# Various Positioning Algorithms based on Received Signal Strength and/or Time/Direction (Difference) of Arrival for 2D and 3D Scenarios

Minh Hoang Le

# Various Positioning Algorithms based on Received Signal Strength and/or Time/Direction (Difference) of Arrival for 2D and 3D Scenarios

Dissertation

*submitted to*

Sorbonne Université

*in partial fulfillment of the requirements for the degree of Doctor of Philosophy*

*Author:*

**Minh Hoang LE**

*Scheduled for defense on 31 JANUARY 2022, before a committee composed of:*

*Reviewers*

| | | |
|---|---|---|
| **Prof.** | **Luc DENEIRE** | Université Côte d'Azur, France |
| **Prof.** | **Nel SAMAMA** | Télécom SudParis, France |

*Examiners*

| | | |
|---|---|---|
| **Prof.** | **Dirk SLOCK** | EURECOM, France |
| **Dr.** | **Jean-Pierre ROSSI** | Orange Labs, France |
| **Prof.** | **Jérôme HÄRRI** | EURECOM, France |
| **Dr.** | **Maria-Joao RENDAS** | I3S, Sophia Antipolis, France |




# Divers Algorithmes de Positionnement basés sur l'Intensité du Signal Reçu et/ou (la Différence de) Temps/Direction d'Arrivée pour les Scénarios 2D et 3D

Thèse

*soumise à*

Sorbonne Université

*pour obtenir le grade de*
*Doctor Philosophiæ*

*Auteur:*

**Minh Hoang LE**

*Soutenue le 31 JANVIER 2022, devant un comité composé de :*

*Rapporteurs*

| | | |
|---|---|---|
| **Prof.** | **Luc DENEIRE** | Université Côte d'Azur, France |
| **Prof.** | **Nel SAMAMA** | Télécom SudParis, France |

*Examinateurs*

| | | |
|---|---|---|
| **Prof.** | **Dirk SLOCK** | EURECOM, France |
| **Dr.** | **Jean-Pierre ROSSI** | Orange Labs, France |
| **Prof.** | **Jérôme HÄRRI** | EURECOM, France |
| **Dr.** | **Maria-Joao RENDAS** | I3S, Sophia Antipolis, France |

*If you want to have what you've never had, you must try to do what you've never done...*

# Abstract

LOCALIZATION has fascinated researchers for centuries and has motivated a large amount of studies and developments in communications.

The aim of positioning problems is to determine the position of the mobile device. Positioning algorithms can be divided into 3 methods: Trilateration, Multilateration and Triangulation.

Trilateration utilizes the distances between the mobile device and all the base stations around to estimate the mobile position. These distances can be estimated via the Time of Arrival (ToA) or the Received Signal Strength (RSS).

In Multilateration, the position location is based on measured quantities whose values are a function of the Time Difference of Arrival (TDoA) of the two ToAs.

As for Triangulation, the directions of the incident signals play the most crucial role in the localization. Therefore, it is also referred to as Direction-based Localization. The Direction of Arrival (DoA) of each incident wave is taken into account to solve the positioning problem. Each DoA is expressed by a single angle in 2D scenarios, and a pair of angles in 3D scenarios. There are noticeable differences between Network-Positioning implemented at the Network of Base Stations and Self-Positioning implemented at the Mobile Device. In Network-Positioning, the mobile device is directly localized based on the DoAs of the incident signals; meanwhile, in Self-Positioning, its position is estimated by the Direction Difference of Arrival (DDoA) between each pair of incident signals, because the DoA of each signal arriving to the Mobile Device is ambiguous.

In this dissertation, we study all the localization approaches described above. Our spotlight is for Triangulation, which has many sub-scenarios to analyze. The results are obtained by MATLAB simulations.

# Résumé

LOCALISATION a fasciné les chercheurs pendant des siècles et a motivé un grand nombre d'études et de développements dans dans le domaine des communications.

L'objectif du positionnement est de localiser le dispositif mobile. Les algorithmes de positionnement peuvent être divisés en 3 méthodes : Trilatération, Multilatération et Triangulation.

La trilatération utilise les distances entre le dispositif mobile et toutes les stations de base environnantes pour estimer la position du mobile. Ces distances peuvent être estimées par le Temps d'Arrivée (Time of Arrival - ToA) ou l'Intensité du Signal Reçu (Received Siganl Strength - RSS).

Dans la multilatération, la localisation est basée sur des quantités mesurées dont les valeurs sont une fonction de la Différence de Temps d'Arrivée (Time Difference of Arrival - TDoA) des deux ToAs.

La Triangulation, quand à elle, utilise les directions des signaux incidents. Par conséquent, elle est également appelée localisation basée sur la direction. La Direction d'Arrivée (Direction of Arrival - DoA) de chaque onde incidente est comptée pour résoudre le problème de positionnement. Chaque DoA est exprimée par un angle dans les scénarios 2D, et par une paire d'angles dans les scénarios 3D. Il existe des différences notables entre le positionnement par le réseau mis en œuvre dans le réseau de stations de base et l'autopositionnement mis en œuvre dans le dispositif mobile. Dans le positionnement par le réseau, le dispositif mobile est directement localisé sur la base des DoA des signaux incidents ; tandis que dans l'autopositionnement, sa position est estimée par la Différence de Direction d'Arrivée (Direction Difference of Arrival - DDoA) entre chaque paire de signaux incidents, car la DoA de chaque signal arrivant au dispositif mobile est ambiguë.

Dans cette thèse, nous étudions toutes les approches de localisation ci-dessus. Nous nous focalisons plus particulièrement sur la Triangulation, qui comporte de nombreux sous-scénarios à analyser. Les résultats sont obtenus par des simulations MATLAB.

# Acknowledgements

This dissertation is submitted to Sorbonne University. This work is funded by French government FUI project GEOLOC.

Firstly, I would like to express my deep and sincere gratitude to my supervisors, Dirk SLOCK from EURECOM and Jean-Pierre ROSSI from Orange Labs. Their invaluable advice has been broadened my knowledge, given me the correct directions so that I can complete my dissertation. In addition, my thankfulness is for Florian KALTENBERGER and Mohsen AMADI (EURECOM). Besides, my gratefulness is also for Fabien FERRERO (Laboratoire d'Electronique, Antennes et Télécommunications - LEAT) and Le-Huy TRINH (Ho Chi Minh City University of Information Technology), who introduced me to this PhD thesis in 2018.

Furthermore, I would like to express my thankfulness to my parents and my sister, who are always beside me in the most difficult time and encourage me. My best appreciations are also for my best friends who work and study in France, for always being an important part of my personal life.

Last but not least, my colleagues in Orange Labs and EURECOM have been giving considerable helps during three year, which is unforgettable to me. I thank them all.

Minh Hoang LE

Nice - France, January 2022

# Contents

# List of Figures

# List of Tables

# Acronyms and Abbreviations

The acronyms and abbreviations used throughout the manuscript are specified in the following. They are presented here in their singular form, and their plural forms are constructed by adding an *s*, e.g. TX (transmitter) and TXs (transmitters). The meaning of an acronym is also indicated the first time that it is used.

| | |
|---|---|
| AoA | Angle of Arrival. |
| AoD | Angle of Departure. |
| app. | Appendix. |
| AVE | Average Position Error. |
| AWGN | Additive White Gaussian Noise. |
| BC | Broadcast Channel. |
| BS | Base Station. |
| BE | Bayesian Estimation. |
| CB | Coordinated Beamforming. |
| CDF | cumulative density function. |
| CSI | Channel State Information. |
| CSIT | Channel State Information at the Transmitter. |
| CSIR | Channel State Information at the Receiver. |
| CIR | Channel Impulse Response. |
| CRB | Cramer-Rao Bound. |
| DDoA | Direction Difference of Arrival. |
| DL | Downlink. |
| DoA | Direction of Arrival. |
| DoF | Degrees of Freedom. |
| ESNR | Signal to estimation noise ratio. |
| eq. | Equation. |
| FDD | Frequency Division Duplexing. |
| gNB | Next Generation Node B. |
| GNSS | Global Navigation Satellite System. |
| i.i.d. | independent and identically distributed. |
| IC | Interference Channel. |
| KF | Kalman Filter. |
| LHS | left hand side. |
| LE | Location Estimation. |

| | |
|---|---|
| LoS | Line Of Sight. |
| LTE | Long Term Evolution. |
| LS | Least Squares. |
| MAP | Maximum a Posteriori. |
| MIMO | Multiple Input Multiple Output. |
| MISO | Multiple Input Single Output. |
| ML | Maximum Likelihood. |
| NID | Network Interface Device. |
| NLoS | Non Line of Sight. |
| NR | New Radio. |
| OFDM | Orthogonal Frequency Division Multiplexing. |
| PDF | probability density function. |
| PL | Position Location. |
| PMF | probability mass function. |
| PRS | Positioning Reference Signals. |
| PSS | Primary Synchronization Signal. |
| Rel-16 | Release 16. |
| RHS | right hand side. |
| RMSE | Root Mean Square Error. |
| RSS | Received Signal Strength. |
| RVQ | Random Vector Quantization. |
| RX | Receiver(s). |
| RZF | Regularized Zero-Forcing. |
| SBR | Shooting and Bouncing Ray. |
| SIMO | Single Input Multiple Output. |
| SINR | Signal to Noise and Interference Ratio. |
| SNR | Signal-to-Noise Ratio. |
| SISO | Single-Input Single-Output. |
| SSB | Synchronization Signals Block. |
| SSS | Secondary Synchronization Signal. |
| TDL | Tapped Delay Line. |
| TDoA | Time Differences of Arrival. |
| ToA | Time(s) of Arrival. |
| TLS | Total Least Squares. |
| TRP | Transmission and Reception Point. |
| TX | Transmitter(s). |
| UE | User Equipment. |
| UL | Uplink. |
| WLS | Weighted Least Squares. |
| w.r.t. | with respect to. |
| w.l.o.g. | Without loss of generality. |
| ZF | Zero Forcing. |
| 2D (3D, 4D) | two (three, four) dimensional. |
| 3GPP | 3rd Generation Partnership Project. |

# Notations

The next list describes an overview on the notation used throughout this manuscript. We use boldface uppercase letters ($\boldsymbol{X}$) for matrices, boldface lowercase letters for vectors ($\boldsymbol{x}$), and regular lowercase letters for scalars ($x$). The vector can be represented by a regular lowercase accented by a right arrow $\left(\overrightarrow{d}\right)$.

| | |
|---|---|
| $\boldsymbol{A}^{\dagger}$ | Moore-Penrose pseudo inverse of matrix $\boldsymbol{A}$. |
| $\mathbf{exp}(a)$ | denotes an exponentiation with the base $e$ and the power $a$. |
| $\mathbf{diag}(a_1, a_2, \ldots, a_n)$ | the diagonal matrix whose diagonal elements are $a_1$, $a_2$, …, $a_n$. |
| $\mathbf{log}(\cdot)$ | stands for decimal logarithm. |
| $\mathbf{ln}(\cdot)$ | stands for natural logarithm. |
| $\mathbf{E}(\cdot)$ | stands for mean value. |
| $\mathbf{var}(\cdot)$ | stands for variance. |
| $x_i$ | abscissa ($x$-coordinate) of the $i$-th base station. |
| $y_i$ | ordinate ($y$-coordinate) of the $i$-th base station. |
| $z_i$ | applicate ($z$-coordinate) of the $i$-th base station. |
| $x$ | abscissa ($x$-coordinate) of the mobile device. |
| $y$ | ordinate ($y$-coordinate) of the mobile device. |
| $z$ | applicate ($z$-coordinate) of the mobile device. |
| $\boldsymbol{x}$ | coordinate vector of the mobile device. In 2D problem, $\boldsymbol{x} = [x \;\; y]^T$. In 3D problem, $\boldsymbol{x} = [x \;\; y \;\; z]^T$. |
| $\boldsymbol{\varphi}_i$ | In 2D problem, it represents the DoA of the signal related to the $i$-th base station. In 3D problem, it represents the azimuth angle of the signal related to the $i$-th base station. |
| $\theta_i$ | In 3D problem, it represents the elevation angle of the signal related to the $i$-th base station. |
| $\phi_{i,j}$ | In 2D self-positioning problem, it represents the DDoA between the signals coming from the $i$-th base station and $j$-th base station. |
| $\beta_{i,j}$ | In 3D self-positioning problem, it represents the DDoA between the signals coming from the $i$-th base station and $j$-th base station. |
| $\gamma_{i,j}$ | In 3D self-positioning problem, $\gamma_{i,j} = \cos\beta_{i,j}$. |

In addition, we define some following less common mathematical functions:

$$\text{sign}(x) = \begin{cases} 1 & , \ x \geq 0 \\ -1 & , \ x < 0 \ . \end{cases} \tag{1}$$

Then for $(x, y) \neq (0, 0)$, we have the definition of atan2 function

$$\text{atan2}(y, x) = \arctan\left(\frac{y}{x}\right) - (\text{sign}(x) - 1)\,\text{sign}(y)\,\frac{\pi}{2}\,. \tag{2}$$

With $(a, b) \in \mathbb{R}^2$, the modulo operation is defined as:

$$\text{mod}(a, b) = r \iff a = bq + r \tag{3}$$

where $q \in \mathbb{Z}$ and $0 \leq r < b$

# Chapter 1

# Introduction

## 1.1 Motivations

The primary goal of position location (PL) is to find or estimate the location of a point of interest in a two-dimensional (2D) or three-dimensional (3D) space inside a coordinate system established using certain known references. A new position is determined in general location scenarios and at the site of interest, considering the displacement from a previously known reference location. This could indicate some inertial and direction estimation. We evaluate gadgets with transmission or reception capabilities that help with the location procedure due to the widespread use of wireless technologies.

### 1.1.1 The demand for position location in history [1]

For millennia, scientists have been interested in the art of determining position. The first positioning methods were probably devised several millennia ago, when people realized how important it was to know where they were in order to travel in a systematic manner. Orientation at natural landmarks such as mountains, rivers, or coasts is a simple approach for accomplishing this goal. Early man-made monuments were paths and ways that were often established for trading, such as the famed Silk Road, which united Europe and Eastern Asia and dated back to roughly 500 B.C. Lighthouses are another man-made landmark. They give orientation in repetitive areas, including at night, for ships near to the coast, for example. On the other hand, landmarks are absent on the high seas. The dead reckoning method, which involves gauging direction and velocity to keep track of a journey, was the simple method employed by early ocean navigators. Another method that uses well-known objects as position references is celestial navigation. The latitude is calculated by measuring the angle of the pole star above the horizon. The determination of longitude, which is directly tied to the precise measurement of time due to the Earth's rotation, has been a key concern for a long time.

Famous scientists such as Isaac Newton and Edmond Halley proposed and encouraged astronomical procedures, or predictable astronomical phenomena, to determine time. Such concepts include the 'lunar distance' from a fixed star or the ecliptic of Jupiter's moons. The problem was overcome when accurate chronometers were invented, obviating

the necessity for astronomical methods. During a five-week cruise to Jamaica in 1761, John Harrison's H.4 marine chronometer, built in 1759, showed a time inaccuracy of 5 seconds. All approaches that rely on visual observations at least in part necessitate clear vision. This restricts the applicability of these techniques to specific periods of the day or favorable weather conditions. The late-nineteenth-century discovery of radio waves ushered in the discipline of radio navigation.

