct link it holds that

$$x[n] = \sqrt{2P_c}\, s\, |\gamma_0 - \gamma_1|\, G_{\text{Tx}}\, G_{\text{Rx}} \left(\frac{\lambda}{4\pi d_0^{(n)}}\right)^2 \exp\left(-j4\pi \frac{d_0^{(n)}}{\lambda}\right), \tag{66}$$

where P_c is the carrier transmission power at passband, s is the power reflection coefficient, γ_0, γ_1 are the antenna loads, λ is the wavelength and $d_0^{(n)}$ is the distance of the direct path between the tag and the n-th transceiver. Let $\left(x_T^{(n)}, y_T^{(n)}\right)$ the coordinates of the n-th transceiver and $(x_{\text{tag}}, y_{\text{tag}})$ are the coordinates of the tag. Thus,

$$d_0^{(n)} = \sqrt{\left(x_{\text{tag}} - x_T^{(n)}\right)^2 + \left(y_{\text{tag}} - y_T^{(n)}\right)^2}. \tag{67}$$

Finally, G_{Tx} is the gain of the transmitting antenna in the direction of the receiving antenna and G_{Rx}[3] is the gain of the receiving antenna.

The equation 66 is approximate because, in general, the quantities of G_{Tx} and G_{Rx} are dependent from the wavelength, λ, therefore, from the modulation frequency. What stays

---

[3]Note that we set $G_{\text{Tx}} = G_{\text{Rx}} = 1$ and s is expressed in the amplitude of the signal.

approximate is that "the power of the received signal at the edge of the receiving antenna increases when the wavelength increases, that is, the modulation frequency decreases".

According to Fig. 2 the clutter is given by the equation

$$z[n] = \gamma_w \sqrt{2P_c}\, s\, |\gamma_0 - \gamma_1|\, G_{\text{Tx}}\, G_{\text{Rx}} \left(\frac{\lambda}{4\pi d_0^{(n)}}\right) \left(\frac{\lambda}{4\pi d_r^{(n)}}\right) \exp\left(-j2\pi \frac{\left(d_0^{(n)} + d_r^{(n)}\right)}{\lambda}\right), \quad (68)$$

where $\gamma_w \in (0,1)$ is a parameter that specifies the percentage of the signal that will fade when the signal hits the wall. Also $d_r^{(n)}$ is the distance of the indirect path and is calculated using the virtual point. As a result, if there is a wall at position x_w parallel to y-axis the distance between the virtual point and the n-th transceiver will be

$$d_r^{(n)} = \sqrt{\left(2x_w - x_{\text{tag}} - x_T^{(n)}\right)^2 + \left(y_{\text{tag}} - y_T^{(n)}\right)^2}. \quad (69)$$

White Complex Gaussian noise assumed $w[n] \sim \mathcal{CN}\left(0, \sigma^2\right)$. More precisely,

$$\sigma^2 = \frac{\mathcal{E}\{|x[n] + z[n]|^2\}}{10^{\frac{\text{SNR}}{10}}}. \quad (70)$$

Thus, the noisy received signal is

$$y[n] = x[n] + z[n] + w[n]. \quad (71)$$

4.2 Plots

4.2.1 Measured Phase in different carrier frequencies

It is important to consider how phase measurements are affected by the carrier frequency, for a given topology.
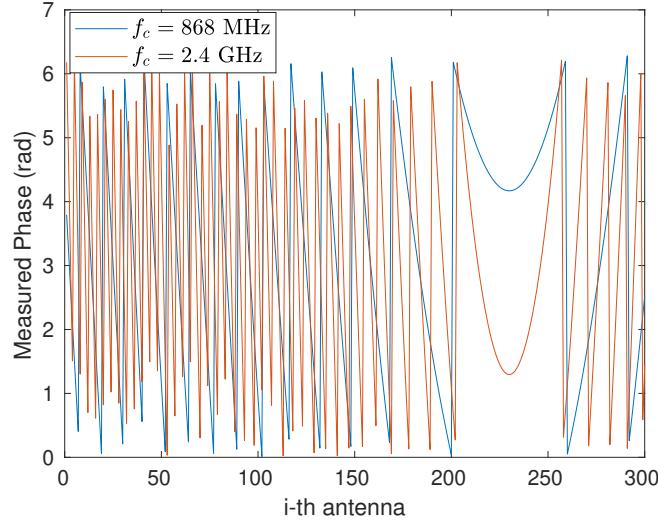
Figure 11: Measured phase for a given topology. The phase for frequency $f_c = 860$ MHz is presented in blue, and the phase for frequency $f_c = 2.4$ GHz in red.

Observing Fig. 11 one can easily see that when the carrier frequency increases then the phase seems to have large fluctuations. Hence, for adjacent antenna positions, more intensely phase differences are possible for large frequencies, something that results in greater resolution.

4.2.2 Evaluation - 2D Model

Below, we will examine how our algorithms behave in the case of two dimensions, i.e. the reader's antenna is at the same height as the tag. In cases where only the direct link was modeled, according to section 4.1, the parameters were selected as $P_c = 0.1$ Watt, $s = 0.1$, $G_{\text{Tx}} = 1$, $G_{\text{Rx}} = 1$, $\gamma_0 = 0$, $\gamma_1 = 1$. For the case of a possible reflector, the γ_w parameter was set to 0.5. The number of measurements (N), the carrier frequency (f_c), and the distance between the transceivers (inter-element spacing) will always be indicated in the title of the figures. Finally, it is worth noting that in all the following experiments, we assumed a 1×1 grid with resolution $d_x = d_y = 0.1$, and placed the tag in the coordinates of the grid so that we can guarantee that it is possible to accurately locate the tag.

Direct path modeling - 2D Model

In Fig. 12, we investigate the case of the presence of only the direct path; for instance, the signal propagates freely in space reflecting only on the target and not on an adjacent wall.

N=100 measurements - inter-element spacing 1.9 (cm)
$f_c = 868$ MHz

(a)

(b)
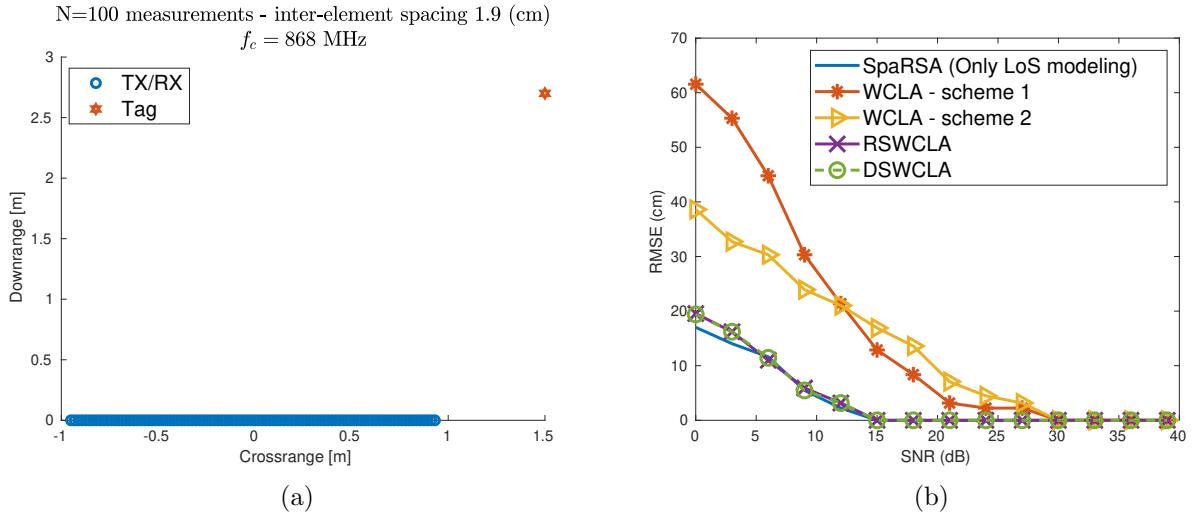
Figure 12: (a) Topology (2d) of the transceivers and the tag, no wall assumed. (b) SNR vs RMSE 100 Monte Carlo experiments in the presence of white complex gaussian noise.

We observe that all algorithms from 30 SNR in dB and above result in zero error in estimating the tag coordinates. Moreover, the algorithms SpaRSA, DSWCLA, and RSWCLA seem to be more resilient to stronger noise compared to WCLA methods for both schemes.

In Fig. 13, we produce the same scenario as in Fig. 12 with the only difference that we have placed a wall at position $x_w = 3$, in order to study the effect of a reflector on the algorithms in terms of performance, despite the under-modeling.

N=100 measurements - inter-element spacing 1.9 (cm)
$f_c = 868$ MHz

(a)

(b)
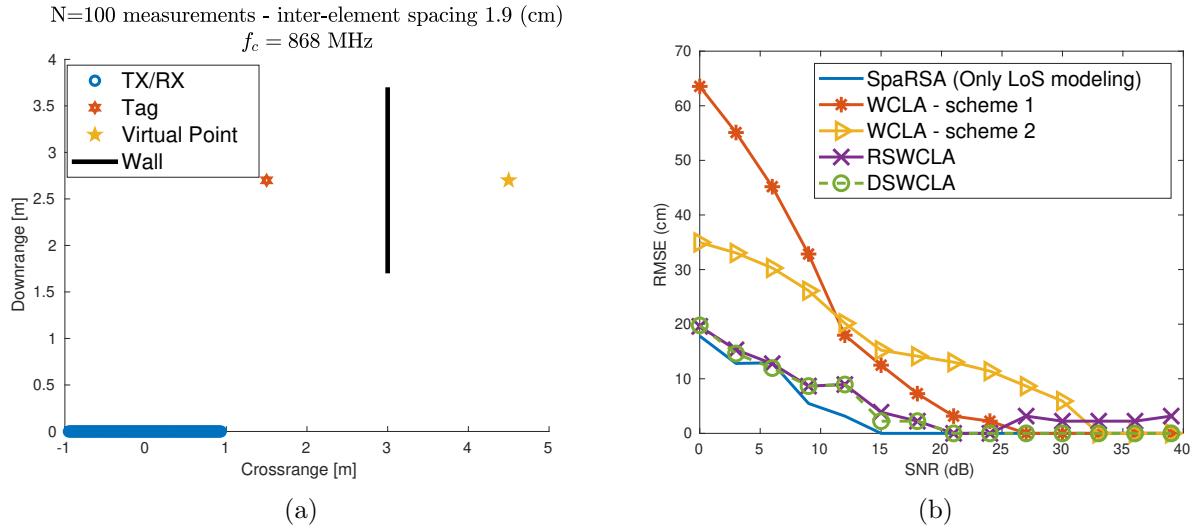
Figure 13: (a) Topology (2d) of the transceivers and the tag, a wall assumed at position $x_w = 3$. (b) SNR vs RMSE 100 Monte Carlo experiments in the presence of white complex gaussian noise.

What we find is that the non-white noise introduced by the multipath seems to reduce the performance of the algorithms in terms of estimation error. In particular, we observe that zero error is achieved for slightly higher SNRs than in the previous case, where we assumed only LoS, and that the RSWCLA algorithm does not seem to be able to achieve zero error for the given number of measurements.

Direct and indirect path modeling - 2D Model

Our next goal was to build a scenario through which the performance of the modeling of a possible reflector in space will be evaluated. Modeling requires knowledge of the location of the adjacent wall where some reflection will occur. It makes sense in the same scenario to compare the results of the under-modeling that SpaRSA can offer.
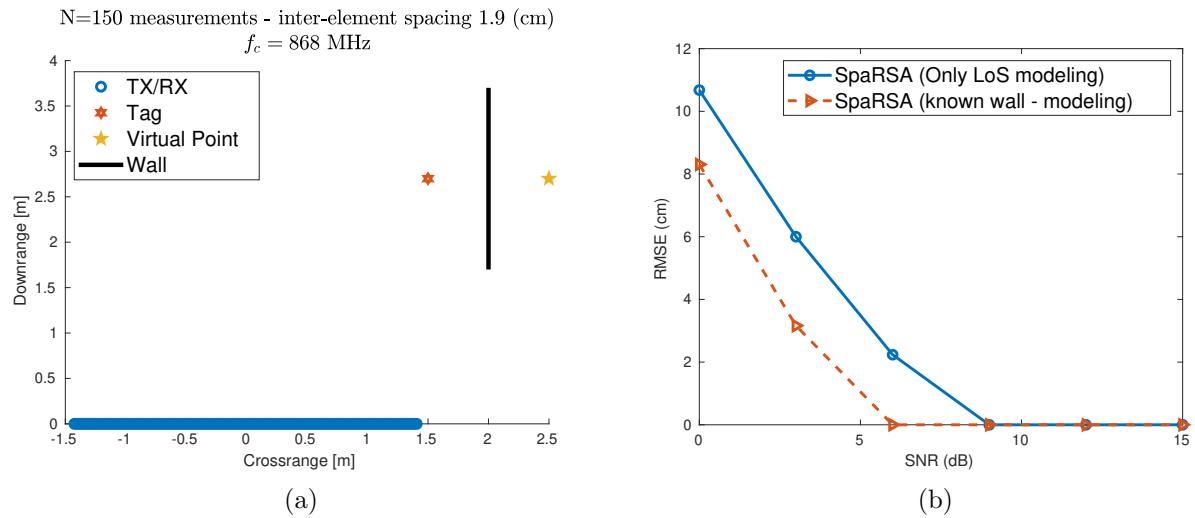

Figure 14: (a) Topology (2d) of the transceivers and the tag, a wall assumed at position $x_w = 2$. (b) SNR vs RMSE 100 Monte Carlo experiments in the presence of white complex gaussian noise.

Thus, in Fig. 14 we have placed a wall in position $x_w = 2$ to enhance the multipath effect, but we have increased the number of measurements from 100 to 150. What we find is that wall modeling greatly enhances the performance of the algorithm SpaRSA by solving a group sparse problem given the specific topology, while wall under-modeling leads to degraded performance.

After checking the behavior of the system for parallel walls with respect to the $y'y$-axis, our next thought was to test a scenario in which the wall would be parallel to the $x'x$-axis and therefore parallel to the transceivers.
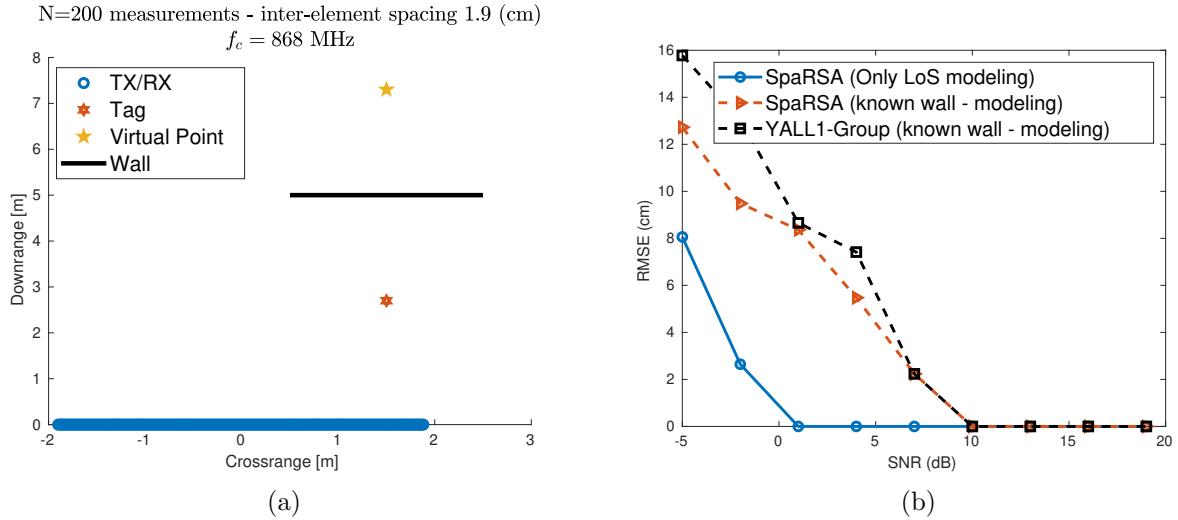
Figure 15: (a) Topology (2d) of the transceivers and the tag, a wall assumed at position $y_w = 5$ (wall is parallel to transceivers), low carrier frequency and inter-element spacing $< \frac{\lambda}{10}$ between the antennas was considered. (b) SNR vs RMSE 100 Monte Carlo experiments in the presence of white complex gaussian noise.

By observing Fig. 15, we find that given the topology and the distance between the transceivers, the modeling of the reflector leads to a larger error than if we did not model it at all. To discover if this is not due to an inaccurate solution of the optimization problem that SpaRSA solves we also included the Dual-Based ADM for Group Sparsity algorithm that has been implemented under the framework called YALL1-Group. We find that the algorithms have almost the same performance for different SNR values, so our next thought was to check what effect the distance between the transceivers has on the estimation.
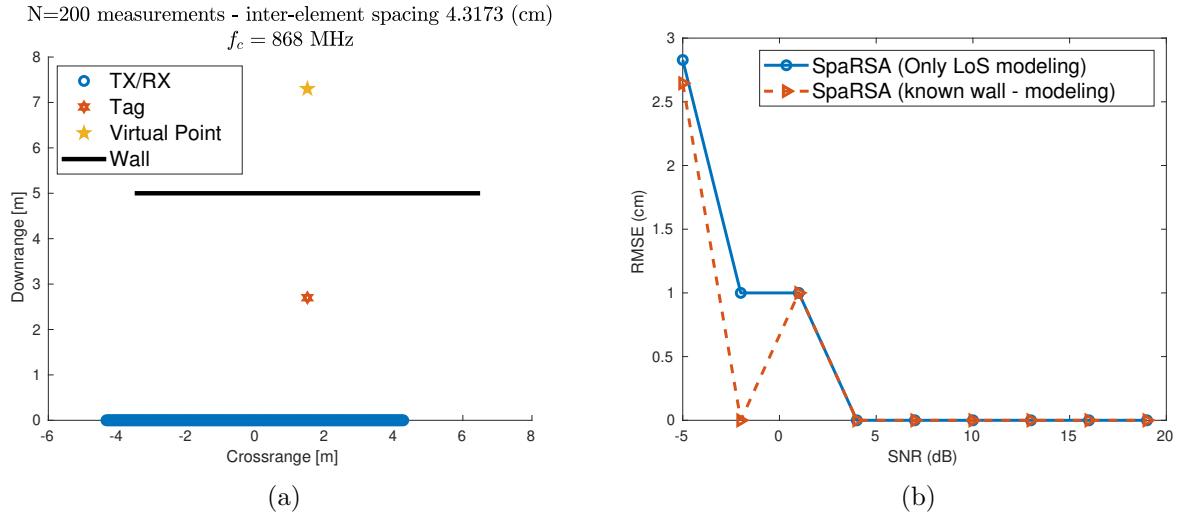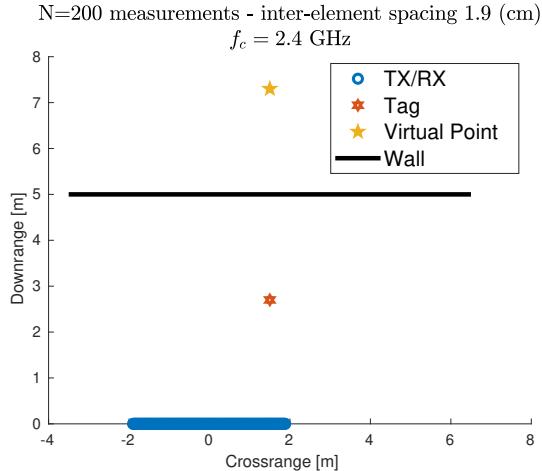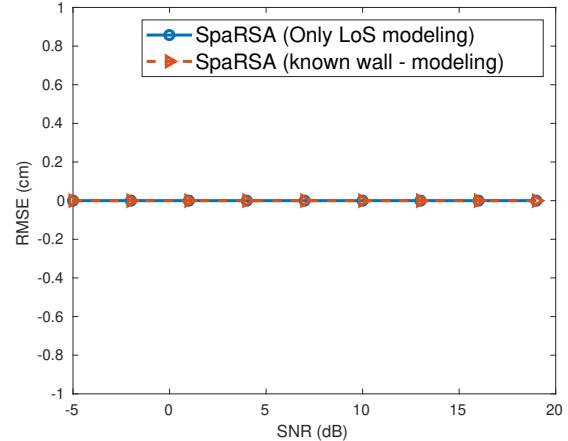
Figure 16: (a) Topology (2d) of the transceivers and the tag, a wall assumed at position $y_w = 5$ (wall is parallel to transceivers), low carrier frequency and inter-element spacing $> \frac{\lambda}{10}$ between the antennas was considered. (b) SNR vs RMSE 100 Monte Carlo experiments in the presence of white complex gaussian noise.