### 1.1.2 Literature Review

In order to determine the position of a mobile device, a network of base stations, whose positions are already known, is required. In localization processes, the base stations can be the transmitters (TXs) which transmit the signals to the mobile device, which is the receiver (RX), or vice versa. The essential techniques of positioning systems are described in this section. Various positioning systems are built on different combinations of these techniques. The authors of [2] have summarized the following techniques:

#### Received Signal Strength (RSS)

RSS estimation enables for range or distance measurement, allowing for localization. To locate a mobile device, multiple base stations work together. The strength of the received signal represents the signal's travel distance in this approach [3]. Although RSS systems are very sensitive to shadowing and to NLoS scenarios, it still plays a crucial role in localization because it requires the simplest hardware. For a coplanar example, at least three base stations with three RSS measurements are required, given that the transmission strength and channel (or environment in which the signal is moving) parameters are known.

The position of the mobile device is estimated within a circle of radius $d_i$, with the base station $\mathbf{BS}_i$ at the center of the circle, using distance measurements. This position can be determined by either base stations or the mobile device itself (see Fig. 1.1). At least four base stations are required in non-coplanar cases.

#### Time of Arrival (ToA)

In ToA, similar to the RSS, many base nodes collaborate to locate a target node, (see Fig. 1.1). Instead of measuring RSS at the receivers, the ToA is used to make the position estimation. In some cases, numerous base stations may work together to determine their own position before attempting to locate the mobile device without using GPS. With particular, in inhomogeneous medium, the ToA estimation method would be complicated [4, 5]. A good example of such a medium is the human body.

ToA appears to be a reliable technique, on the other hand, it has several weaknesses [6]:

(a) It necessitates the perfect synchronization of all nodes (base stations and mobile devices): a little timing error might result in a substantial error in the estimation of the distance $d_i$.

(b) To allow the receiver to know the moment at which the signal was generated at the target node, the transmitted signal must be labeled with a timestamp. This extra timestamp adds to the complexity of the transmitted signal, perhaps creating another source of error.

**Time Difference of Arrival (TDoA)**

TDoA estimate necessitates the measurement of the time difference between signals arriving at two receivers, as the name indicates. This approach, like ToA estimation, presupposes that base station placements are known [6]. A hyperbola is used to show the difference in ToA at each receiver. This hyperbola is the locus of a point in a plane such that the difference of distances from two fixed points (known as the foci) remains constant.

To locate a mobile device in a coplanar situation, three base stations with two TDoA measurements are required (see Fig. 1.2). One base station is designated as the reference base station and is numbered as $\mathbf{BS}_1$ in the diagram. TDoA measurements are taken with the reference base station in mind. Non-coplanar scenarios need the placement of four base stations and three TDoA measurements.

TDoA overcomes ToA's first weakness by eliminating the need to synchronize the mobile device's clock with the base station's clock. All base stations receive the same signal provided by the mobile device in uplink TDoA. As a result, the inaccuracy in the arrival time at each base node owing to unsynchronized clocks is the same as long as the base stations' clocks are synchronized. In the case of downlink TDoA, the synchronization issue is completely eliminated.

ToA is the time interval between the start time ($t_s$) of the signal at the transmitter and its end time($t_i$) at the receiver. TDoA, on the other hand, is the time difference between the end timings ($t_i$ and $t_j$) of the broadcast signal associated to two base stations $\mathbf{BS}_i$ and $\mathbf{BS}_j$, as illustrated in Figure 1.2. As a result, with the TDoA approach, there is no need to synchronize the clocks of the mobile device and the base stations. It significantly decreases the complexity of implementations.

**Direction of Arrival (DoA)**

In DoA estimation, the receivers determine the angle of the incident signals (Fig. 1.3). Base stations should be equipped with antenna arrays, and each antenna array should have radio-frequency (RF) front-end components to allow them to estimate DoA. However, this comes at a higher price, with more complexity and power consumption. The DoA estimate is becoming increasingly exact with 5G technology. As a result, this technology is quite promising and has the potential to be used for future localization.

In contrast to ToA and TDoA, only two base stations and two DoA measurements are required in both coplanar and non-coplanar scenario. The primary lobe of an antenna array is directed in the direction of the incident signal's highest incoming energy to identify the DoA [7].

**Direction Difference of Arrival (DDoA)**

In DoA estimation, the orientation of the receiver must be fixed. Therefore, DoA-based localization is feasible for Network-Positioning. However, when the receiver is the mobile device, DoA estimation of the incident signal is a really challenging task, which convolutes the localization. In this situation, DDoA is a suitable approach to tackle the direction estimation problem [8, 9] .

Similar to TDoA, DDoA measures the difference in directions of incident waves. The DDoA-based positioning algorithms are more complex than the DoA-based ones in general, and their accuracy is also usually lower. Nonetheless, this method is still crucial in practical localization.

**Hybrid localization**

In some situations, the mobile device is localized by two or more methods combined. This cooperation is expected to give more accurate position estimations.

## 1.2 Methodology

Localization requires a mobile device whose position has to be determined and a network of base stations whose positions are fixed and already known. Based on where the position is calculated, there are two approaches for positioning determination [10]:

- **Self-Positioning:** Signals are transmitted from the network base stations and received by the mobile device. The position is then estimated at the device to localize itself.
- **Network-Positioning:** A signal is transmitted by the mobile device and received from the network of base stations. The position is calculated at a unit in the network of base stations.

Positioning algorithms can be divided into three main methods:

- **Trilateration:** Position location is determined based on the distances from the mobile device to the base stations. These distances can be estimated by Time of Arrival (ToA) or Received Signal Strength (RSS) of the transmitted signals. Chapter 2 details this type of algorithm.
- **Multilateration:** The differences in distances are used to determine position. The Time Difference of Arrival (TDoA) approach is commonly used to estimate these differences. The distance difference between two stations leads in an infinite number of potential subject locations that match the TDoA. When these potential sites are mapped, a hyperbolic curve emerges. Multilateration uses many TDoAs to pinpoint the exact location of the subject along that curve. A second TDoA involving a different pair of stations (usually one station is common to both pairings, so only one station is new) will yield a second curve that intersects the first in two

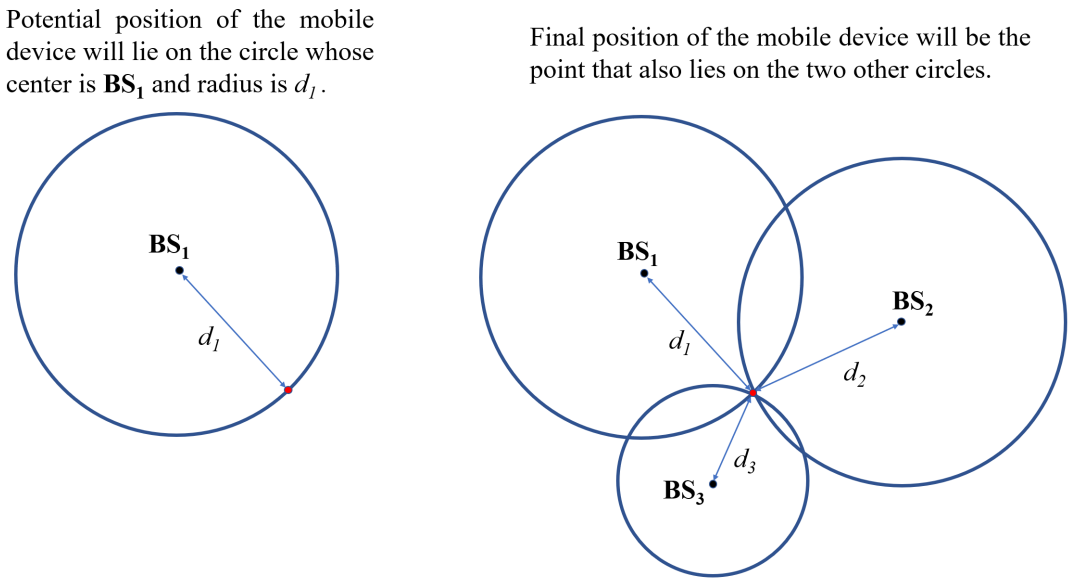



Figure 1.1: Operation of Trilateration in 2D

dimensions. When the two curves are compared, only a few viable user locations (usually two) emerge. Chapter 3 details this type of algorithm.

- **Triangulation:** Literally, triangulation is the process of determining the location of a point by forming triangles to the point from known points. To form the necessary triangles, the Direction of Arrival (DoA) of the signals will be required. Consequently, triangulation is referred as to direction-based positioning algorithm.

  Unlike trilateration and multilateration, triangulation has different algorithms between network-positioning and self-positioning. The DoA expressions are different at the mobile device and at the base station. Furthermore, the orientation of the mobile device is ambiguous and probably inconstant. Therefore, self-positioning triangulation and network-positioning triangulation are generally contrasting. They must be particularly analyzed in distinct chapters. Chapters 4, 5, 6 demonstrate the studies for each sub-case of triangulation.

Furthermore, the cooperation of two or more methods above is also considered in this thesis.

## 1.3 Contributions

In this dissertation, we address all the problems mentionned in the previous section in order to make the localization processes more efficient. In particular,

**Chapter 2.** In Chapter 2, we start with Trilateration, the basic method of localization. We study the geometric approach by forming the concerned circles and finding their

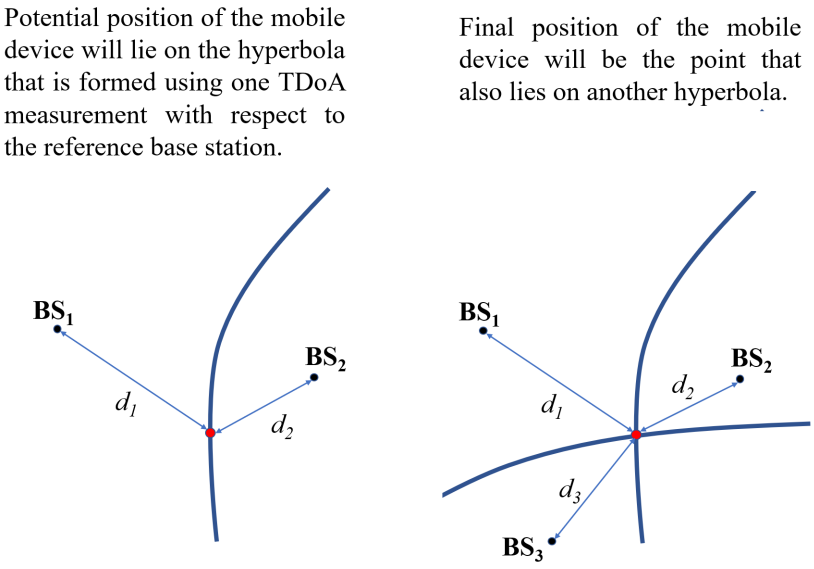



Figure 1.2: Operation of Multilateration in 2D

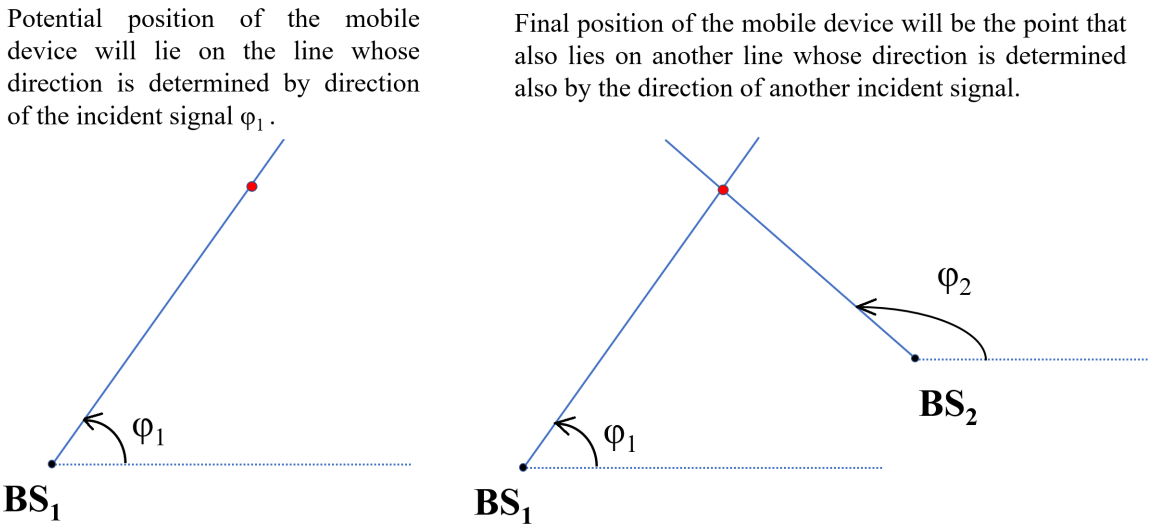



Figure 1.3: Operation of Triangulation in 2D

intersection point. In practical scenarios where noises always affect distance estimations, such a point may not exist. Consequently, another geometric solution is proposed to tackle this problem, regardless of the existence of the intersection point. An interpretation using matrix is presented, which simplifies the solution.

**Chapter 3.** In Chapter 3, we tackle the localization problem in Multilateration. TDoA estimation in multi path channels is investigated. Least-Squares method and Maximum Likelihood estimator are also proposed. Practice-like simulations are implemented.

**Chapter 4.** From this chapter, the direction-based algorithms (Triangulation) are considered. The localization problems at base stations are analyzed in both 2D and 3D scenarios. We introduce a new definition for azimuth angle estimation, using atan2 function and phase jump corrections. This improves considerably the accuracy of localization.

**Chapter 5.** In Chapter 5, we focus on Self-Positioning Triangulation in 2D scenarios. As the DoA estimation of the incident wave to the mobile device is impossible to be correctly estimated, the position algorithm based on Direction Difference of Arrival (DDoA) is suggested. This chapter focus on the positioning problem in 2D scenarios, where the DDoA is easily obtained by subtracting two concerned DoAs.

**Chapter 6.** In Chapter 6, we expand the problem, which is solved in chapter 5, into 3D scenarios. Computing DDoA from the related DoAs is much more challenging. In addition, it is not straightforward to determine the mobile position from the DDoAs obtained. Two methods are considered. One is localization based on pure DDoA, another is the ToA-DDoA hybrid localization. In both methods, Least Squares and Maximum Likelihood estimator are studied.

**Chapter 7.** In Chapter 7, we conclude the dissertation, and discuss about possible further developments.

## 1.4 Publications

Some publications related to this dissertation:

**Papers already published**

- H. M. Le, D. Slock, J-P. Rossi, "A geometric interpretation of trilateration for RSS-based localization", *28th European Signal Processing Conference (EUSIPCO)*, 2020.
- H. M. Le, D. Slock, J-P. Rossi, "2D DDoA-based self-positioning for mobile devices", *29th European Signal Processing Conference (EUSIPCO)*, 2021.
- H. M. Le, D. Slock, J-P. Rossi, "3D DoA-based localization with phase jump corrections", *IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT)*, 2021.
- H. M. Le, D. Slock, J-P. Rossi, "2D DoA-based Positioning with Phase Jump Corrections and An Approximate Maximum Likelihood Estimator", *3rd IEEE Middle East and North Africa COMMunications Conference (MENACOMM)*, 2021.