Surprisingly, looking at Fig. 16, we find that by changing the distance between the transceivers so that it is greater than $\frac{\lambda}{10}$, the performance of the algorithm with the modeling becomes meaningful again. However, in practical scenarios such as being in a room, it is not easy to spread the measurements enough to meet the requirements for good modeling. To deal with this difficulty we can increase the carrier frequency in order to reduce the wavelength. What we expect based on section 2.6.1 is to build a matrix $\tilde{\boldsymbol{\Phi}}$, which will lead to a better solution by solving the group sparse optimization problem, despite the relatively short distance between the transceivers.

N=200 measurements - inter-element spacing 1.9 (cm)
$f_c = 2.4$ GHz

(a)

(b)

Figure 17: (a) Topology (2d) of the transceivers and the tag, a wall assumed at position $y_w = 5$ (wall is parallel to transceivers), high carrier frequency and inter-element spacing $> \frac{\lambda}{10}$ between the antennas was considered. (b) SNR vs RMSE 100 Monte Carlo experiments in the presence of white complex gaussian noise.

Fig. 17 confirms that a high frequency can lead to zero error; it is very impressive that for all SNRs the performance of the algorithms either with full modeling or with under-modeling is perfect. This finding pushes us in the direction of making custom tags that will work on Wi-Fi frequencies, except for the already existing standard function of commercial tags around 868 MHz.

4.2.3 Evaluation - 3D Model

In the last part of the simulations, we will examine the operation of our algorithms in the case of three dimensions, which implies that the reader's antenna and the tag are at different heights. The parameters were selected as in section 4.2.2. Finally, we should highlight that in all the following experiments we assumed a $1 \times 1 \times 1$ grid with resolution $d_x = d_y = d_z = 0.1$, and placed the tag in the coordinates of the grid so that we can guarantee that we can accurately target the solution.

Direct path modeling - 3D Model

In Fig. 18, we produce a scenario closer to real conditions given that we generalize to the three dimensions. In the first place, only a direct path is assumed.

50

(a)

(b)
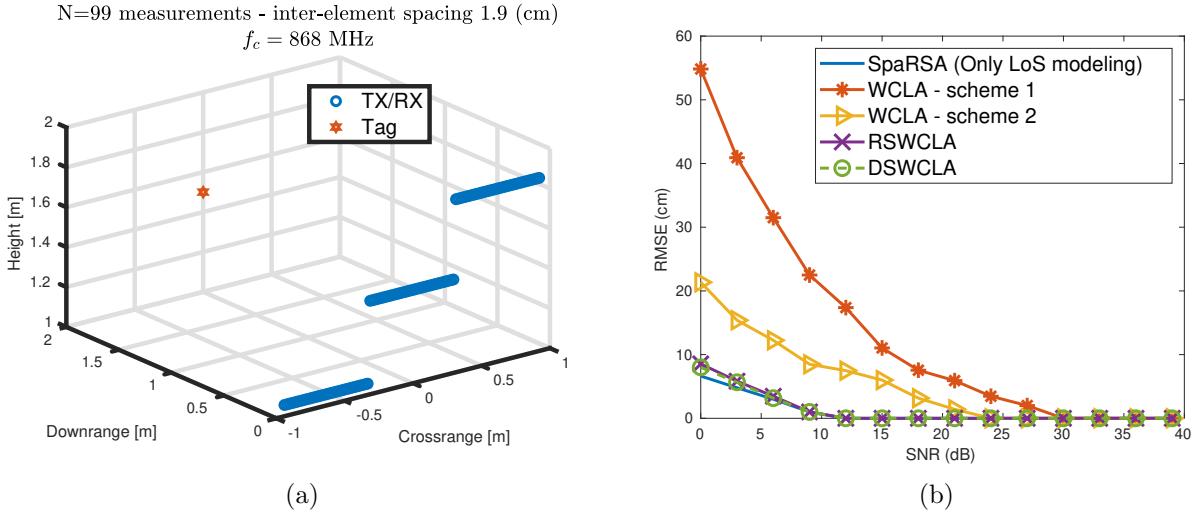
Figure 18: (a) Topology (3d) of the transceivers and the tag, no wall assumed. (b) SNR vs RMSE 100 Monte Carlo experiments in the presence of white complex gaussian noise.

As in the case of the two dimensions, we observe again that all algorithms from 30 SNR in dB and above lead to zero error in estimating the target. It is noted again that the SpaRSA, DSWCLA, and RSWCLA algorithms are superior to all the others in low SNRs given the topology.

It would be interesting to see again how our algorithms behave in the presence of a wall in three dimensions by under-modeling the reflection that is created. What we expect is a degraded performance due to the reflector that we do not take into account.
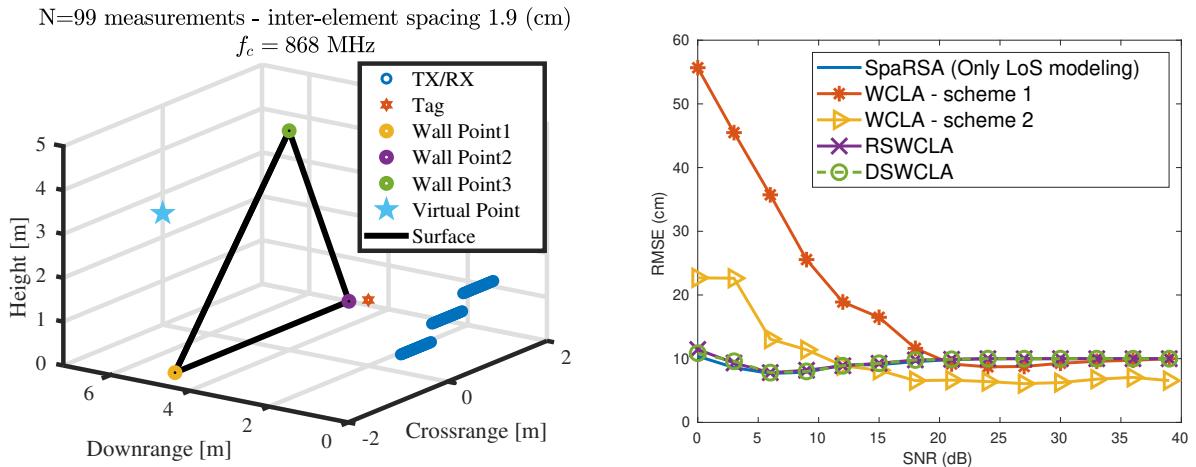

Figure 19: (a) Topology (3d) of the transceivers and the tag, a wall assumed, which is fully described by the three points $\{(-1.5, 5, 0), (2, 5, 0), (0, 4, 5)\}$. (b) SNR vs RMSE 100 Monte Carlo experiments in the presence of white complex gaussian noise. Under-modeling was adopted by all methods.

51

Indeed, based on Fig. 19, we confirm our assumptions, finding that even for large SNRs, the algorithms never manage to make an accurate estimation of the target.

Direct and indirect path modeling - 3D Model

Finally, we inspect the performance of the SpaRSA algorithm taking into account the reflection created by a known wall in three dimensions. As mentioned in section 2.3, to fully describe a surface in three dimensions we need three points that belong to it. Through these three points the vector that is perpendicular to the surface can be found and consequently, the virtual point. Therefore, after considering these points, we simulate the modeling and produce Fig. 20, which suggests that if we take the reflection created in this wall into account, the error becomes zero for SNR values greater than 18.
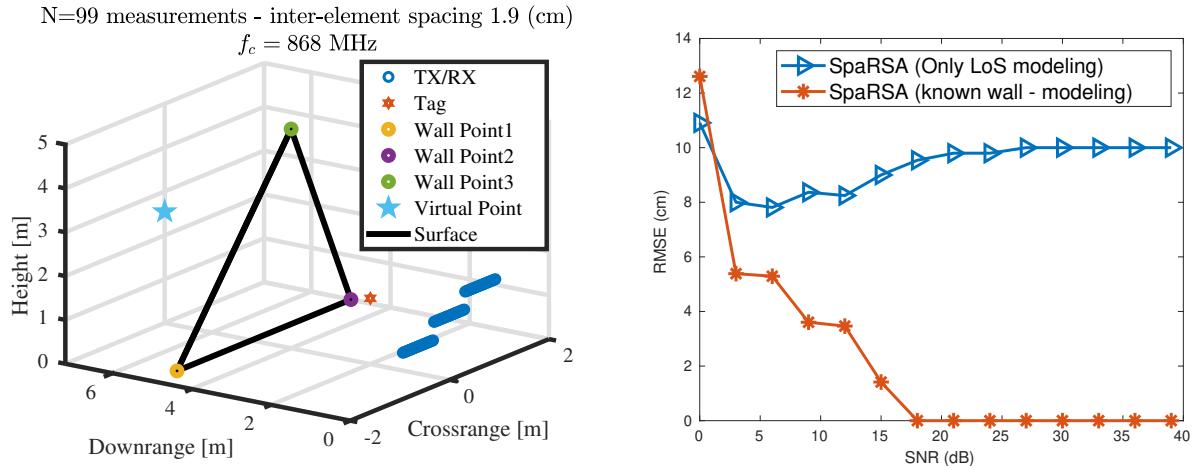

Figure 20: (a) Topology (3d) of the transceivers and the tag, a wall assumed, which is fully described by the three points $\{(-1.5, 5, 0), (2, 5, 0), (0, 4, 5)\}$. (b) SNR vs RMSE 100 Monte Carlo experiments in the presence of white complex gaussian noise. Under-modeling vs Exact-modeling is depicted.

Chapter 5

Real data experiments

Four indoor experimental campaigns were performed, with an Impinj Speedway R420 RFID reader. By exploiting Impinj's LTK API, a software was developed to control the reader parameters [19].

5.1 Measurement Processing

The SpaRSA method, which solves an optimization problem, uses data described in section 2.1 theoretically, and in section 4.1 modeling the signal of a tag at the simulation level. In practice, the reader that we used in our experiments provides RSSI measurements and phase measurements. Due to the modeling we described in the previous sections, it makes sense to use both phase and RSSI measurements. In particular, if $\boldsymbol{\mu}_{RS}$ is the RSSI measurement vector, which is converted from dBms to volts and $\boldsymbol{\phi}_{reader}$ is the phase measurement vector in the interval $[0, 2\pi)$, then the measurements that will be included in the algorithm are

$$\boldsymbol{y} = \boldsymbol{\mu}_{RS} \odot e^{-j\boldsymbol{\phi}_{reader}}. \tag{72}$$

The probabilistic methods as mentioned in section 3 utilize only the phase vector $\boldsymbol{\phi}_{reader}$.

For all the campaigns the carrier frequency was regulated at $f_c = 866.9$ MHz, and for the algorithms 4, 6 the first measurement was always chosen as the reference measurement, for all the cases. Therefore, we assume that $r = 1$. Moreover, the regularization parameter λ in algorithm 1 was set at $\lambda = 0.95\|\tilde{\boldsymbol{\Phi}}^\dagger \boldsymbol{y}\|_\infty$ and $T = 1$ for all the campaigns, which indicates that we assume no grid division. Finally, for the algorithm 8, l parameter was chosen as $l = 4$ for campaigns 1,2 and $l = 80$ for campaigns 3,4, due to the increased number of measurements.

How to convert dBm to watts and watts to volts

The power conversion of dBm to watts is given by the formula

$$P(W) = 1W \; \frac{10^{\left(\frac{P_{(\text{dBm})}}{10}\right)}}{1000} = 1W \; 10^{\frac{\left(P_{(\text{dBm})} - 30\right)}{10}}. \tag{73}$$

Power to Voltage Conversion in a $R = 50\ \Omega$ System is

$$V_{\text{peak}}(V) = \sqrt{2P(W)R} \quad [V]. \tag{74}$$

For campaigns 1,2, and 3 a 1×1 grid with resolution $d_x = d_y = 0.01$ was considered for all the methods.

5.2 Campaign-1: Hall Measurements (2D)

In the first campaign Fig. 21, the reader was connected to an MTI MT-242032 7 dBi antenna via a 0.9 dB loss coaxial cable. Alien ALN-9540 (Higgs-2) tag was employed at fixed distances. Reader transmission (Tx) power was configured in the range of 20 dBm to 30 dBm. The antenna was moved manually along a 3 m line at intervals of 5 cm, tag was placed at the bisector of the antenna's trajectory at distance of 1 m. The antenna and the tag were placed at a height of 1.52 m. Tag interrogations lasting 4 seconds were tested at every position of the antenna, for each of the different Tx power levels.

(a)   (b)
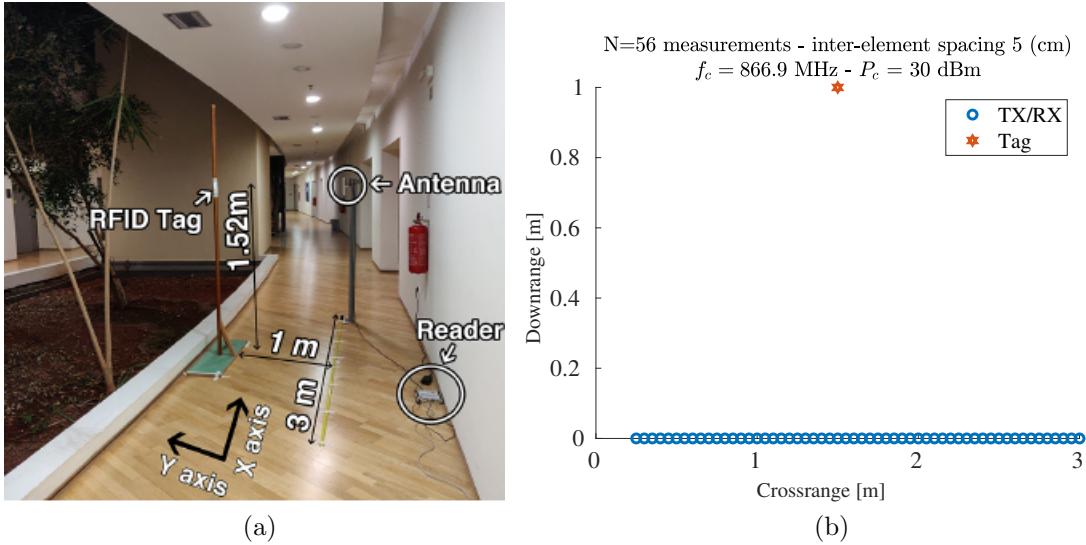
Figure 21: (a) Baseline experimental campaign-1: light multipath environment with manual movement of the reader antenna. (b) Topology at $P_c = 30$ dBm.

Based on Fig. 22, we find that as we increase the transmission power, the number of measurements also increases.
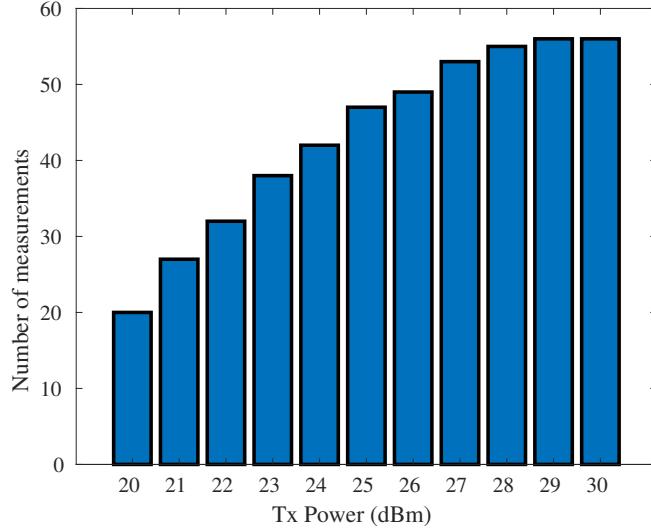
Figure 22: Number of measurement vs Tx Power.