- H. M. Le, D. Slock, J-P. Rossi, "3D Self-Positioning Algorithm at Mobile Devices based on joint DDoA-ToA", *3rd IEEE Middle East and North Africa COMMunications Conference (MENACOMM)*, 2021.

**Pending paper**

- H. M. Le, M. Ahadi, D. Slock, F. Kaltenberger, J-P. Rossi, "3D Localization in 5G New Radio Networks Using Downlink Time Difference of Arrival", *European Conference on Networks and Communications (EuCNC) and 6G Summit*, 2022.

# Chapter 2

# RSS-based Trilateration in 2D

Trilateration is a popular approach in localization. Many related geometric approaches have been proposed for 2D scenarios. In general, each approach has a standard case in which the main solution is applied, and many specific cases. Each specific case has a particular solution, which makes the algorithm more complex. This chapter introduces a novel geometric approach that covers all the cases considered by previous algorithms. It turns out though that this approach is a special case of an existing approach, for which we hence provide a geometric interpretation. Numerical results illustrate the method in RSS-based localization while estimating simultaneously the path loss exponent.

## 2.1 Introduction

In Trilateration, we calculate the distances from a mobile station and the base stations around and use them to deduce its position. For ToA, the distance is estimated based on the difference between the time instants when the signal was transmitted and when the signal was received. For RSS, we need more parameters to estimate this distance. Beside the ratio between transmitted signal power and received signal power, we also need path loss exponent, a parameter showing the reduction in power density of an electromagnetic wave as it propagates through space. In a 2D model, the received signals from at least 3 base stations are required to estimate position of a mobile station.

RSS-based localization with simultaneous path loss estimation is the subject of several previous research papers [11–14].

This chapter proposes a new trilateration algorithm that uses 3 base stations of known positions to locate one mobile station in 2D scenarios, by a geometric approach. It avoids case division, which is the main weakness of some previously proposed algorithms. The original motivation for the proposed approach was to estimate the position of a mobile who possibly has a different height also and hence is not in the plane spanned by the three BS. Compared to the usual trilateration, this scenario corresponds to one possible error case, of the three estimated distances being larger than their true values.

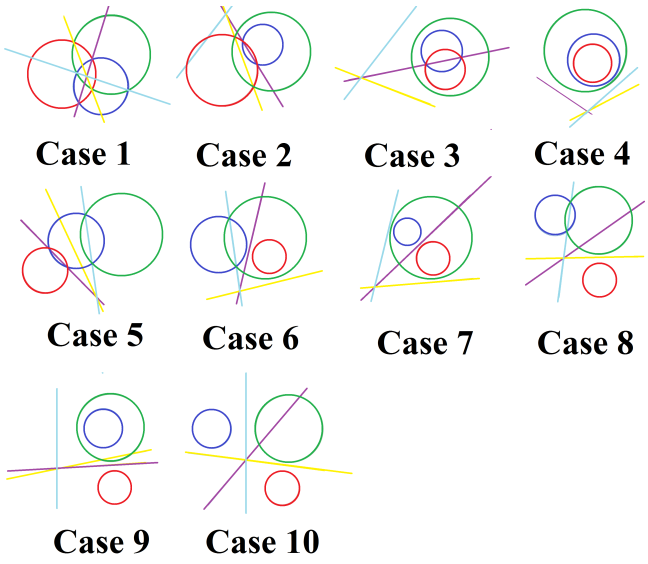



Figure 2.1: All possible cases resulting from orientation and relative radii of the 3 circles, as well as their **radical axes**. The yellow line is the radical axis of the red and green circles. The purple line is the radical axis of the red and blue circles. The cyan line is the radical axis of the green and blue circles. The 3 radical axes are always concurrent.

## 2.2 Trilateration

### 2.2.1 Previous works

Many algorithms on trilateration have been proposed so far. In 2D models, the position of a mobile station is considered to be the point whose distance to the each stations is equal to the distance between the mobile station and the corresponding base station. It means that they find the intersection point of the circles whose centers are the base stations (usually 3 base stations). The radius of each circle is the estimated distance from the mobile station to the base station concerned. In the ideal scenario where there is no noise as well as no inaccuracy in measurements, there must be a unique point in which the 3 circles intersect. Undoubtedly, it is the position of the mobile station.

However, such an ideal scenario never exists in reality. The estimated distance is never exactly the correct true distance. Therefore, the 3 circles never intersect in 1 point. [15] summarizes all the possible cases of relative position of 3 circles. 2 circles can intersect each other at 2 common points; **or** touch each other internally or externally (1 common point); **or** lie inside or outside each other (0 common point). Hence, **the total number of intersection points of each pair of 3 circles** is no more than 6. All ten possible cases are illustrated in Fig. 2.1.

Frequently, the 3 circles intersect each other at 6 points (case 1). The composition of 3 points that are closest to each other will be selected. [16] proposed a rule to select the correct 3 points needed. This rule selects the 3 points that stay closest to each other. Nonetheless, when the noise is quite considerable, this rule leads to select the wrong

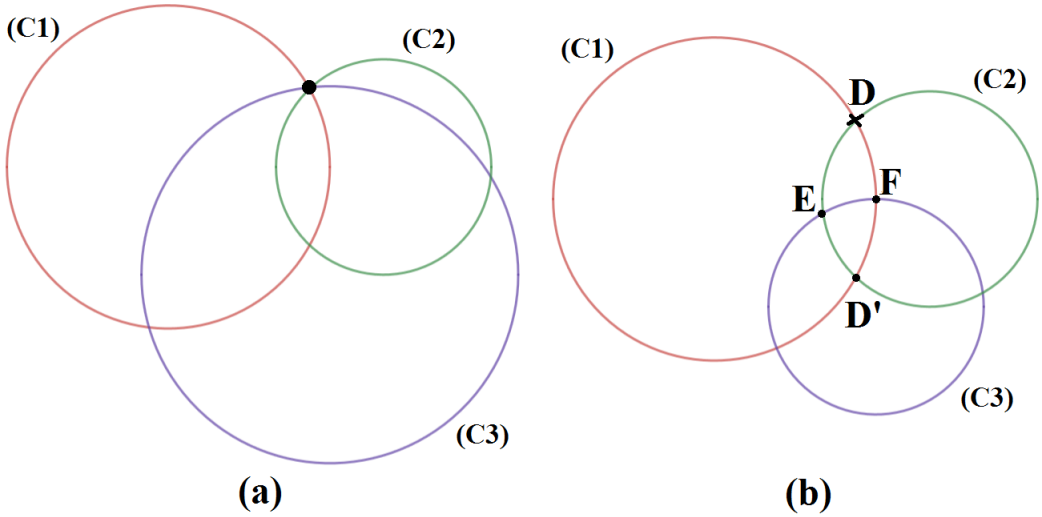



Figure 2.2: Intersections of 3 circles in an ideal scenario (a) and in a scenario with erroneous measurements (b).

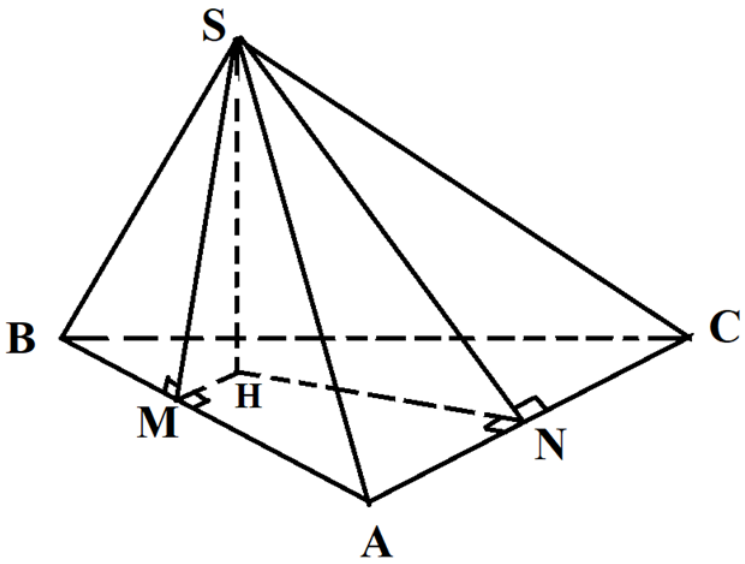



Figure 2.3: Tetrahedron S.ABC formed by 3 base stations A, B, C and the intersection point S of the 3 spheres

composition of 3 points. Fig. 2.2a shows the ideal scenario when the 3 circles intersect each other at one point. In Fig. 2.2b, a noticeable noise makes the radius of circle (C3) much smaller. Based on the rule in [16], the selected points are D', E, F because they are closest to each other; instead of the correct composition of points D, E, F. As a result, the estimated position can be sizeably erroneous.

The selected 3 points form a triangle. In [16], the **Fermat Point** (the point such that the total distance from the three vertices of the triangle to the point is the minimum possible) of the triangle is taken as the position of the mobile station. The authors of [17] suggested the position to be the **centroid** of the triangle. Nevertheless, those algorithms only work in the standard case where $n = 6$. As for the other cases ($n < 6$), more functions, more analyses need to be performed.

In the paper [18] on localization in a 3D model, the author proposed a method to position in 3D. Similar to 2D models where circles are drawn, in 3D spheres appear. Each sphere has the center at the base station concerned and the radius as the distance between this base station and the mobile station. The intersection points of the 3 spheres are found, they are considered to be the position of the mobile station. Fig. 3 demonstrates a tetrahedron SABC where A, B, C are the base stations and S is the intersection point of the 3 spheres. Although this method could not solve the problem of the cases when the intersection points don't exist, it suggests a novel idea for positioning in 2D. We consider the orthogonal projection point of the apex of the tetrahedron onto the base plane as the position of the mobile station (point H is Fig. 2.3). We are going to prove that there exists **one and only one** point like this in all the cases, even when there is no intersection point of 3 spheres.

### 2.2.2 The most usual case

In this chapter, $d_{AB}$ denotes the length of the line segment AB, and (A, $d_A$) is the circle of center A and radius $d_A$.

We consider the 3 base stations as the 3 vertices of a triangle. Generally, the 3 spheres intersects each other at *at least* 1 point. This point together with the 3 base stations form a tetrahedron in which it is the apex and the base plane is the plane through 3 base station (Fig. 2.3). In our work, we consider the orthogonal projection of the apex S on the base plane (ABC) as the estimated position of the mobile station.

We find that orthogonal projection point, as well as the foot of the tetrahedron's altitude through point S.

Defining $d_A$, $d_B$ and $d_C$ are the measured distance from the mobile station to 3 base stations A, B and C, respectively. We have SA = $d_A$; SB = $d_B$ and SC = $d_C$.

Let M be the foot of the altitude through point S of triangle SAB, N be the foot of the altitude through point S of triangle SAC.

In plane (ABC), we draw a line perpendicular to AB through M and a line perpendicular to AC through N. The 2 lines intersect at point H. We have:

SM $\perp$ AB and HM $\perp$ AB $\Rightarrow$ (SHM) $\perp$ AB $\Rightarrow$ SH $\perp$ AB

SN $\perp$ AC and HN $\perp$ AC $\Rightarrow$ (SHN) $\perp$ AC $\Rightarrow$ SH $\perp$ AC

SH $\perp$ AB and SH $\perp$ AC $\Rightarrow$ SH $\perp$ (ABC)

Therefore, H is the foot of the altitude through point S of the tetrahedron SABC. We consider H as the estimated position of the mobile station.

SM is the altitude of the triangle SAB so

$$d_{SM}^2 = d_{SA}^2 - d_{AM}^2 = d_{SB}^2 - d_{BM}^2 \tag{2.1}$$

Thus

$$d_{AM}^2 - d_{BM}^2 = d_{SA}^2 - d_{SB}^2 = d_A^2 - d_B^2 \tag{2.2}$$

Similarly, triangle SAC, HAB, HAC has altitude SN, HM, HN, respectively.

$$d_{AN}^2 - d_{CN}^2 = d_{SA}^2 - d_{SC}^2 = d_A^2 - d_C^2 \tag{2.3}$$

$$d_{AM}^2 - d_{BM}^2 = d_{HA}^2 - d_{HB}^2 \tag{2.4}$$

$$d_{AM}^2 - d_{CN}^2 = d_{HA}^2 - d_{HC}^2 \tag{2.5}$$

From (2.2)-(2.5), we deduce that

$$d_{HA}^2 - d_{HB}^2 = d_{SA}^2 - d_{SB}^2 = d_A^2 - d_B^2 \tag{2.6}$$

$$d_{HA}^2 - d_{HC}^2 = d_{SA}^2 - d_{SC}^2 = d_A^2 - d_C^2 \tag{2.7}$$

Therefore, H stays on the two lines: **radical axis** of circles (A, $d_A$) and (B, $d_B$) (*line 1*); **radical axis** of circles (A, $d_A$), (C, $d_C$) (*line 2*). As a result, H is the **radical center** of 3 circles (A, $d_A$), (B, $d_B$), (C, $d_C$).

**Short explanations:** Radical axis of 2 non-concentric circles is the locus of a point having equal power with regard to them. The geometric power of a point with respect to a circle is a real number that reflects the relative distance of the point from the circle. It is *positive* or *0* or *negative* when the point stays *outside* or *on* or *inside* the circle, respectively. The theory of **radical axis** of 2 circles and **radical center** of 3 circles are demonstrated in [19].

### 2.2.3 Suggested algorithm

The interesting thing is that we can always get the point H, even if we cannot form a tetrahedron SABC or even if we cannot form any or all of the 3 triangles SAB, SAC, SBC. Estimating the position of the mobile station becomes calculating the coordinate of the radical center of 3 circles.

1. Name the 3 base stations A, B and C. Estimate the distance $d_A$, $d_B$, $d_C$ from the mobile station to each base station, respectively.

2. Determine the 3 circles: (A, $d_A$), (B, $d_B$), (C, $d_C$). Then determine their 3 radical axes, taken in 3 pairs.

3. The 3 radical axes must be concurrent [19]. **The common point of the 3 lines is estimated position of the mobile station (Fig. 2.1).**

Let $x_A$, $x_B$, $x_C$ denote the abscissas of point A, B, C respectively and $y_A$, $y_B$, $y_C$ denote the ordinates of point A, B, C respectively.

As the 3 radical axes are concurrent, we only need to find the intersection point of any two of them.

The equation of the radical axis of circles (A, $d_A$) and (B, $d_B$)

$$(x_B - x_A)x + (y_B - y_A)y = d_B^2 - x_B^2 - y_B^2 - d_A^2 + x_A^2 + y_A^2 \tag{2.8}$$

The equation of the radical axis of circles (A, $d_A$) and (B, $d_B$)

$$(x_C - x_C)x + (y_C - y_A)y = d_C^2 - x_C^2 - y_C^2 - d_A^2 + x_A^2 + y_A^2 \tag{2.9}$$

Let $\boldsymbol{x} = \begin{bmatrix} x_H & y_H \end{bmatrix}^T$ be the coordinate vector of the point H. Since H is the intersection point of the 2 radical axes above:

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \tag{2.10}$$

where

$$\boldsymbol{A} = \begin{bmatrix} x_B - x_A & x_C - x_A \\ y_B - y_A & y_C - y_A \end{bmatrix} \tag{2.11}$$

$$\boldsymbol{b} = \begin{bmatrix} d_B^2 - x_B^2 - y_B^2 - d_A^2 + x_A^2 + y_A^2 \\ d_C^2 - x_C^2 - y_C^2 - d_A^2 + x_A^2 + y_A^2 \end{bmatrix} . \tag{2.12}$$

Hence

$$\boldsymbol{x} = \boldsymbol{A}^{-1}\,\boldsymbol{b} . \tag{2.13}$$

(2.13) is the equation to compute the coordinate of point H, which is taken as the estimated position of the mobile station.

When the number of base stations $N$ is larger than 3, we will obtain $N(N-1)/2$ radical axes. To estimate the radical center of those circles, Least Square (LS) is applied to get the solution of the overdetermined equation system. It turns out that this proposed method corresponds to method LS mentioned in [20], where an interesting state of the art appears, including optimally weighted least-squares versions.

### 2.2.4 Least Squares method

In matrix formulation, we define

$$\boldsymbol{A} = \begin{bmatrix} -2x_2 + 2x_1 & -2y_2 + 2y_1 \\ -2x_3 + 2x_1 & -2y_3 + 2y_1 \\ \cdots & \cdots \\ -2x_N + 2x_1 & -2y_N + 2y_1 \end{bmatrix} ; \quad \boldsymbol{x} = \begin{bmatrix} x \\ y \end{bmatrix}$$

$$\hat{\boldsymbol{b}} = \begin{bmatrix} \hat{d}_2^2 - x_2^2 - y_2^2 - \hat{d}_1^2 + x_1^2 + y_1^2 \\ \hat{d}_3^2 - x_3^2 - y_3^2 - \hat{d}_1^2 + x_1^2 + y_1^2 \\ \cdots \\ \hat{d}_N^2 - x_N^2 - y_N^2 - \hat{d}_1^2 + x_1^2 + y_1^2 \end{bmatrix}$$

We then have the equation of approximation

$$\boldsymbol{A}\,\boldsymbol{x} \approx \hat{\boldsymbol{b}} \tag{2.14}$$

Therefore, the estimate of $\boldsymbol{x}$ is

$$\hat{\boldsymbol{x}} = \min_{\boldsymbol{x}} \|\boldsymbol{A}\boldsymbol{x} - \hat{\boldsymbol{b}}\|^2 \tag{2.15}$$

$\hat{\boldsymbol{x}}$ is calculated by Least-Square estimation of $\boldsymbol{x}$

$$\hat{\boldsymbol{x}} = \hat{\boldsymbol{A}}^{\dagger}\hat{\boldsymbol{b}} \tag{2.16}$$

where $\boldsymbol{A}^{\dagger} = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T$ is the Moore-Penrose pseudo inverse of matrix $\boldsymbol{A}$.

### 2.2.5 Weighted Least Squares estimation

A Weighted Least Square estimation can also be applied. The estimate of $\boldsymbol{x}$ is compute by:

$$\hat{\boldsymbol{x}} = \left(\boldsymbol{A}^T\boldsymbol{W}^{-1}\boldsymbol{A}\right)^{-1}\boldsymbol{A}^T\boldsymbol{W}^{-1}\hat{\boldsymbol{b}} \tag{2.17}$$

where $\boldsymbol{W}$ is the covariance matrix of $\hat{\boldsymbol{b}}$:

$$\boldsymbol{W} = \mathrm{var}(d_1^2) \bullet \mathbf{1} \bullet \mathbf{1}^T + \mathrm{diag}\left(\mathrm{var}(d_2^2), \mathrm{var}(d_3^2), \ldots, \mathrm{var}(d_N^2)\right) \tag{2.18}$$

with $\mathrm{var}(d_i^2)$ is the variance of $d_i^2$ and $\mathbf{1} = [1 \;\; 1 \;\; \ldots \;\; 1]^T$ is the all-one vector.

## 2.3 RSS Model and Attenuation Exponent Estimation

### 2.3.1 RSS Model

RSS is the average power received over a wireless link. Field trials have validated that the disturbance in RSS due to shadowing is log-normal distributed. Accordingly, the log-normal path loss model can be expressed as:

$$P_i = P_0 + 10\alpha_i \log(d_0) - 10\alpha_i \log(d_i) + n_{\mathrm{RSS},i} \tag{2.19}$$

where $P_0$ is the power received (in dBm) at a reference point at distance $d_0$, $d_i$ is the actual distance from the $i$-th base station to the mobile station, $P_i$ is the power received (in dBm) at that base station, $\alpha_i$ is the path loss exponent of the corresponding transmission

link, $n_{\mathrm{RSS},i}$ is the log normal disturbance. The Path Loss Model (PLM) assumes that this disturbance has a Gaussian distribution with zero mean and a variance of $\sigma^2_{\mathrm{RSS},i}$.

The RSS measurement is simplified to

$$\zeta_{\mathrm{RSS},i} \triangleq P_i - P_0 - 10\alpha \log d_0 \tag{2.20}$$

From the Appendix B, we have the expected value and variance of $d_i^2$ as follows:

$$\widehat{d_i^2} = \mathrm{E}(d_i^2) = \exp\left(\frac{-(\ln 10)\zeta_{\mathrm{RSS},i}}{5\alpha_i} - \frac{(\ln 10)^2\sigma^2_{\mathrm{RSS},i}}{50\alpha_i^2}\right) \tag{2.21}$$

$$\mathrm{var}(d_i^2) = \exp\left(\frac{-(\ln 10)\zeta_{\mathrm{RSS},i}}{2.5\alpha_i} - \frac{(\ln 10)^2\sigma_i^2}{12.5\alpha_i^2}\right)\left[\exp\left(\frac{(\ln 10)^2}{25\alpha_i^2}\sigma_i^2\right) - 1\right] \tag{2.22}$$

### 2.3.2 ML Position Optimization with Steepest-Descent

To optimize $\hat{\boldsymbol{x}}$ obtained in the section 2.2.4, an iterative Maximum Likelihood estimator is applied.

In vector formulation, we denote

$$\boldsymbol{\zeta}_{\mathrm{RSS}} = \begin{bmatrix}\zeta_{\mathrm{RSS},1} & \zeta_{\mathrm{RSS},2} & \dots \zeta_{\mathrm{RSS},N}\end{bmatrix}^T \tag{2.23}$$

$$\boldsymbol{f}_{\mathrm{RSS}}(\boldsymbol{x}) = \begin{bmatrix} -10\alpha_1 \log\sqrt{(x-x_1)^2+(y-y_1)^2} \\ -10\alpha_2 \log\sqrt{(x-x_2)^2+(y-y_2)^2} \\ \dots \\ -10\alpha_N \log\sqrt{(x-x_N)^2+(y-y_N)^2} \end{bmatrix} \tag{2.24}$$

where $\boldsymbol{x} = \begin{bmatrix} x & y \end{bmatrix}^T$.

The vector $\boldsymbol{\zeta}_{\mathrm{RSS}}$ is Gaussian distributed with mean vector of $\boldsymbol{f}$ and covariance matrix $\boldsymbol{C}_{\mathrm{RSS}}$, we have the probability density function (pdf) [21]:

$$\mathrm{p}(\boldsymbol{\zeta}_{\mathrm{RSS}}|\boldsymbol{x}) = \frac{(2\pi)^{-\frac{N}{2}}}{|\boldsymbol{C}_{\mathrm{RSS}}|^{\frac{1}{2}}} \exp\left(\frac{-1}{2}(\boldsymbol{\zeta}_{\mathrm{RSS}} - \boldsymbol{f}_{\mathrm{RSS}})^T \boldsymbol{C}_{\mathrm{RSS}}^{-1}(\boldsymbol{\zeta}_{\mathrm{RSS}} - \boldsymbol{f}_{\mathrm{RSS}})\right) \tag{2.25}$$

where

$$\boldsymbol{C}_{\mathrm{RSS}} = \mathrm{diag}(\sigma^2_{\mathrm{RSS},1}, \sigma^2_{\mathrm{RSS},2}, \dots, \sigma^2_{\mathrm{RSS},N}) \tag{2.26}$$

Maximizing the pdf in (2.25) is equivalent to finding

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} J_{\mathrm{RSS}} \tag{2.27}$$

where $J_{\mathrm{RSS}}(\boldsymbol{x}) = (\boldsymbol{\zeta}_{\mathrm{RSS}} - \boldsymbol{f}_{\mathrm{RSS}}(\boldsymbol{x}))^T \boldsymbol{C}_{\mathrm{RSS}}^{-1}(\boldsymbol{\zeta}_{\mathrm{RSS}} - \boldsymbol{f}_{\mathrm{RSS}}(\boldsymbol{x}))$

We consider iterative Steepest-Descent procedure [22] for $\hat{\boldsymbol{x}}$. At iteration $(u+1)$:

$$\hat{\boldsymbol{x}}^{(u+1)} = \hat{\boldsymbol{x}}^{(u)} - \mu_0 \nabla(J_{RSS}(\hat{\boldsymbol{x}}^{(u)}) \tag{2.28}$$

with $\nabla(J_{\text{RSS}})(\hat{\boldsymbol{x}}^{(k)})$ is the Gradient vector of $J_{\text{RSS}}$ at the value $\hat{\boldsymbol{x}}^{(u)}$ which is detailed in Appendix A; and $\mu_0$ is the step-size.

Stopping criterion: $\|\boldsymbol{x}^{(u+1)} - \boldsymbol{x}^{(u)}\|_2 < \varepsilon_{\text{RSS}}$, where $\varepsilon_{\text{RSS}}$ is a sufficiently small positive constant.

The initiation value $\hat{\boldsymbol{x}}^{(1)}$ is obtained by the geometric approach or Weighted Least Squares method.

## 2.4 Estimation of Path Loss Exponent

To estimate the path loss exponent, beside the $N$ base stations given, we also use $M$ emitters at known positions. In absence of disturbance:

$$P_{i,j} = P_0 + 10\alpha \log(d_0) - 10\alpha_{i,j} \log(d_{i,j}) + n_{\text{RSS},i,j} \tag{2.29}$$

where $P_{i,j}$ is the power received at $i$-th base station, emitted by the $j$-th emitter, and $d_{i,j}$ is the actual distance between the $i$-th base station and the $j$-th emitter and $n_{\text{RSS},i,j}$ is the corresponding lognormal shadowing. We denote:

$$\boldsymbol{\zeta}_\alpha = \begin{bmatrix} P_{1,1} - P_0 \\ P_{1,2} - P_0 \\ \cdots \\ P_{1,N} - P_0 \\ \cdots \\ P_{M,1} - P_0 \\ P_{M,2} - P_0 \\ \cdots \\ P_{M,N} - P_0 \end{bmatrix}, \ \boldsymbol{\psi}_\alpha = \begin{bmatrix} 10 \log \frac{d_0}{d_{1,1}} \\ 10 \log \frac{d_0}{d_{1,2}} \\ \cdots \\ 10 \log \frac{d_0}{d_{1,N}} \\ \cdots \\ 10 \log \frac{d_0}{d_{M,1}} \\ 10 \log \frac{d_0}{d_{M,2}} \\ \cdots \\ 10 \log \frac{d_0}{d_{M,N}} \end{bmatrix}, \ \boldsymbol{n}_{\text{RSS}} = \begin{bmatrix} n_{\text{RSS},1,1} \\ n_{\text{RSS},1,2} \\ \cdots \\ n_{\text{RSS},1,N} \\ \cdots \\ n_{\text{RSS},M,1} \\ n_{\text{RSS},M,2} \\ \cdots \\ n_{\text{RSS},M,N} \end{bmatrix}. \tag{2.30}$$

We assume that all the transmission links have the same path loss exponent, or $\alpha_{1,1} = \cdots = \alpha_{M,N} = \alpha$. Therefore, the joint measurements can be written as

$$\boldsymbol{\zeta}_\alpha = \boldsymbol{\psi}_\alpha \, \alpha + \boldsymbol{n}_{\text{RSS}}. \tag{2.31}$$

The Least-Squares estimate is then

$$\hat{\alpha} = \boldsymbol{\zeta}_\alpha^\dagger \boldsymbol{\psi}_\alpha \tag{2.32}$$

## 2.5 Simulation Results

### 2.5.1 Environment setup

Our simulation scenario considers an area of size 1000m x 1000m. As for other parameters, $P_0$ = -45 dBm at $d_0$ = 10m.

The 3 base stations' coordinators are (400; 400), (600; 400) and (500; 600). 1000 mobiles stations are randomly picked in the area (Fig. 2.4).

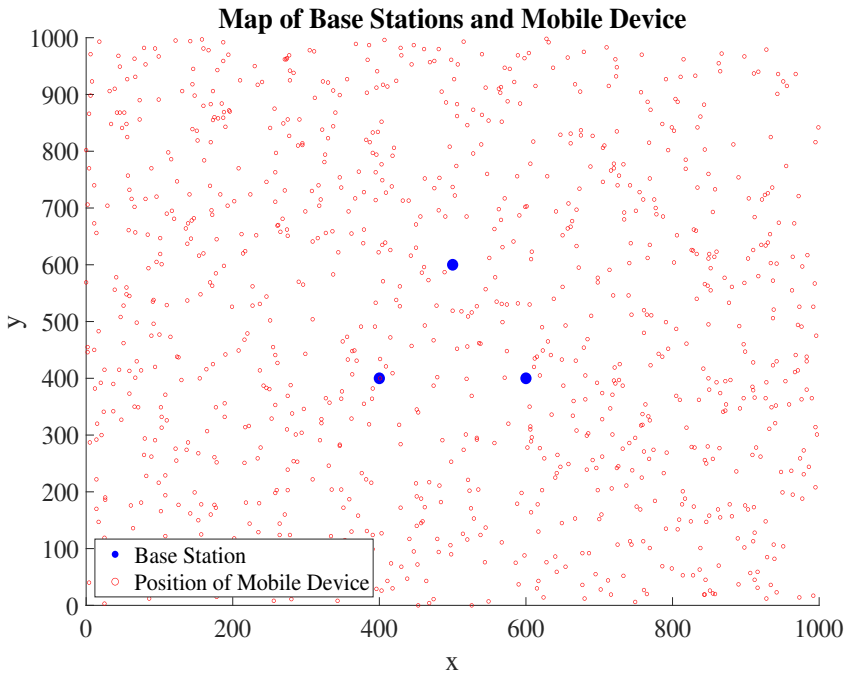



Figure 2.4: Map of Base stations and Positions of Mobile Device

We assume that all the transmission links have the same path loss exponents ($\alpha_1 = \alpha_2 = \cdots = \alpha_N$) and all the log-normal shadowings have the same standard deviation ($\sigma_{\text{RSS},1} = \sigma_{\text{RSS},2} = \cdots = \sigma_{\text{RSS},N} = \sigma_{\text{RSS}}$).