Results for the light multipath environment are depicted in Fig. 23 , where we observe that the algorithms SpaRSA with only modeling the direct link, RSWCLA and DSWCLA perform better than all the others in terms of "absolute" error as we expected based on the simulations. Note that for this campaign we are not referring to mean absolute localization error, because we have only one tag in different power levels. Finally, the other methods also perform satisfactorily but with several fluctuations.
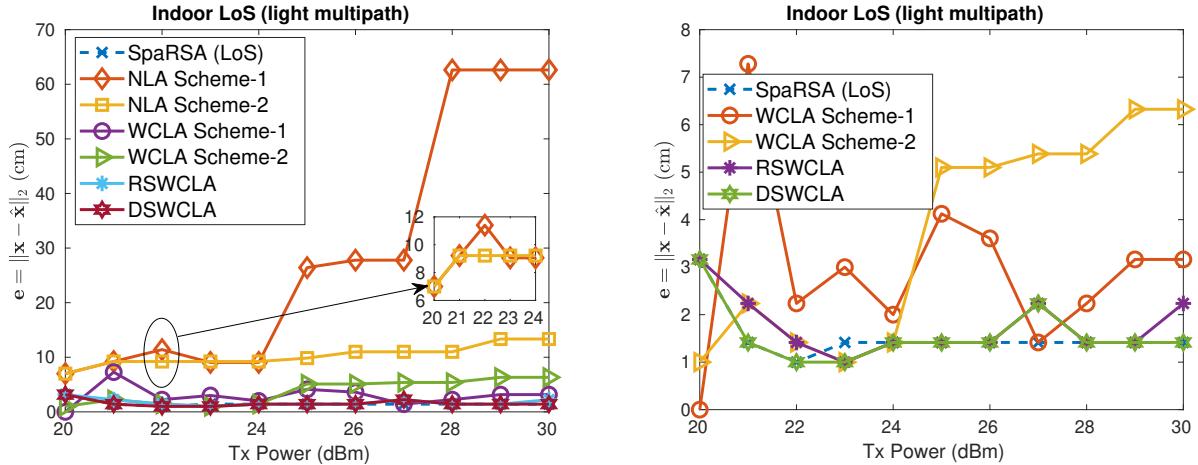

Figure 23: Campaign-1 localization error vs Tx power.

5.3 Campaign-2: Manual Measurements Lab (2D)

In the second campaign Fig. 24, a manual reader movement setup was tested again with

55

15 tags on books, packed on a bookshelf, with an approximately 5 cm distance between them. Measurements were captured every 5 cm and reader transmission (Tx) power was configured in the range of 20 dBm to 30 dBm with step 1, similar to the first campaign. The purpose of this campaign was to test the algorithms in a harsh environment, given the presence of reflections from the interior walls.
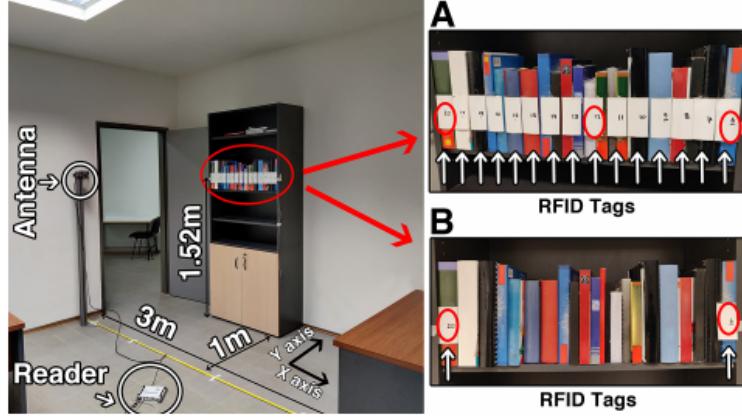

Figure 24: Indoor harsh multipath environment a commercial RFID reader on a static pole.

Increasing the transmission power we notice that more measurements are captured Fig. 25. However, this increase in power is capable of producing more reflections and confuse the algorithms. In addition, due to the reduced capacity of the room, the measurements can not be too many given the fact that we required a distance of 5 cm between the measurements.
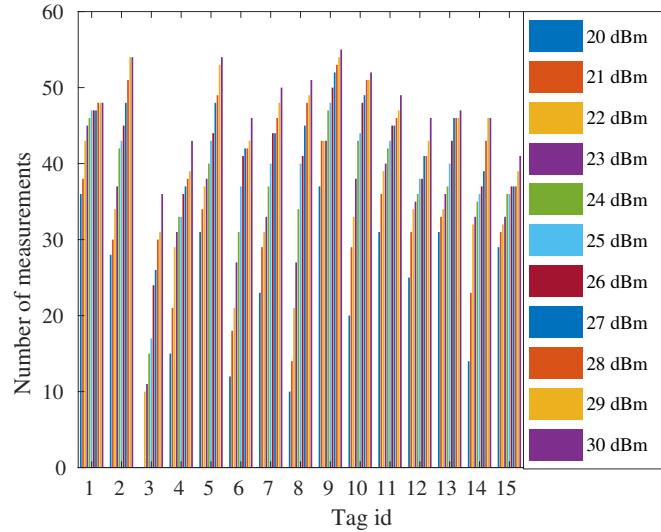

Figure 25: Number of measurements per tag at different power levels - Manual Measurements (2d).

The performance of the algorithms in terms of estimation error is shown in Fig. 26. As expected, the average error for 15 tags seems quite high compared to that of the first campaign, as the reflections generated by the room have not been taken into account. However, we get satisfactory results from the SpaRSA, RSWCLA and DSWCLA methods. Especially, SpaRSA delivers a significant performance at 30 dBm with an error of less than 10 cm in terms of MAE.
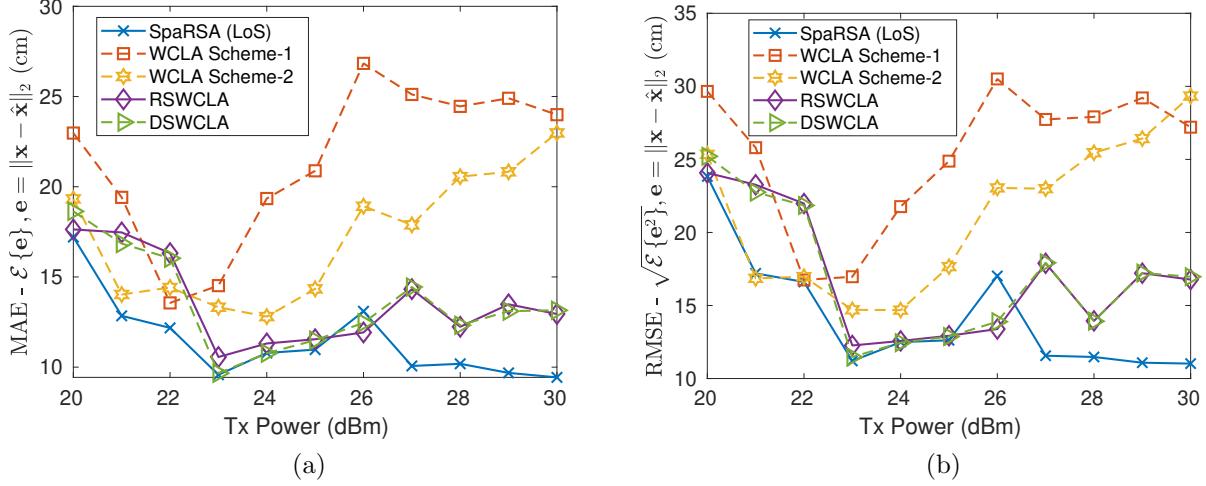
Figure 26: Campaign-2 localization error vs Tx power across 15 tags.

## 5.4 Campaign-3: Kobuki Measurements Lab (2D)
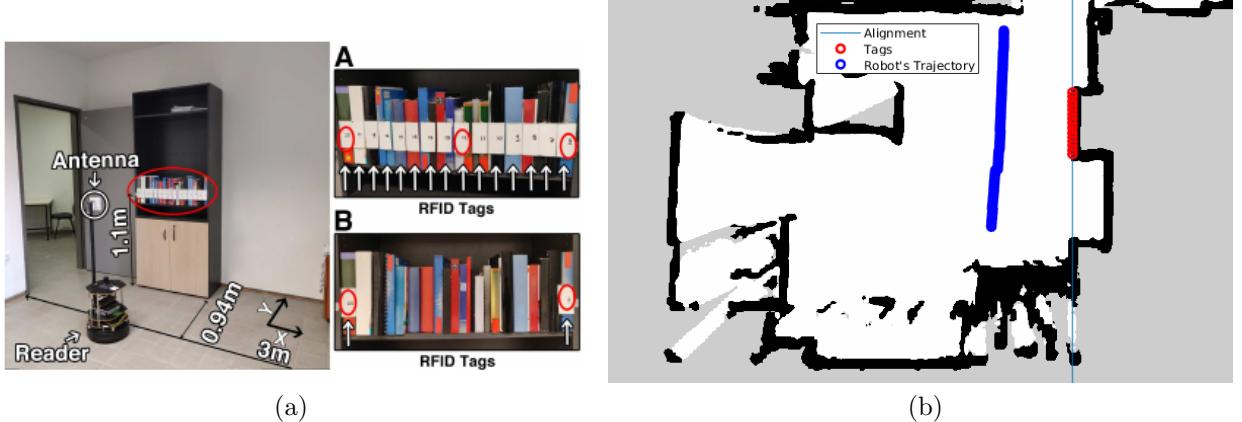
Figure 27: (a) Indoor harsh multipath environment. A commercial RFID reader on an autonomous robotic platform scans and localizes the tags. (b) Real map of the office produced by the Cartographer algorithm. The robot's estimated trajectory from AMCL along with the true tag locations can be seen.

In the third campaign Fig. 27, a custom robotic setup was utilized as in [19]. The robot was a Turtlebot2 with a Kobuki mobile base for motion support, equipped with a Hokuyo UST-20LX LIDAR sensor for SLAM operations. The same reader as in campaigns 1 and 2 was placed upon the robot and measurements were received at 20 and 30 dBm, respectively. The cable and antenna were replaced by smaller counterparts, a 0.74 dB loss coaxial cable, and a FlexiRay SF-2110 5 dBi antenna. The antenna and 15 Alien ALN-9740 (Higgs-4) tags were placed at a lower height of 1.1m further increasing the contribution of multipath. During the experiments, the localization was performed using AMCL [20] and a map of the room was created using Google's Cartographer [21] Fig. 27 (b). The robot's speed was regulated at 10 cm/sec and measurements were received continuously with an average distance between them 0.5 cm.