### 2.5.2 Results

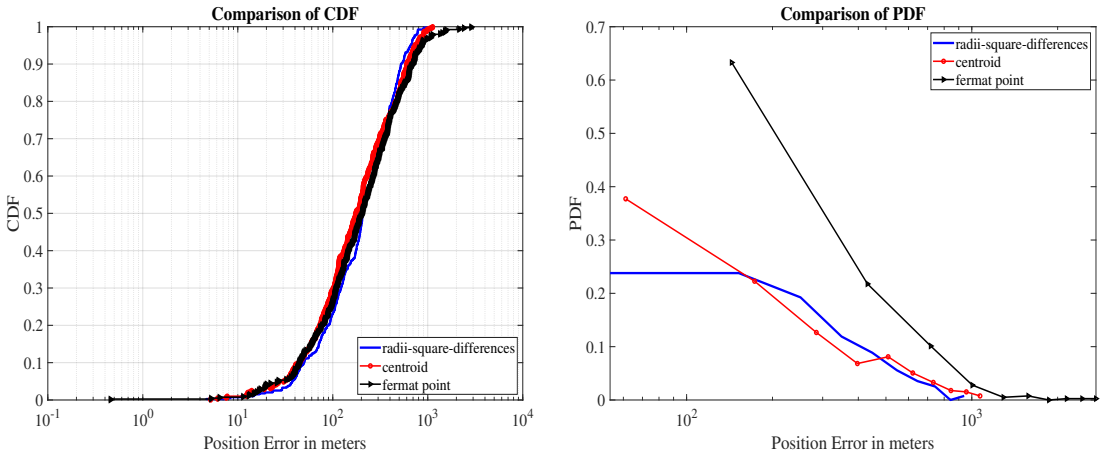
To compare the results among the algorithms, we calculate the Average Position Error (AVE), which is defined:

$$\text{AVE} \triangleq \frac{1}{Z} \sum_{l=1}^{Z} \|\boldsymbol{x}^{(l)} - \hat{\boldsymbol{x}}^{(l)}\|_2 \tag{2.33}$$

where $\boldsymbol{x}^{(l)}$ is the $l$-th actual position of the mobile station, $\hat{\boldsymbol{x}}^{(l)}$ is the $l$-th estimated position of the mobile station, Z is the number of positions randomly picked up (in this setup, Z = 1000).

#### Three geometric solutions

There are 3 geometric solutions for Trilateration menntionned above: Centroid [17], Fermat Point [16] and radii-square-differences (our proposed solution). We implement simulations of the three solutions. The expected value and variance of $d_i^2$ are estimated by the equations (2.21) and (2.22), respectively.




(a) CDF comparison (b) PDF comparison

Figure 2.5: Comparisons of CDF and PDF among 3 geometric solutions, when $\alpha = 2$ and $\sigma_{\text{RSS}} = 2$.

Fig. 2.5 compares the Cumulative Distribution Function (CDF) and Probability Distribution Function (PDF) of the AVE of the three solutions, when $\alpha = 2$ and $\sigma_{\text{RSS}} = 2$. It is obvious that our proposed solution (radii-difference-square) has the most accurate positioning estimations.

**Estimation of path loss exponent**

Table 2.1 illustrates the AVE obtained when we do not re-estimate the path loss component. The value of $\alpha$ varies from 2 to 6, the variance of additive disturbance is 1 and 2.

Table 2.2 illustrates the AVE obtained when we re-estimate the path loss component before estimating the position of the mobile station. 5 fixed emitters are placed at (0; 0), (0; 1000), (1000; 0), (1000; 1000) and (500; 500). The estimated value of path loss components are also presented in the table.

In a nutshell, the results shown in the two tables above are summarized in Fig. 2.6, in order to give a clear comparison.

## 2.6 Conclusions

In the range of Trilateration positioning, this section proposes a new geometric method that can be applied in all measurement error cases. The numerical results shows, compared to the centroid algorithm and the Fermat Point algorithm, that our proposed approach helps to significantly improve the accuracy in localization and reduce the complexity by avoiding case division. Furthermore, experimental results also demonstrate that integrating path loss estimation can make the position estimation more accurate.

Nevertheless, only results in 2D simulation are shown. As for 3D models, algorithms are more complicated and are currently being investigated.

| | $\sigma = 1$ | $\sigma = 2$ |
|---|---|---|
| $\alpha = 2$ | 79.459 | 161.481 |
| $\alpha = 2.5$ | 57.289 | 130.180 |
| $\alpha = 3$ | 47.187 | 102.248 |
| $\alpha = 3.5$ | 41.074 | 83.031 |
| $\alpha = 4$ | 34.003 | 75.210 |
| $\alpha = 4.5$ | 30.890 | 63.755 |
| $\alpha = 5$ | 27.036 | 58.434 |
| $\alpha = 5.5$ | 25.604 | 51.040 |
| $\alpha = 6$ | 23.480 | 46.820 |

Table 2.1: Average Position Error in localization without estimation of path loss component

| | $\sigma = 1$ | | $\sigma = 2$ | |
|---|---|---|---|---|
| $\alpha = 2$ | $\hat{\alpha} = 2.0055$ | 76.953 | $\hat{\alpha} = 2.0075$ | 156.388 |
| $\alpha = 2.5$ | $\hat{\alpha} = 2.504$ | 56.524 | $\hat{\alpha} = 2.511$ | 122.647 |
| $\alpha = 3$ | $\hat{\alpha} = 3.0026$ | 47.489 | $\hat{\alpha} = 3.0099$ | 105.778 |
| $\alpha = 3.5$ | $\hat{\alpha} = 3.5024$ | 41.337 | $\hat{\alpha} = 3.5148$ | 88.051 |
| $\alpha = 4$ | $\hat{\alpha} = 4.0015$ | 33.963 | $\hat{\alpha} = 4.0166$ | 74.850 |
| $\alpha = 4.5$ | $\hat{\alpha} = 4.5036$ | 28.722 | $\hat{\alpha} = 4.5043$ | 63.919 |
| $\alpha = 5$ | $\hat{\alpha} = 5.0008$ | 27.663 | $\hat{\alpha} = 5.0192$ | 58.098 |
| $\alpha = 5.5$ | $\hat{\alpha} = 5.5045$ | 25.751 | $\hat{\alpha} = 5.5134$ | 50.735 |
| $\alpha = 6$ | $\hat{\alpha} = 6.0018$ | 22.740 | $\hat{\alpha} = 6.0091$ | 47.981 |

Table 2.2: Average Position Error in localization with estimation of path loss component

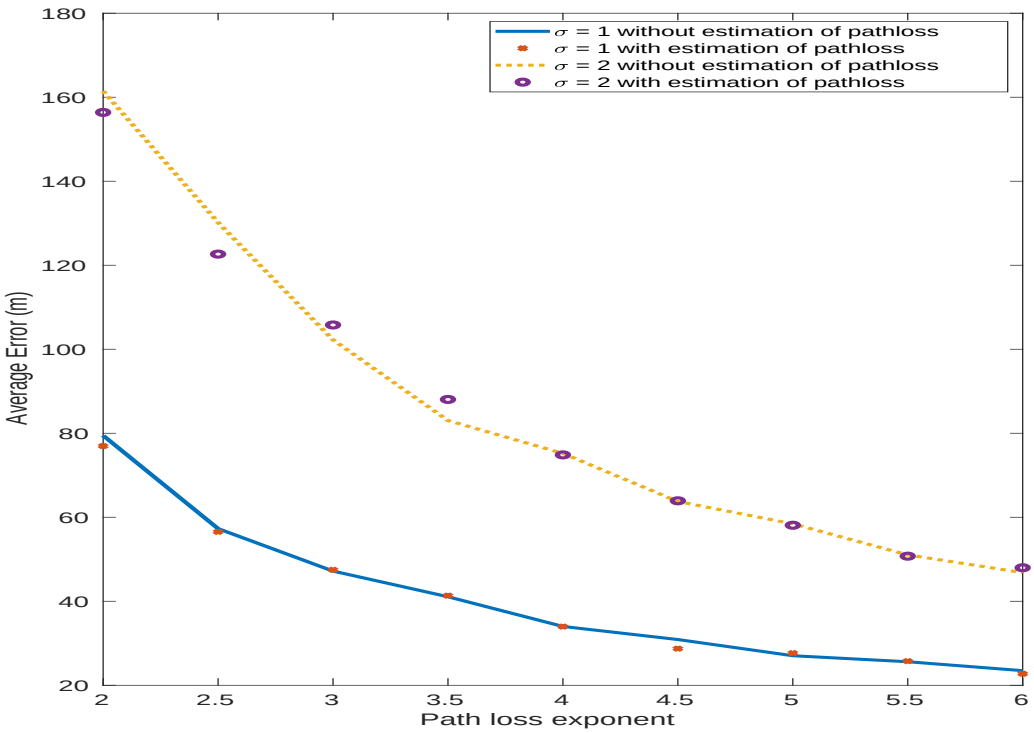



Figure 2.6: AVE comparison of localization with and without prior estimation of path loss

# Chapter 3

# Multilateration: TDoA-based localization

## 3.1 Introduction

Immense amount of data can be shared on the recently developed open source platforms since 5G wireless networks are developing. Location information of mobile devices are greatly advantageous among the new services which can be developed with these data, for instance, supporting wireless operators to enhance the network performance [23].

TDoA systems avoid the requirements of clock synchronization at the point of interest or tag-end by considering the different arrival times of signals that originate at two distinct reference points [24]. The time difference calculation effectively cancels out time synchronization errors at the tag. We notice that TDoA-based positioning schemes still require clock synchronization among all the beacons in the system. Measurements of TDoA also involve the correlation of transmitted and received signals to determine the arrival time of the strongest correlation peak. Thus, excluding synchronization issues, the sources of error of TDoA-based systems are the same as the ones described for ToA schemes.

TDoA-based positioning algorithm is a popular topic of many previous works. The authors of [25] propose a geometric approach for positioning problems, but optimization is not considered. In the paper, [26–29], the theories for TDoA-based positioning algorithms are proposed to make the position estimations much more robust. However, the practical implementations are not well-considered, neither are the practice-like simulations. They also assume that all the TDoA estimations have the same characteristics, which is difficult to witness in practice.

In this chapter, we measure TDoA estimation in practice scenarios. Section 3.2 illustrates the TDoA estimation in a selected practical scenario. All the related positioning algorithms are analyzed in section 3.3. In the last two sections, we provide the simulations, their results, and our conclusions.

## 3.2 TDoA estimation

### 3.2.1 General description of NR positioning

The 3GPP NR technology, particularly in Release-16, contributes to location-based services such as precision UE positioning. This operation occurs in the low and high-frequency bands FR1 and FR2 (below and above 6GHz), allowing for the utilization of wide signal bandwidths of 100MHz and 400MHz for FR1 and FR2, respectively. This will result in new performance constraints for approaches such as DL TDoA, which use timing measurements to find the user. However, it has some propagation issues in mmWave communications. In comparison to LTE, more bandwidth and large antenna arrays (massive MIMO) in NR positioning systems offer more degrees of freedom for improved location accuracy. The following are NR design goals for commercial positioning use cases:

- Support for a variety of accuracy, latency, and device types.
- For some use cases, support accuracy and latency as described in TR 22.862.
- Reduced network complexity.

### 3.2.2 General description of NR positioning

For evaluating baseline performance, scenarios (with various options/configurations) are defined below for RAT- dependent positioning techniques for NR positioning study:

- Scenario 1: Indoor Office for FR1 and FR2 (Open office).
- Scenario 2: UMi street canyon for FR1 and FR2 (ISD 200m).
- Scenario 3: UMa (ISD 500m) for FR1 only (Macro cell only deployment scenario).

For regulatory use cases, the following requirements are considered as a minimum performance targets for NR positioning:

- Horizontal positioning error $\leq$ 50m for 80% of UEs.
- Vertical positioning error $<$ 5m for 80% of UEs.

As a starting point for commercial use cases, the following requirements are considered:

- Horizontal positioning error $<$ 3m for 80% of UEs in indoor deployment scenarios.
- Vertical positioning error $<$ 3m for 80% of UEs in indoor deployment scenarios.
- Horizontal positioning error $<$ 10m for 80% of UEs in outdoor deployments scenarios.
- Vertical positioning error $<$ 3m for 80% of UEs in outdoor deployment scenarios

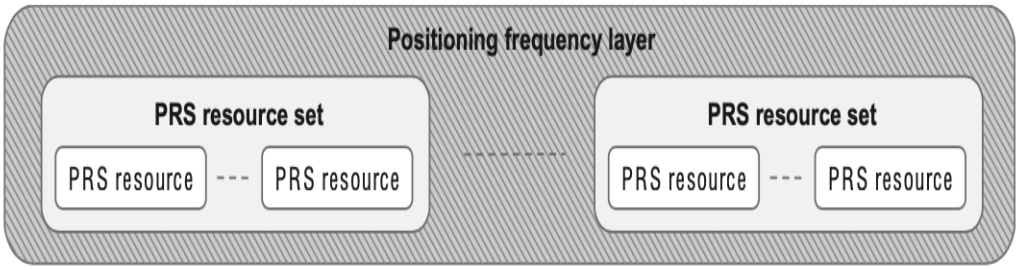



Figure 3.1: PRS resource and resouce set

**PRS Resource Set and Resource**

One positioning frequency layer is made up of one or more PRS resource sets distributed over one or more sites, all of which use the same carrier frequency and OFDM numerology. Each PRS resource in a resource set corresponds to a beam from a single location. When configuring a device to measure on a specific PRS resource in a PRS resource set, the location server learns not only which site the reported measurements for this PRS resource set belong to, but also the specific beam from that site.

**PRS Comb Size**

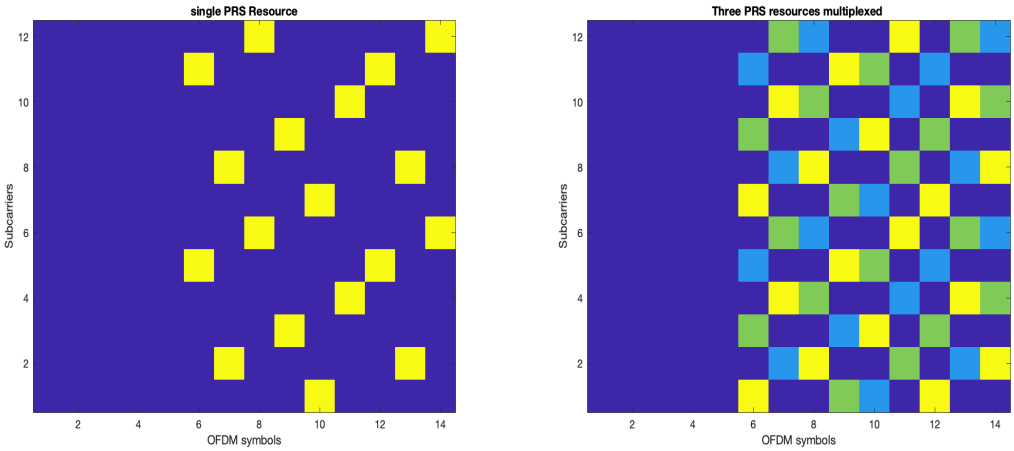
PRS setup enables permuted staggered comb. In this application, permuted indicates that the comb in each OFDM symbol has a distinct, and not necessarily monotonically increasing, offset in the frequency domain. The comb factor can be adjusted to 2, 4, 6, or 12 subcarriers, which means that the comb is used on every 2nd to every 12th subcarrier. Using various combs, numerous simultaneous PRSs may be multiplexed using a comb.

**PRS Bandwidth**

A PRS resource in the frequency domain can be configured to have a bandwidth of up to 272 resource blocks (100MHz), with all PRS resources in a PRS resource set having the same bandwidth and frequency domain location. In the time domain, one PRS resource is represented by two, four, six, or twelve OFDM signals. A PRS resource occurs periodically in each cell in the time domain, with the periodicity adjustable from a few milliseconds to several seconds. The beginning points and periodicities of various PRS resources within the same PRS resource set may differ.

**PRS Muting Patterns**

Because the device must listen to positional reference signals from relatively distant locations, the near-far problem must be considered. As a result, a method is necessary to guarantee that the close base station remains silent while measuring on the far base station. This is achieved by utilizing a bitmap to describe various muting patterns. When a bit in the bitmap is set to zero, the positioning reference signals at that time instant




(a) single PRS resource (b) multiple PRS

Figure 3.2: PRS resources arranged in comb-6

are not sent. When this is combined with not sending data from that location at the same time, the overall result is a silent gap from that location, allowing the device to measure on a positing reference signal from a more distant base station.

### 3.2.3 Propagation model

To calculate propagation paths, ray tracing models employ 3-D environment geometry [30]. They estimate the path loss and phase shift of each ray using electromagnetic analysis, which includes tracking the horizontal and vertical polarization of a signal over the propagation path. The path loss includes both free-space and reflection losses. The model calculates losses on horizontal and vertical polarization for each reflection using the Fresnel equation, the incidence angle, and the relative permittivity and conductivity of the surface material at the specified frequency. Ray tracing models compute multiple propagation pathways, whereas the other available models compute single propagation paths. These models support both 3-D outdoor and indoor environments.

#### Image Method

This technique determines precise propagation paths and supports up to two path reflections. The amount of reflections raises the computational complexity significantly. The Image technique calculates the propagation path of a single reflection ray from a transmitter, Tx, to a receiver, Rx, as shown in the figure 3.3. Tx's image is located with regard to a flat reflection surface using the Image technique. The technique then uses a line segment to link Tx' and Rx. A valid path from Tx to Rx occurs if the line segment touches the planar reflection surface, represented as Q in the image. By recursively expanding these stages, the technique discovers paths with numerous reflections.

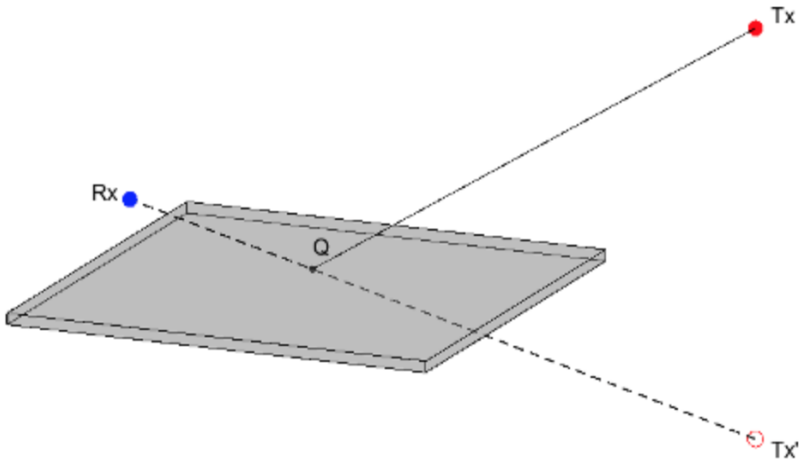



Figure 3.3: Ray tracing Image Method

**SBR Method**

In Shooting and Bouncing Ray (SBR), Up to ten path reflections are supported, and estimated propagation paths are calculated. As a result, the SBR method's estimated receiver site positions are not precise. As the length of the routes grows longer, the accuracy of the computed propagation paths diminishes. The amount of reflections raises the computational complexity linearly. As a result, the SBR technique is quicker than the image method in most cases.

This figure indicates how the SBR technique estimates the identical ray's propagation route. From a geodesic sphere centered at Tx, the SBR technique fires numerous rays. The technique then follows each ray from Tx as it reflects, diffracts, refracts, or scatters off nearby objects. It's worth noting that the implementation solely takes into account reflections. The technique surrounds Rx with a sphere, called a receiving sphere, having a radius proportional to the angular spacing of the launched rays and the distance the ray travels. The model considers the ray a legitimate path from Tx to Rx if it crosses the sphere. Both approaches may be used in conjunction with rain, fog, and gas in the environment.

### 3.2.4 NR TDL channel

To obtain the channel-impaired signal, the TDL-Channel System transmits an input signal through a tapped delay line (TDL) multi-input multi-output (MIMO) link-level fading channel. TDL-delay profile can be contumely configured using the propagation model outputs, Path Delay and Path Gain. More properties are available to be configured in this channel model such as Fading Distribution which can be Rician to consider the small scale fading as well as Sample Rate which is extracted from the input signal.

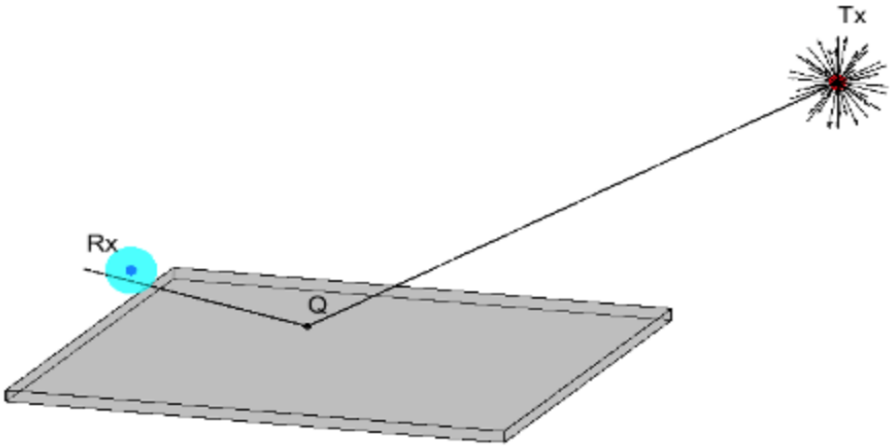



Figure 3.4: Ray Tracing SBR Method

### 3.2.5 Synchronization at signal reception

By the time that positioning signals are generated and transmitted through the channel, it is time for signal reception by UEs [31]. The first step before extracting PRS signals is for UEs to sync themselves with one gNB. since gNBs have tight timing synchronization, in the DL TDOA method we try to sync our UE to the same clock as gNBs. We obtain PSS (Primary Synchronization Signal) from SSB (Synchronization Signals Block), which is at the start of each radio slot, and cross-correlate it with the reference signals generated from NID2 values. For each transmitting cell, we repeat the same process to locate NID1 to get SSS(Secondary Synchronization Signal). We may compute their cell ID and pick the best of them in terms of received signal strength as the reference gNB to sync with once we have both NID1 and NID2 from all transmitters. There are 1008 unique physical-layer cell identities given by

$$N_{\mathrm{ID}}^{\mathrm{cell}} = 3N_{\mathrm{ID}}^{(1)} + N_{\mathrm{ID}}^{(2)} \tag{3.1}$$

where $N_{\mathrm{ID}}^{(1)} \in \{0, 1, \ldots, 335\}$ and $N_{\mathrm{ID}}^{(2)} \in \{0, 1, 2\}$.

We can extract various PRS symbols using their reference grid after demodulating the summation of received signals on the receiver side. We may begin to comprehend how our channel truly influenced the sent waveforms by using the least square approximation between the received and reference PRS.

### 3.2.6 Channel estimation

We average our recovered pilot symbols across time, then frequency bandwidth, with a growing window size that is proportional to the channel coherence time. Finally, we interpolate the findings over the whole subcarriers in the bandwidth, leaving us with channel estimation and some peaks that demonstrate when we received positional signals at different symbols over time.

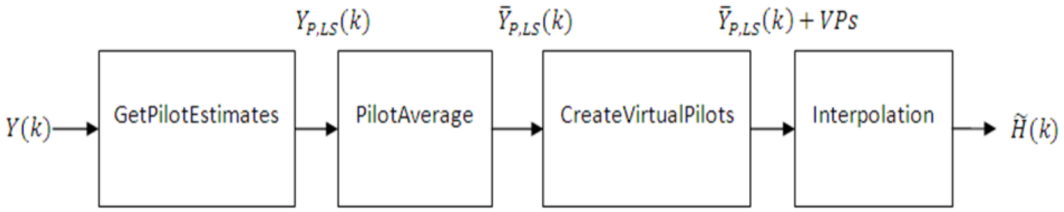
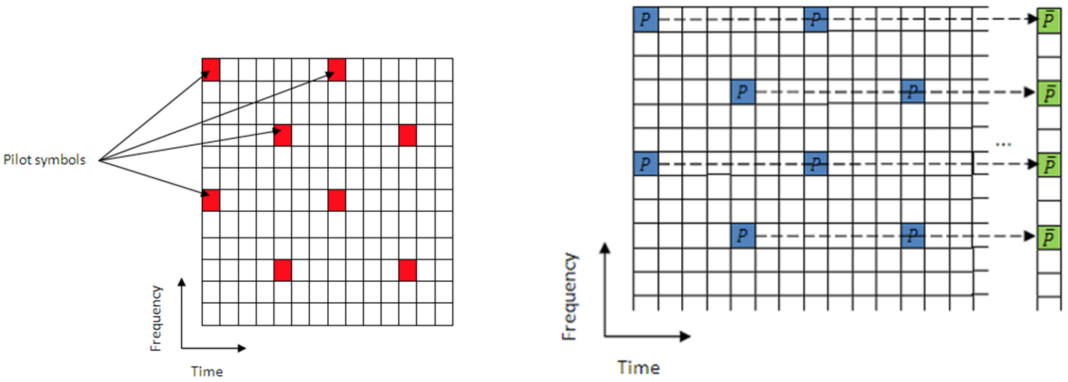
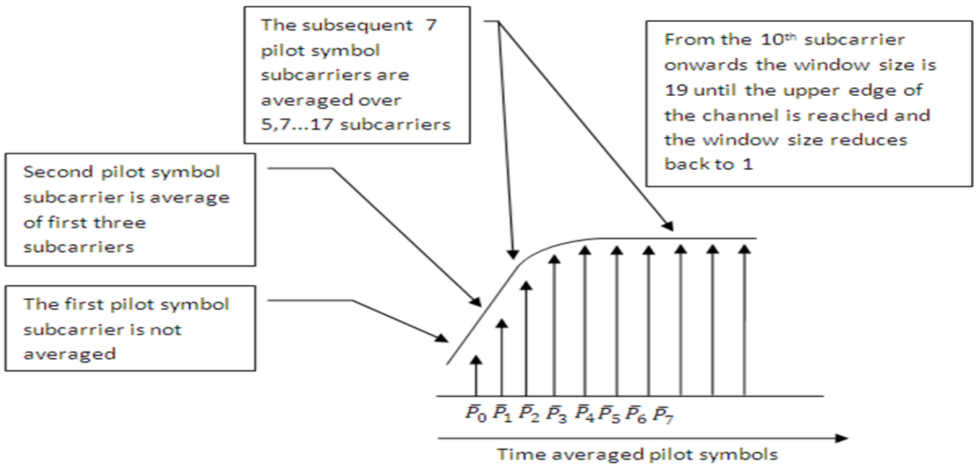



Figure 3.5: Channel Estimation Procedure




(a) PRS Pilots

(b) Time Averaging




(c) Frequency Averaging

Figure 3.6: Channel Estimation over PRS

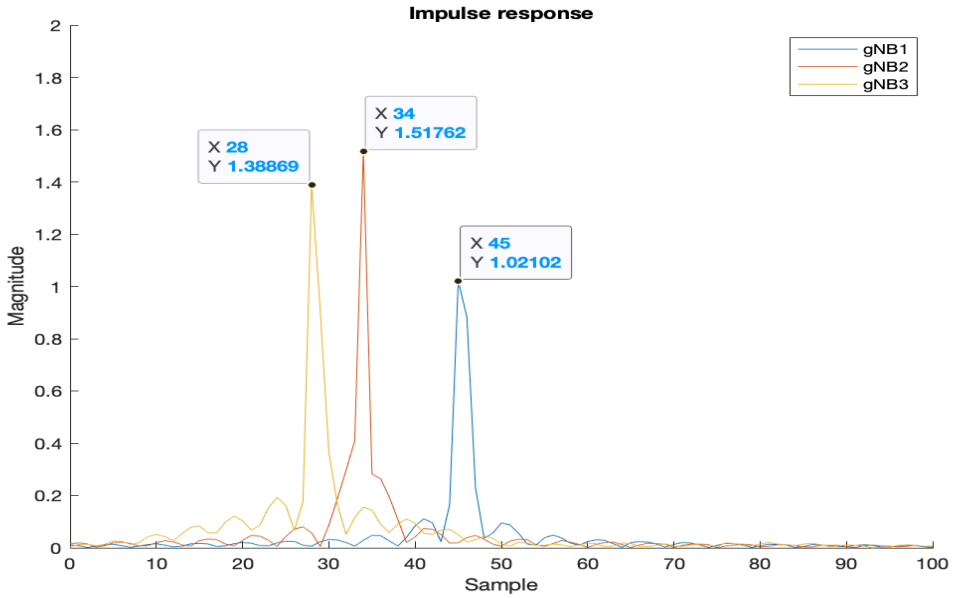



Figure 3.7: PRS Channel Impulse Response

## 3.3 Downlink TDoA-based localization

Let $t_i$ be the true propagation time of the signal from the $i$-th base station to the mobile deivce and $\hat{t}_i$ be its estimate, with $i$ from 1 to $N$ ($N$ is the number of base stations). We have the following equation

$$\hat{t}_i = t_i + \tilde{t}_i \tag{3.2}$$

where $\tilde{t}_i$ is the ToA estimation error and assumed to be Gaussian distributed with zero-mean and variance of $\sigma^2_{\text{ToA},i}$.

As a result, the distance is estimated based on multiplying $\hat{t}_i$ by $c$ ($c$ is the speed of light), denoted by $\hat{d}_i$ is modeled as:

$$\hat{d}_i = d_i + \tilde{d}_i \tag{3.3}$$

where $d_i = ct_i$ is the true distance and $\tilde{d}_i = c\tilde{t}_i$ is the distance estimation error based on the $i$-th ToA with zero-mean and variance of $\sigma_i^2 = c^2\sigma^2_{\text{ToA},i}$.

Thus, when $i \geq 2$, the distance differences are modeled:

$$r_i = d_i - d_1 \tag{3.4}$$

In presence of noises:

$$\hat{r}_i = d_i - d_1 + \tilde{r}_i \tag{3.5}$$

where $\tilde{r}_i = \tilde{d}_i - \tilde{d}_1$ is the error measurement, which has zero-mean and variance of $c^2(\sigma^2_{\text{ToA},i} + \sigma^2_{\text{ToA},1})$.