Given the short inter-element spacing (0.5 cm in average) the number of measurements increased by a very large percentage for each tag compared to campaign 2 Fig. 28.
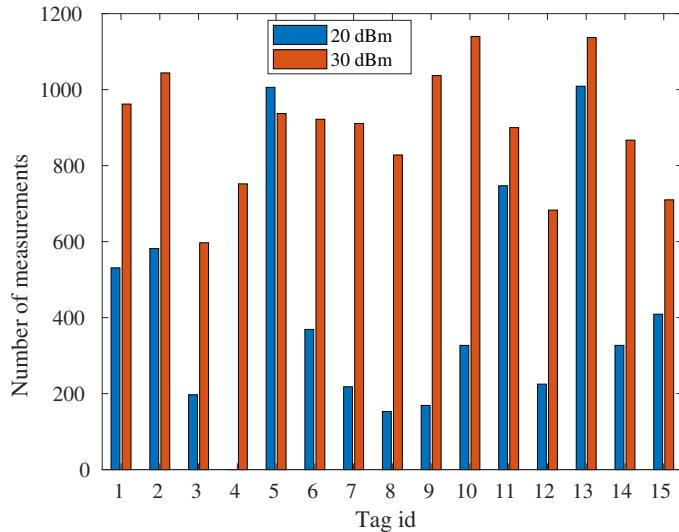

Figure 28: Number of measurements per tag at different power levels of the reader equipped on kobuki (2d).

The results of this campaign in terms of average error across 15 tags are depicted in Fig. 29. What we notice is that the error has increased sharply for all methods compared to the previous 2 campaigns, although the number of measurements has increased. This is probably due to the fact that the robot introduces more error with its movement and position estimation from measurement to measurement. We also observe that the SpaRSA method without modeling a reflector, RSWCLA and WCLA considering the first measurement as a reference, behave much better at 30 dBm antenna emission power compared to 20 dBm. The reason is that at 30 dBm the number of measurements for all tags is much larger; the reader with stronger transmission power can interrogate many times each tag separately.
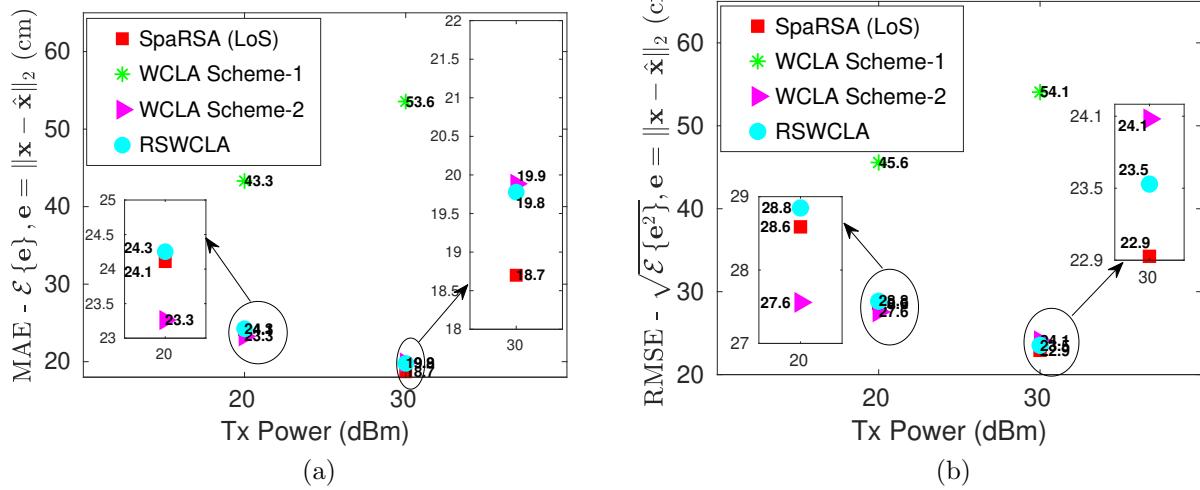
Figure 29: Campaign-3 localization error vs Tx power across 15 tags.

5.5   Campaign-4: Dance Hall (3D)
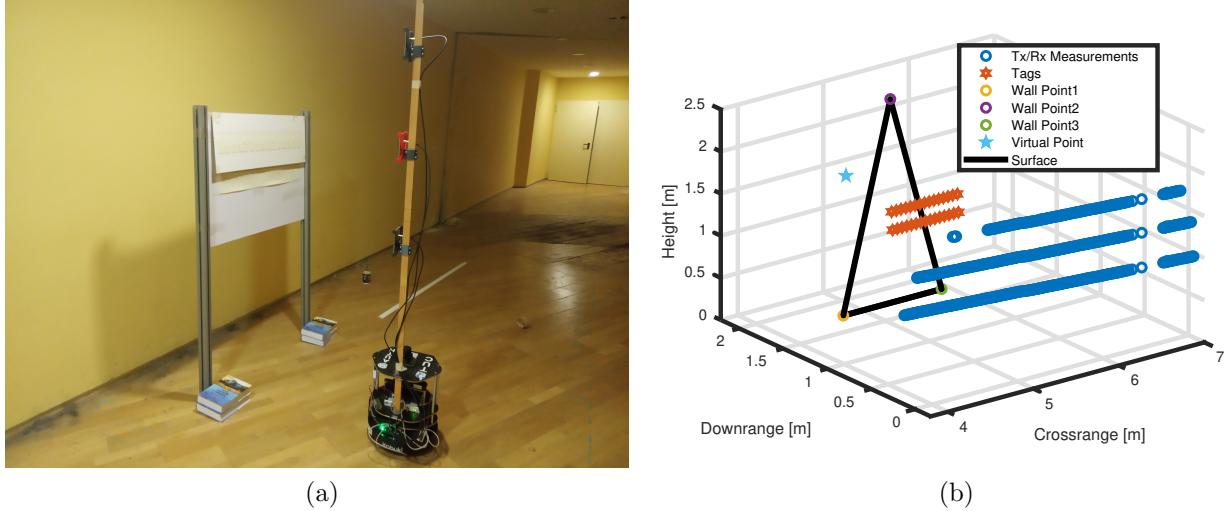
Figure 30: (a) Setup at Dance Hall (TUC). (b) Topology of (a) scenario.
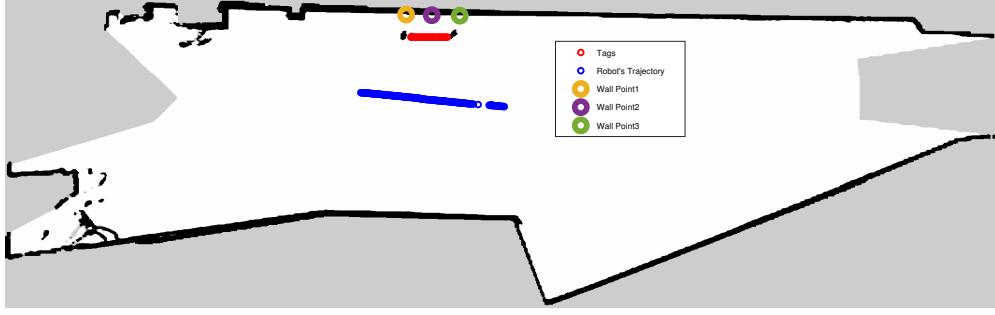
Figure 31: Real map of the dance hall produced by the Cartographer algorithm, including robot's estimated trajectory.

In the final campaign Fig.30, the aforementioned robotic setup was utilized again in a controlled multipath environment in order to test the modeling of a dominant reflector using the group sparse technique and to evaluate the performance of all the methods in 3d. Two rows of 15 Alien ALN-9740 (Higgs-4) tags were placed one below the other with a vertical distance of 22 cm among them. The 30 in total tags were set up between two static poles at a distance of about 45 cm from the wall. The horizontal distance from tag to tag was 5.5 cm. As for the robotic platform, a longer pole was attached so it could accommodate more antennas at larger heights. The number of FlexiRay SF-2110 antennas increased to 3, at heights 1.06, 1.46, and 1.86 m. The same reader was equipped on the robot as in previous campaigns and measurements were received at 20 and 30 dBm, respectively.

For all the methods, a $1 \times 1 \times 1$ grid with resolution $d_x = d_y = d_z = 0.07$ was considered, also the SpaRSA method was used twice. Once without modeling a wall (reflector), and once more modeling the wall behind the tags. The wall is fully described by three points, which were found through the map produced by the robot Fig. 31 assuming some height.