### 3.3.1 Equations

$(x, y, z)$ is the coordinate of the undefined mobile device and $(x_i,\ y_i,\ z_i)$ is the coordinate of the $i$-th base station, i from 1 to $N$.

When $i \geq 2$, we have

$$r_i = \sqrt{(x-x_i)^2 + (y-y_i)^2 + (z-z_i)^2} - \sqrt{(x-x_1)^2 + (y-y_1)^2 + (z-z_1)^2} \tag{3.6}$$

$$\sqrt{(x-x_i)^2 + (y-y_i)^2 + (z-z_i)^2} = r_i + \sqrt{(x-x_1)^2 + (y-y_1)^2 + (z-z_1)^2} \tag{3.7}$$

$$(x-x_i)^2 + (y-y_i)^2 + (z-z_i)^2 = r_i^2 + 2r_i\sqrt{(x-x_1)^2 + (y-y_1)^2 + (z-z_1)^2} \tag{3.8}$$

$$\begin{aligned} &2(x_1-x_i)(x-x_1) + 2(y_1-y_i)(y-y_1) + 2(z_1-z_i)(z-z_1) - 2r_iR \\ &= r_i^2 - (x_1-x_i)^2 - (y_1-y_i)^2 - (z_1-z_i)^2 \end{aligned} \tag{3.9}$$

with

$$R = \sqrt{(x-x_1)^2 + (y-y_1)^2 + (z-z_1)^2} \tag{3.10}$$

### 3.3.2 Geometric approach

(3.10) is the equation of an hyperboloid surface. In ideal scenarios where there is no error in TDoA estimation, the position of the mobile device is the intersection point of all the hyperboloids concerned.

Nevertheless, in practice, noises always exist in measurements, which cause certain errors in TDoA estimations. It is impossible to obtain such a common intersection point.

Algorithms have been proposed to overcome this issue. Naturally, a geometric approach was firstly considered [25]. In geometry, 3 hyperboloids have a limited number of the intersection points. Consequently, with a composition of 3 hyperboloids, one or two possible mobile positions can be obtained. The estimated position of the mobile, in geometric approach, is supposed to be the center of gravity (centroid) of all the intersection points mentioned above.

### 3.3.3 Least Squares method

In matrix formulation

$$\hat{\boldsymbol{A}}_{\text{TDoA}} = \begin{bmatrix} 2(x_1-x_2) & 2(y_1-y_2) & 2(z_1-z_2) & -2\hat{r}_2 \\ 2(x_1-x_3) & 2(y_1-y_3) & 2(z_1-z_3) & -2\hat{r}_3 \\ \cdots & \cdots & \cdots & \cdots \\ 2(x_1-x_N) & 2(y_1-y_N) & 2(z_1-z_N) & -2\hat{r}_N \end{bmatrix};\ \boldsymbol{\omega}_{\text{TDoA}} = \begin{bmatrix} x-x_1 \\ y-y_1 \\ z-z_1 \\ R \end{bmatrix}$$

$$\hat{\boldsymbol{b}} = \begin{bmatrix} \hat{r}_2^2 - (x_1 - x_2)^2 - (y_1 - y_2)^2 - (z_1 - z_2)^2 \\ \hat{r}_3^2 - (x_1 - x_3)^2 - (y_1 - y_3)^2 - (z_1 - z_3)^2 \\ \dots \\ \hat{r}_N^2 - (x_1 - x_N)^2 - (y_1 - y_N)^2 - (z_1 - z_N)^2 \end{bmatrix}$$

We have the equation of approximation

$$\hat{\boldsymbol{A}}_{\text{TDoA}}\, \boldsymbol{\omega}_{\text{TDoA}} \approx \hat{\boldsymbol{b}}_{\text{TDoA}} \tag{3.11}$$

We have

$$\hat{\boldsymbol{\omega}}_{\text{TDoA}} = \min_{\boldsymbol{\omega}_{\text{TDoA}}} \|\hat{\boldsymbol{A}}_{\text{TDoA}}\boldsymbol{\omega}_{\text{TDoA}} - \hat{\boldsymbol{b}}_{\text{TDoA}}\|^2 \tag{3.12}$$

leading to the estimate of $\boldsymbol{\omega}_{\text{TDoA}}$ being calculated by Least-Square estimation:

$$\hat{\boldsymbol{\omega}}_{\text{TDoA}} = \hat{\boldsymbol{A}}^{\dagger}_{\text{TDoA}}\hat{\boldsymbol{b}}_{\text{TDoA}} \tag{3.13}$$

where $\boldsymbol{A}^{\dagger} = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T$

Estimated coordinate vector of the mobile device are the 3 first elements of $\hat{\boldsymbol{\omega}}_{\text{TDoA}}$

$$\hat{\boldsymbol{x}} = \begin{bmatrix} [\hat{\boldsymbol{\omega}}_{\text{TDoA}}]_1 + x_1 & [\hat{\boldsymbol{\omega}}_{\text{TDoA}}]_2 + y_1 & [\hat{\boldsymbol{\omega}}_{\text{TDoA}}]_3 + z_1 \end{bmatrix}^T \tag{3.14}$$

### 3.3.4 Iterative Procedure for Optimization

To optimize $\hat{\boldsymbol{x}}$ obtained in (3.14), iterative procedures are applied.
In vector form, we denote

$$\hat{\boldsymbol{r}} = \begin{bmatrix} \hat{r}_2 & \hat{r}_3 & \dots \hat{r}_N \end{bmatrix}^T \tag{3.15}$$

$$\boldsymbol{f}_{\text{TDoA}}(\boldsymbol{x}) = \begin{bmatrix} \sqrt{(x - x_2)^2 + (y - y_2)^2 + (z - z_2)^2} - \sqrt{(x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2} \\ \sqrt{(x - x_3)^2 + (y - y_3)^2 + (z - z_3)^2} - \sqrt{(x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2} \\ \dots \\ \sqrt{(x - x_N)^2 + (y - y_N)^2 + (z - z_N)^2} - \sqrt{(x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2} \end{bmatrix} \tag{3.16}$$

where $\boldsymbol{x} = \begin{bmatrix} x & y & z \end{bmatrix}^T$.

We have the Cost Function

$$J_{\text{TDoA}} = (\hat{\boldsymbol{r}} - \boldsymbol{f}_{\text{TDoA}})^T(\hat{\boldsymbol{r}} - \boldsymbol{f}_{\text{TDoA}}) \tag{3.17}$$

The optimized estimate of $\boldsymbol{x}$ is the value that minimizes this Cost Function:

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}}(\hat{\boldsymbol{r}} - \boldsymbol{f}_{\text{TDoA}}(\boldsymbol{x}))^T(\hat{\boldsymbol{r}} - \boldsymbol{f}_{\text{TDoA}}(\boldsymbol{x})) \tag{3.18}$$

We consider Gauss-Newton procedure [22] for $\hat{\boldsymbol{x}}$. At iteration $(u+1)$:

$$\hat{\boldsymbol{x}}^{(u+1)} = \hat{\boldsymbol{x}}^{(u)} + (\boldsymbol{G}^T_{\text{TDoA}}\boldsymbol{G}_{\text{TDoA}})^{-1}\boldsymbol{G}^T_{\text{TDoA}}(\hat{\boldsymbol{r}} - \boldsymbol{f}(\hat{\boldsymbol{x}}^{(u)}) \tag{3.19}$$

where $\boldsymbol{G}_{\text{TDoA}}$ is the Jacobian matrix of $\boldsymbol{f}(\boldsymbol{x})$, which is introduced in Appendix A.

$$\boldsymbol{G}_{\text{TDoA}} = \boldsymbol{G}(\hat{\boldsymbol{x}}^{(u)}) \,,\ \boldsymbol{G}(\boldsymbol{x}) = \frac{\partial \boldsymbol{f}(\boldsymbol{x})}{\partial \boldsymbol{x}^T} \,. \tag{3.20}$$

A procedure is expected to terminate when $\|\hat{\boldsymbol{x}}^{(u+1)} - \hat{\boldsymbol{x}}^{(u)}\|_2 < \varepsilon_{\text{TDoA}}$, for the stopping criterion $\varepsilon_{\text{TDoA}}$ sufficiently small. Then, the final position of the procedure is considered to be the coordinates of the mobile device in the $xyz$ space.

However, the iterative procedures do not always converge. There are **three** possible outcomes for an iterative procedure:

- **Convergence:** The procedure quickly meets the stopping criterion and reaches the finite values.
- **Divergence:** The procedure reaches infinite values, and then it is forced to stop.
- **Oscillation:** The procedure oscillates with 2 or more repeated finite values. It does not diverge, but it is not able to converge. Experiments show that convergence or divergence appears in tens of iterations. Therefore, we set up the maximum number of iterations for each procedure is 1000. If at 1001st value, the stopping criterion is not met, the iterative procedure will be considered as an oscillating procedure and then forced to stop.

We take the final position of a converging procedure as the estimated position for the mobile device. As for a diverging procedure or an oscillating procedure, the initial position is selected.

In a nutshell, the Algorithm 1 is proposed for the Gauss-Newton iterative procedure of Maximum Likelihood estimator.

**Algorithm 1:** Proposed Maximum Likelihood estimtor with Gauss-Newton procedure

Take the estimated TDoA-based distance differences $\hat{r}_2, \hat{r}_3, ..., \hat{r}_N$.
Assign $u = 1$ and $\varepsilon_{\text{TDoA}}$ sufficiently small.
Compute the estimation $\hat{\boldsymbol{x}}$ by (3.14) as the first estimated coordinates of the mobile device.
**repeat**
    Compute the following estimated coordinates $\hat{\boldsymbol{x}}^{(u+1)}$ of the mobile device by (3.19).
    $u = u + 1$;
**until** $\|\hat{\boldsymbol{x}}^{(u+1)} - \hat{\boldsymbol{x}}^{(u)}\|_2 < \varepsilon_{TDoA}$ *or* $u > 1000$ *or* $\hat{\boldsymbol{x}}^{(u+1)} = \pm\infty$;
**if** $u > 1000$ *or* $\hat{\boldsymbol{x}}^{(u+1)} = \pm\infty$ **then**
    $\hat{\boldsymbol{x}}^{(1)}$ is the estimated position of the mobile device;
**else**
    $\hat{\boldsymbol{x}}^{(u)}$ is the estimated position of the mobile device;

| Base station | Latitude | Longitude | Elevation |
|---|---|---|---|
| 1 | 43.62177433110298 N | 7.064024322237882 E | 26 |
| 2 | 43.62328962324262 N | 7.070071763377428 E | 27 |
| 3 | 43.62254112669301 N | 7.075979940828803 E | 28 |
| 4 | 43.61976908041876 N | 7.075641936109399 E | 29 |
| 5 | 43.61771330834758 N | 7.075141249700595 E | 30 |
| 6 | 43.61675026847668 N | 7.071122828360069 E | 31 |
| 7 | 43.61813047305591 N | 7.066031724625405 E | 32 |

Table 3.1: Spherical coordinates of the base stations

## 3.4 Simulations and Results

### 3.4.1 MatLab Simulation Setup

#### MatLab toolboxes

A couple of Matlab toolboxes are used to create the simulation environment where several gNBs and UEs can be placed. 5G toolbox is a key enabler for configured PRS properties such as PRS resource set period, Number of PRS Symbols in time and where they start in time, and how they multiplex with each other from different transmitters using their specific comb size and repetition period as well as Muting patterns. Another Useful tool is Communication Toolbox which allows defining transmitter and receiver sites by their Geographic location coordinates [Latitude, Longitude]. Antenna type, height, and angle would be configurable in this toolbox as well as system loss, transmitting frequency and power, also receiver Receiver Sensitivity. After developing the environment, this toolbox can visualize transmitter and receiver sites on the map. Different site distances, angles, and elevation will be available as well as their coverage, radiation pattern, and signal strength.

#### Simulation Setup

In the simulation setup, we pick up 7 base stations and 100 mobile devices.

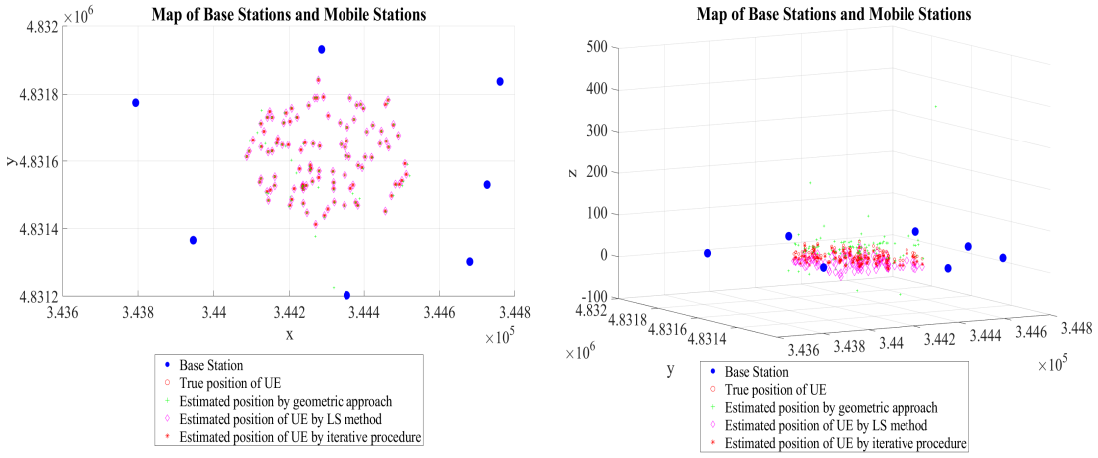
Table 3.1 demonstrates the spherical coordinates of the 7 base stations in the golf course in front of EURECOM, which are illustrated in Fig. 3.8.

With each point, we convert from a spherical coordinate system to Universal Transverse Mercator (UTM) coordinate system [32]. The coordinates in UTM are used in position estimation.

The positions of base stations, the true positions and estimated positions by different algorithms of the mobile devices are illustrated by a view from to (Fig. 3.9a) and a view from one side (Fig. 3.9b). In both figures, the coordinates of all the base stations are already converted into UTM.

The value $\varepsilon_{\text{TDoA}}$ for the stopping criterion is 0.01.

Figure 3.8: Geographic map of the simulated scenario




(a) View from top

(b) View from one side

Figure 3.9: Map of base stations and random positions of the mobile device

### 3.4.2 Results

We define the term Position Error. The *Geographic Position Error* of the $j$-th mobile position ($\text{GPE}_j$) is the distance between the true position and the estimated position. In addition, the two-dimensional position error is also considered. The *Horizontal Position Error* of the $j$-th mobile position ($\text{HPE}_j$) is defined as this distance with only $x$ and $y$ coordinates are involved.

$$\text{GPE}_j = \sqrt{\left(x^{(j)} - \hat{x}^{(j)}\right)^2 + \left(y^{(j)} - \hat{y}^{(j)}\right)^2 + \left(z^{(j)} - \hat{z}^{(j)}\right)^2} \tag{3.21}$$

$$\text{HPE}_j = \sqrt{\left(x^{(j)} - \hat{x}^{(j)}\right)^2 + \left(y^{(j)} - \hat{y}^{(j)}\right)^2} \tag{3.22}$$

where $\left(x^{(j)}, y^{(j)}, z^{(j)}\right)$ are the true coordinates of the $j$-th mobile device and $\left(\hat{x}^{(j)}, \hat{y}^{(j)}, \hat{z}^{(j)}\right)$ are their estimates.