This place (university dance hall) was chosen to fulfill the last campaign due to the large space it provides. This ensures a greater number of measurements Fig. 32, as the robot could cover longer trajectories compared to campaign 3.
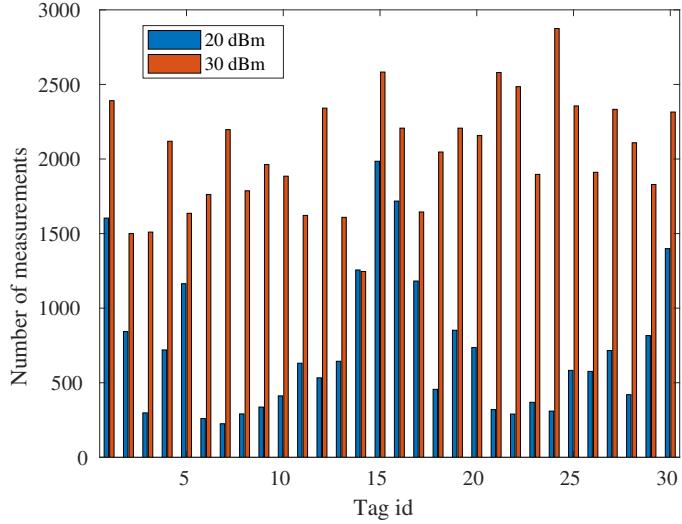
Figure 32: Number of measurements per tag at different power levels of the reader equipped on kobuki (3d).
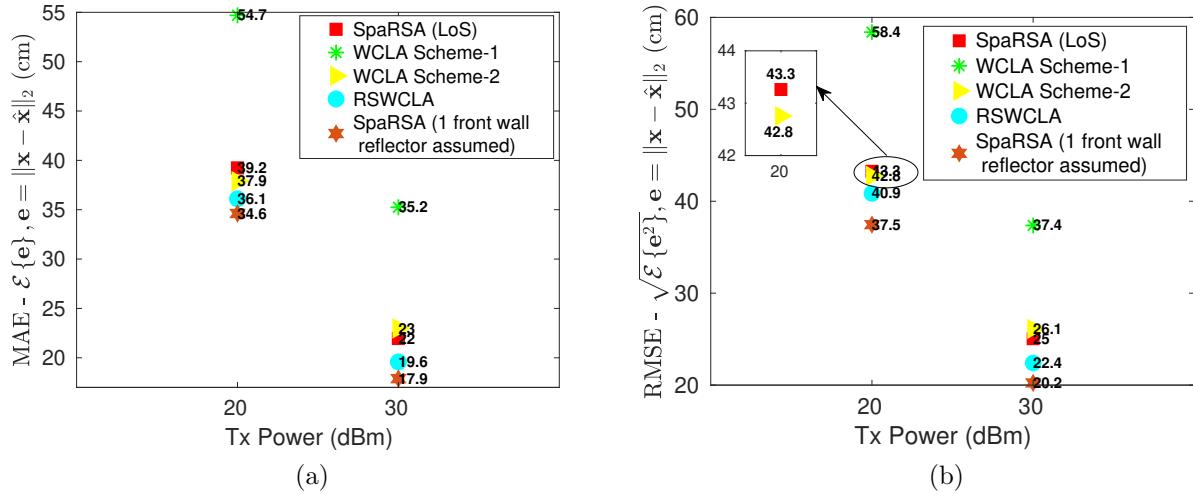
Figure 33: Localization error across 30 tags.

The fact that we have increased the number of antennas in the reader and we have transferred the problem from 2d to 3d we expect to make things difficult for the algorithms as a larger space is required and the complexity increases. Tables 1, 2, 3, 4 offer the average execution time of the most representative algorithms for all the campaigns. Regarding the estimation error Fig. 33, we notice that at 20 dBm all the algorithms have poor performance even SpaRSA considering the wall behind the tags. This, as we mentioned in campaign 3, is expected since at 20 dBm the reader is hardly able to read the tags. However, at 30 dBm

the algorithms seem to do much better with the remarkable performance of SpaRSA, which takes into account the wall behind the tags. This result is a proof of concept that if we take into consideration the reflections that are created inside a room, then we can have better results on average.

Table 1: Execution time in seconds - Campaign 1

Case	SpaRSA (LoS)	WCLA (Scheme-1)	WCLA (Scheme-2)	RSWCLA
20 dBm	0.33	0.23	0.23	1.01
30 dBm	0.37	0.42	0.46	6.08

Table 2: Average execution time in seconds - Campaign 2

Case	SpaRSA (LoS)	WCLA (Scheme-1)	WCLA (Scheme-2)	RSWCLA
20 dBm	0.3	0.2	0.22	4.11
30 dBm	0.38	0.35	0.36	4.66

Table 3: Average execution time in seconds - Campaign 3

Case	SpaRSA (LoS)	WCLA (Scheme-1)	WCLA (Scheme-2)	RSWCLA
20 dBm	3.94	3.24	3.32	28.33
30 dBm	4.81	6.29	6.41	78.58

Table 4: Average execution time in seconds - Campaign 4

Case	SpaRSA (LoS)	SpaRSA (Known Wall Modeling)
20 dBm	0.93	9.25
30 dBm	2.61	15.89

Case	WCLA (Scheme-1)	WCLA (Scheme-2)	RSWCLA
20 dBm	1.73	1.78	21.27
30 dBm	4.83	4.69	125.36

It is worth noting that in the Table 1 we do not refer to average time, since we have only one tag at our disposal.

Chapter 6

Conclusions

6.1 Conclusion

This work provided a review of how multipath affects the localization of passive RFID tags both in simulations and real conditions. Moreover, the results of the CS theory were confirmed, which need to be satisfied to obtain a representative reconstruction of the scene. It is emphasized that Wi-Fi frequencies or long distances between elements are required to ensure a "good" solution. In addition, methods that maximize the MLE appear to be competitive, using appropriately selected weights that mitigate the effect of multipath and thermal noise. Finally, a proof-of-concept is offered, namely, modeling a known wall that is responsible for the existence of multipath can lead to an around 19% improvement, in terms of mean absolute error.

6.2 Future Work

The main objective of this work was to study the effect of multipath in determining the position of commercial tags, under the constraint of a narrowband system. The results are encouraging and indicate that multipath modeling can improve the performance of the algorithms.

There are two important directions that we should follow in the future. First, it is consequential to verify experimentally that increasing frequencies or providing a wider bandwidth is capable of reducing the estimation error. This raises the need of a custom tag that can operate on Wi-Fi frequencies with excessive bandwidth, and the necessity of a USRP illuminator. Last but not least, it is interesting to design an algorithm that will probably solve a non-convex optimization problem, which would jointly estimate the target and the wall that causes the dominant reflection. The latter is a very challenging task, but it could dramatically improve the positioning of passive RFID tags.

Appendices

Operators, Functions and Sets

$\mathbb{N}$	Set of natural numbers
$\mathbb{C}$	Set of complex numbers
$\odot$	Element-wise product
$\mu(\cdot)$	Mutual coherence of a matrix
$\mu_B(\cdot)$	Block-coherence of a matrix
$\rho(\cdot)$	Spectral norm of a matrix, i.e., the maximum singular value
$\mathrm{mod}\,(\cdot)$	Modulo 2π operator
$(\cdot)^{\top}$	Transpose of a vector or matrix
$(\cdot)^{\dagger}$	Conjugate transpose of a vector or matrix
$(\cdot)^{*}$	Complex conjugate of a scalar, vector, or matrix
$|(\cdot)|$	Absolute value of a scalar
$\lceil \cdot \rceil$	Ceil operator, i.e., rounding toward greatest integer
$\|(\cdot)\|_1$	ℓ_1-norm of a vector
$\|(\cdot)\|_2$	Euclidean norm or ℓ_2-norm of a vector
$\|(\cdot)\|_{2,1}$	Mixed ℓ_2/ℓ_1-norm of a vector, i.e., an ℓ_1-norm of groups of ℓ_2-norms
$\|(\cdot)\|_\infty$	Maximum absolute value of an element in a vector

Properties of Modular Arithmetic

- $a \mod n = (a \mod n) \mod n$

- $(a+b) \mod n = [(a \mod n) + (a \mod n)] \mod n$

- $(a-b) \mod n = [(a \mod n) - (a \mod n)] \mod n$

Second Order Taylor Series

We are looking for a quadratic series that approximates the $\cos(\cdot)$ function around the point $x_0 = 0$. In general, the Taylor series of a real or complex-valued function $f(x)$ that is infinitely differentiable at a real or complex number a is the series

$$\sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!}(x-a)^n = f(a) + \frac{f'(a)}{1!}(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \frac{f'''(a)}{3!}(x-a)^3 + \cdots, \quad (75)$$

where $n!$ denotes the factorial of n and $f^{(n)}(a)$ denotes the n-th derivative of f evaluated at the point a (the derivative of order zero of f is defined to be f itself and $(x - a)^0 = 0! = 1$). When $a = 0$, the series is also called a Maclaurin series.

Thus, the Maclaurin for the $\cos(\cdot)$ function is written as follows

$$
\begin{aligned}
\cos(x) &= \sum_{n=0}^{\infty} \frac{\cos^{(n)}(0)}{n!}(x - a)^n \\
&= \cos(0) + \frac{\cos'(0)}{1!}x + \frac{\cos''(0)}{2!}x^2 + \frac{\cos'''(0)}{3!}x^3 + \frac{\cos''''(0)}{4!}x^4 + \dots \\
&= 1 - \sin(0)x - \frac{\cos(0)}{2!}x^2 + \frac{\sin(0)}{3!}x^3 + \frac{\cos(0)}{4!}x^4 + \dots \\
&= 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \dots
\end{aligned}
\tag{76}
$$

When we seek an approximation through second order Taylor series, then we approximate the function with a quadratic polynomial. More specifically, for $\cos(\cdot)$, it holds that

$$
\cos(x) \approx 1 - \frac{x^2}{2!} = 1 - \frac{x^2}{2}.
\tag{77}
$$

Solving for $-x^2$, we easily get that

$$
-x^2 \approx -2\left(1 - \cos(x)\right) = 2\cos(x) - 2.
\tag{78}
$$

Differential Mitigation

The ultimate goal is to prove that

$$
\Delta\varphi_d^{[n,n-1]} = \begin{cases} \mathrm{mod}\left(\frac{4\pi}{\lambda}\Delta d^{[n,n-1]}\right) - 2\pi, & \Delta\varphi_d^{[n,n-1]} \in (-2\pi, 0) \\ \mathrm{mod}\left(\frac{4\pi}{\lambda}\Delta d^{[n,n-1]}\right), & \Delta\varphi_d^{[n,n-1]} \in [0, 2\pi). \end{cases}
$$

Proof. The differences of the adjacent measured phases are

$$
\Delta\varphi_d^{[n,n-1]} = \mathrm{mod}\left(\frac{4\pi}{\lambda}d\,[n] + \phi_0\right) - \mathrm{mod}\left(\frac{4\pi}{\lambda}d\,[n-1] + \phi_0\right).
\tag{79}
$$

Notice that $\Delta\varphi_d^{[n,n-1]} \in (-2\pi, 2\pi)$.

We observe that if $\Delta\varphi_d^{[n,n-1]} \geq 0$ then

$$
\begin{aligned}
\Delta\varphi_d^{[n,n-1]} &= \mathrm{mod}\left(\frac{4\pi}{\lambda}d[n]+\phi_0\right) - \mathrm{mod}\left(\frac{4\pi}{\lambda}d[n-1]+\phi_0\right) \\
&= \mathrm{mod}\left(\mathrm{mod}\left(\frac{4\pi}{\lambda}d[n]+\phi_0\right) - \mathrm{mod}\left(\frac{4\pi}{\lambda}d[n-1]+\phi_0\right)\right) \\
&= \mathrm{mod}\left(\frac{4\pi}{\lambda}d[n]+\phi_0 - \left(\frac{4\pi}{\lambda}d[n-1]+\phi_0\right)\right) \\
&= \mathrm{mod}\left(\frac{4\pi}{\lambda}d[n]+\phi_0 - \frac{4\pi}{\lambda}d[n-1]-\phi_0\right) \\
&= \mathrm{mod}\left(\frac{4\pi}{\lambda}d[n] - \frac{4\pi}{\lambda}d[n-1]\right) \\
&= \mathrm{mod}\left(\frac{4\pi}{\lambda}(d[n]-d[n-1])\right) \\
&= \mathrm{mod}\left(\frac{4\pi}{\lambda}\Delta d^{[n,n-1]}\right).
\end{aligned}
\tag{80}
$$

if $\Delta\varphi_d^{[n,n-1]} < 0$ then

$$
\begin{aligned}
\Delta\varphi_d^{[n,n-1]} &= \mathrm{mod}\left(\frac{4\pi}{\lambda}d[n]+\phi_0\right) - \mathrm{mod}\left(\frac{4\pi}{\lambda}d[n-1]+\phi_0\right) \\
&= \mathrm{mod}\left(\mathrm{mod}\left(\frac{4\pi}{\lambda}d[n]+\phi_0\right) - \mathrm{mod}\left(\frac{4\pi}{\lambda}d[n-1]+\phi_0\right)\right) - 2\pi \\
&= \mathrm{mod}\left(\frac{4\pi}{\lambda}d[n]+\phi_0 - \left(\frac{4\pi}{\lambda}d[n-1]+\phi_0\right)\right) - 2\pi \\
&= \mathrm{mod}\left(\frac{4\pi}{\lambda}d[n]+\phi_0 - \frac{4\pi}{\lambda}d[n-1]-\phi_0\right) - 2\pi \\
&= \mathrm{mod}\left(\frac{4\pi}{\lambda}d[n] - \frac{4\pi}{\lambda}d[n-1]\right) - 2\pi \\
&= \mathrm{mod}\left(\frac{4\pi}{\lambda}(d[n]-d[n-1])\right) - 2\pi \\
&= \mathrm{mod}\left(\frac{4\pi}{\lambda}\Delta d^{[n,n-1]}\right) - 2\pi.
\end{aligned}
\tag{81}
$$

Therefore we proved that

$$
\Delta\varphi_d^{[n,n-1]} = \begin{cases} \mathrm{mod}\left(\frac{4\pi}{\lambda}\Delta d^{[n,n-1]}\right) - 2\pi, & \Delta\varphi_d^{[n,n-1]} \in (-2\pi,0) \\ \mathrm{mod}\left(\frac{4\pi}{\lambda}\Delta d^{[n,n-1]}\right), & \Delta\varphi_d^{[n,n-1]} \in [0,2\pi). \end{cases}
\tag{82}
$$

$\blacksquare$

References

[1] J. Wang and D. Katabi, "Dude, where's my card? rfid positioning that works with multipath and non-line of sight," *SIGCOMM Comput. Commun. Rev.*, vol. 43, no. 4, p. 51–62, Aug. 2013.

[2] F. Manzoor, Y. Huang, and K. Menzel, "Passive rfid-based indoor positioning system, an algorithmic approach," in *IEEE International Conference on RFID-Technology and Applications*, Jun. 2010, pp. 112–117.

[3] S. N. Razavi and C. T. Haas, "Using reference rfid tags for calibrating the estimated locations of construction materials," *Automation in Construction*, vol. 20, no. 6, pp. 677–685, Feb. 2011, selected papers from the 26th ISARC 2009.

[4] M. Leigsnering, F. Ahmad, M. G. Amin, and A. M. Zoubir, "Compressive sensing based specular multipath exploitation for through-the-wall radar imaging," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, Oct. 2013, pp. 6004–6008.

[5] C. Balanis, *Advanced Engineering Electromagnetics*. New York: NY: John Wiley and Sons, 1989.

[6] S. J. Wright, R. D. Nowak, and M. A. T. Figueiredo, "Sparse reconstruction by separable approximation," *IEEE Trans. Signal Process.*, vol. 57, no. 7, pp. 2479–2493, Jul. 2009.

[7] W. Deng, W. Yin, and Y. Zhang, "Group sparse optimization by alternating direction method," in *Wavelets and Sparsity XV*, D. V. D. Ville, V. K. Goyal, and M. Papadakis, Eds., vol. 8858, International Society for Optics and Photonics. SPIE, Sep. 2013, pp. 242 – 256.

[8] J. Jacques Fuchs, "More on sparse representations in arbitrary bases," *IFAC Proceedings Volumes*, vol. 36, no. 16, pp. 1315–1320, 2003.

[9] S.-J. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinevsky, "An interior-point method for large-scale ℓ_1-regularized least squares," *IEEE J. Sel. Topics Signal Process.*, vol. 1, no. 4, pp. 606–617, Dec. 2007.

[10] E. Candes and T. Tao, "Decoding by linear programming," *IEEE Trans. Inf. Theory*, vol. 51, no. 12, pp. 4203–4215, Dec. 2005.

[11] E. J. Candès, J. K. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Communications on Pure and Applied Mathematics*, vol. 59, pp. 1207–1223, 2005.

[12] D. Donoho and X. Huo, "Uncertainty principles and ideal atomic decomposition," *IEEE Trans. Inf. Theory*, vol. 47, no. 7, pp. 2845–2862, Nov. 2001.

[13] J. Tropp, "Greed is good: algorithmic results for sparse approximation," *IEEE Trans. Inf. Theory*, vol. 50, no. 10, pp. 2231–2242, Oct. 2004.

[14] D. Donoho, M. Elad, and V. Temlyakov, "Stable recovery of sparse overcomplete representations in the presence of noise," *IEEE Trans. Inf. Theory*, vol. 52, no. 1, pp. 6–18, Jan. 2006.

[15] E. J. Candes and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?" *IEEE Trans. Inf. Theory*, vol. 52, no. 12, pp. 5406–5425, Dec. 2006.

[16] Y. C. Eldar, P. Kuppinger, and H. Bolcskei, "Block-sparse signals: Uncertainty relations and efficient recovery," *IEEE Trans. Signal Process.*, vol. 58, no. 6, pp. 3042–3054, Jun. 2010.

[17] Y. C. Eldar and H. Bolcskei, "Block-sparsity: Coherence and efficient recovery," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2009, pp. 2885–2888.

[18] C. Li, E. Tanghe, D. Plets, P. Suanet, N. Podevijn, J. Hoebeke, E. D. Poorter, L. Martens, and W. Joseph, "Phase-based variant maximum likelihood positioning for passive uhf-rfid tags," in *14th European Conference on Antennas and Propagation (EuCAP)*, Mar. 2020, pp. 1–5.

[19] E. Giannelos, E. Andrianakis, K. Skyvalakis, A. G. Dimitriou, and A. Bletsas, "Robust rfid localization in multipath with phase-based particle filtering and a mobile robot," *Proc. IEEE Int. Conf. on RFID*, vol. 5, no. 3, pp. 302–310, Sep. 2021.

[20] D. Fox, "Kld-sampling: Adaptive particle filters." in *Proc. Int. Conf. on Neural Information Processing Systems: Natural and Synthetic (NIPS)*, 01 2001, pp. 713–720.

[21] W. Hess, D. Kohler, H. Rapp, and D. Andor, "Real-time loop closure in 2d lidar slam," in *IEEE International Conference on Robotics and Automation (ICRA)*, May 2016, pp. 1271–1278.

[22] A. Goldsmith, *Wireless Communications*. New York, NY, USA: Cambridge University Press, 2005.

[23] M. Leigsnering, *Sparsity-Based Multipath Exploitation for Through-the-Wall Radar Imaging*. Springer, Cham, 2018.