To analytically compare the quality among of algorithms, we use Geographic Root Mean Square Position Error (GRMSE) and Horizontal Root Mean Square Position Error (HRMSE) which are defined by

$$\text{GRMSE} = \sqrt{\frac{1}{M}\sum_{j=1}^{M}\left(\text{GPE}_j^2\right)} \tag{3.23}$$

$$\text{HRMSE} = \sqrt{\frac{1}{M}\sum_{j=1}^{M}\left(\text{HPE}_j^2\right)} \tag{3.24}$$

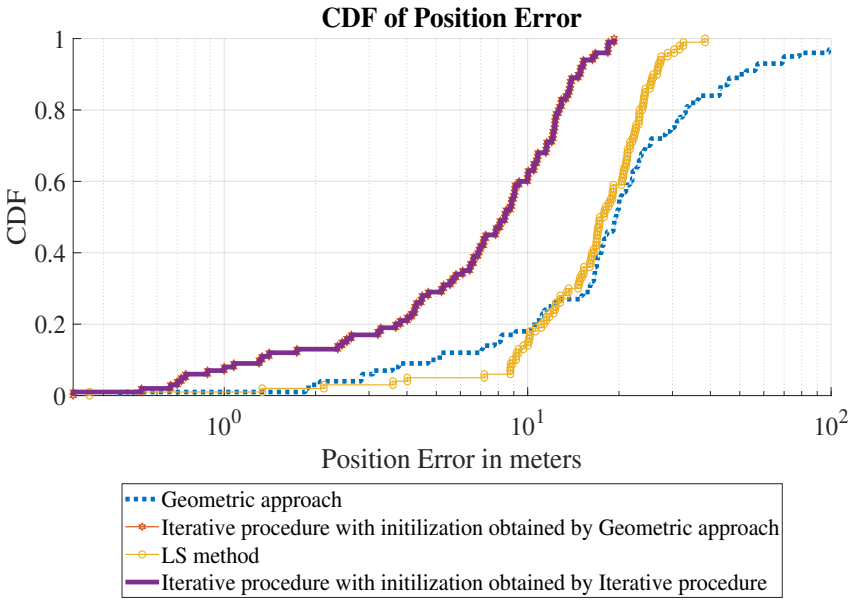
where $M$ is the number of mobile positions in simulations. In our research, $M = 100$.

Table 3.2 illustrates the HRMSE and GRMSE of 4 different algorithms

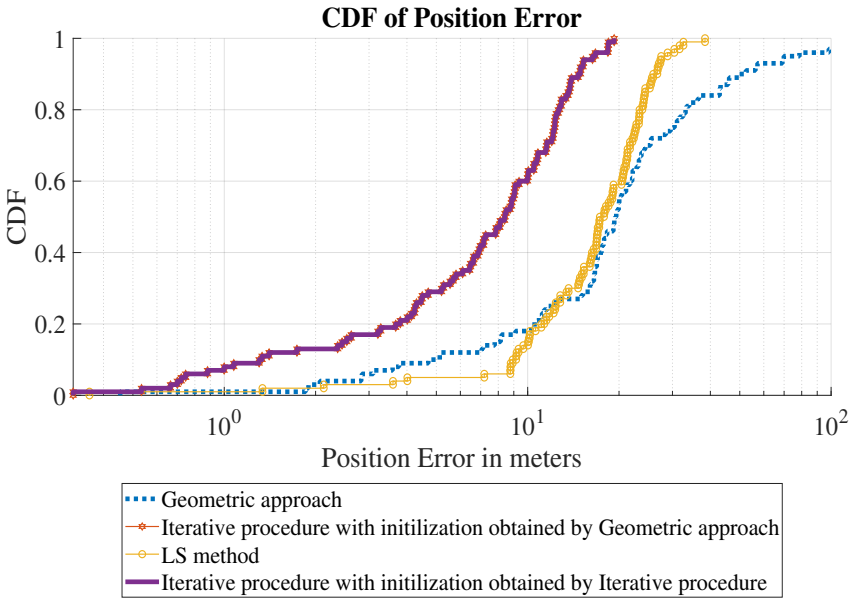
- Geometric approach in section 3.3.2.
- ML estimator with the initialization obtained by geometric approach.
- LS method in section 3.3.3.
- ML estimator with the initialization obtained by LS method.

Fig. 3.10 compares the Cumulative Distribution Functions (CDFs) of HPE and GPE of the 4 algorithms above.

It is obvious that the LS method gives a more accurate position estimation than the geometric approach. The ML estimator with initialization obtained from the LS method also has a better performance. The Position Error is noticeable escalated with the existence of the $z$-coordinate. This explains the Fig. 3.9.




(a) Horizontal Position Error




(b) Geographic Position Error

Figure 3.10: CDF comparisons

| Algorithm | HRMSE | GRMSE |
|---|---|---|
| Geometric approach | 25.4859 | 57.9900 |
| Iterative procedure with initialization obtained by Geometric approach | 0.5306 | 26.2188 |
| LS method | 0.3696 | 19.0952 |
| Iterative procedure with initialization obtained by LS method | 0.3696 | 9.6546 |

Table 3.2: RMSE comparisons of the algorithms

## 3.5 Conclusions

This chapter robustifies the accuracy of TDoA-based localization by a proposed ML estimator, whose performance is also facilitated by a more precise initialization acquired from the LS method. All the results are based on data obtained from realistic ray-tracing simulations, integrated in a MatLab environment.

Practical implementation is currently being carried out, to evaluate all the related positioning algorithms as realistically and objectively as possible.

# Chapter 4

# Direction-based localization at network of base stations

Direction of arrival (DoA) estimation is crucial to improve communications systems' performance, leading to much more accurate results in localization, one of the most vital applications in the Internet of Things (IoT). Unlike the range-based ones, the direction-based positioning algorithms estimate the unknown position by the measured angles whose values must be predefined in an interval of $2\pi$-length. Noisy measurements with values near the edges of this interval can lead to drastic estimation errors, making the convergence of iterative procedures much more challenging. In this chapter, we propose a Maximum Likelihood (ML) estimator, which applies iterative procedures for position estimation. Our procedure is based on the atan2 function, which has the $2\pi$-long codomain to map the DoA. Moreover, a novel mechanism to make the estimation near the edges much more robust, phase jump corrections are proposed to rectify the final estimates. In addition, a new approximate ML estimator, where the effects of approximately normal distributed DoA estimation errors are limited to first-order perturbations, is also introduced. Outputs of this approximate estimator help to enhance the accuracy of the true ML estimator. Simulation results show significant performance improvements.

## 4.1 Introduction

DoA estimation schemes are usually thought of as computationally expensive. However, recent developments propose computationally simple DoA estimation schemes that enable small antenna arrays with a reduced number of elements [24].

Direction-based localization computes the coordinate of the mobile device based on the direction of incident waves to base stations. The numerical expression of this direction is the trigonometric angle between the $x$-direction and the wave (Fig. 4.1). To avoid confusion in measuring angle, all the angles' value must be defined in an interval whose length is $2\pi$. At the bound of the interval, the DoA is very sensitive to noise. On condition that the DoA's set of definition is $[0; 2\pi)$, when the true value of an angle is $\widehat{aOb} = \varepsilon_a$, a small noise of $\varepsilon_a + \varepsilon_b$ ($\varepsilon_a$ and $\varepsilon_b$ are small positive values) can make the

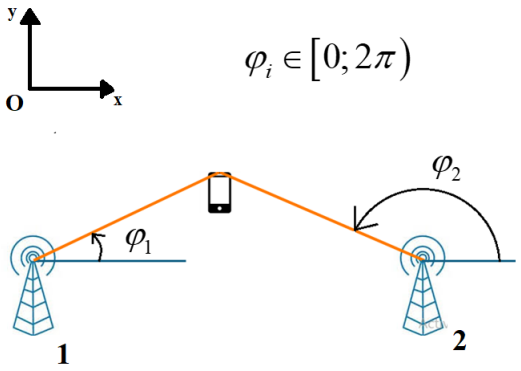



Figure 4.1: DoA-based localization in 2D scenarios

angle's measured value $\widehat{aOb'} = 2\pi - \varepsilon_b$ (Fig. 4.2).

Related papers [33–36] about 2D DoA-based localization use arctan function to define DoA; meanwhile the codomain of that function is $[-\pi/2;\ \pi/2]$ (we assume that arctan $(-\infty) = -\pi/2$ and arctan $(+\infty) = \pi/2$). This codomain does not cover all the possible values of an angle. In the paper [36], a ML estimator is proposed to optimize the positioning. However, the sensitivity to noise of a value near the bound of the set of definition is not well considered. In addition, the localization is studied when all the DoA measurements are considered to be distributed with one common variance.

Our contributions in this chapter are:

- Expressing location in terms of the atan2 function; introducing and optimizing associated phase wrapping correction terms for ML estimator in **section 4.2**. In the definition of DoA, the atan2 function is utilized instead of arctan function. Furthermore, phase jump corrections are added in estimating the estimated DoA to avoid possible huge computing errors caused by small mistakes in practical measurements. Evaluations on the effect of the phase jump corrections are carefully analyzed.

- Propose an approximate ML estimator for DoA-based localization (**section 4.5**).

- In **section 4.6**, we analyse the true ML estimator with the new DoA definition and the additional correction proposed in section 4.2.

In the **last two sections**, we illustrate the simulations and then compare our results to the related results, in order to prove the superiority of our proposed algorithm.

## 4.2 Definition of DoA

Let $(x, y, z)$ be the coordinates of the mobile device and $(x_i, y_i, z_i)$ the coordinates of the $i$-th base station. An azimuth angle is a horizontal angle measured anticlockwise

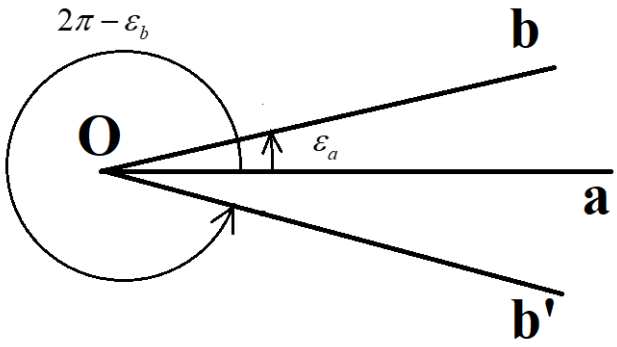



Figure 4.2: Sensitivity to noise of an angle's measured value

from the $x$-direction. Therefore, its value is in the interval $[0; 2\pi)$. We then have the true azimuth of DoA of the signal to the $i$-th base station:

$$\varphi_i = \text{mod}(\text{atan2}(y - y_i, x - x_i), 2\pi) \tag{4.1}$$

Since the codomain of atan2 function is $(-\pi; \pi]$, a modulo operation with the divisor of $2\pi$ is applied to make the true value $\varphi_i$ in the interval $[0; 2\pi)$.

In practical measurements, there are always errors in azimuth estimations. When the size of the angle is near to 0 or $2\pi$, the measured value is very sensitive to noise (a small change in noise can cause a big difference in measured value). To avoid this unexpected difference, a phase jump correction is applied. Consequently, the measured value of $i$-th azimuth angle can be expressed as:

$$\hat{\varphi}_i = \varphi_i + \tilde{\varphi}_i + k_i 2\pi \tag{4.2}$$

where $\tilde{\varphi}_i$ is the error in azimuth estimation. We name the action of adding the phase jump correction of $k_i 2\pi$ as **k-correction**. We then have the expression of $k_i$.

$$k_i = \begin{cases} 1 & , \varphi_i + \tilde{\varphi}_i < 0 \\ -1 & , \varphi_i + \tilde{\varphi}_i \geq 2\pi \\ 0 & \text{otherwise.} \end{cases} \tag{4.3}$$

Thanks to the **k-correction**, the estimated azimuth angle is also in the interval: $0 \leq \hat{\varphi}_i < 2\pi$.

As for the $i$-th elevation angle, its true value is expressed as:

$$\theta_i = \arctan \frac{z - z_i}{\sqrt{(x - x_i)^2 + (y - y_i)^2}} \tag{4.4}$$

The true elevation angle is in the range of $[-\pi/2; \pi/2]$, or $-\pi/2 \leq \theta_i \leq \pi/2$. Its measured value is

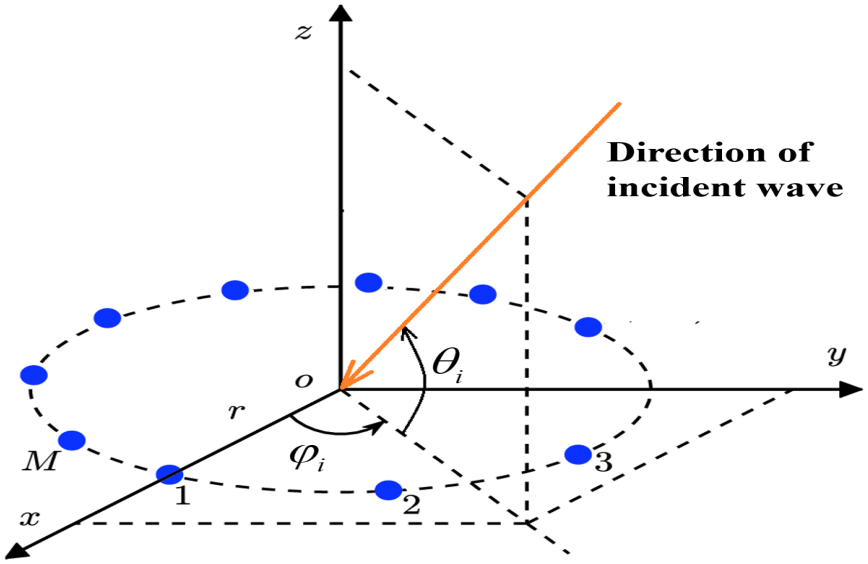



Figure 4.3: $M$-element Uniform Circular Array (UCA) with radius $r$ at the $i$-th base station to estimate azimuth angle $\varphi_i$ and elevation angle $\theta_i$

$$\hat{\theta}_i = \theta_i + \tilde{\theta}_i \tag{4.5}$$

where $\tilde{\theta}_i$ is the error in elevation estimation. When $\theta_i$ is far enough from the 2 boundaries of the interval $[-\pi/2, \pi/2]$ and $\tilde{\theta}_i$ is small enough, the estimated value $\hat{\theta}_i$ of the elevation angle can be considered to be also in this interval: $-\pi/2 \leq \hat{\theta}_i \leq \pi/2$.

At each base station, an $M$-element Uniform Circular Array (UCA) is installed to estimate the azimuth angle and the elevation angle of the incident wave (Fig. 4.3). In [37], it is proved that if noises in received signals are Gaussian distributed, $\tilde{\varphi}_i$ and $\tilde{\theta}_i$ will be asymptotically independently Gaussian distributed with zero-mean. Their variances are $\sigma^2_{az,i}$ and $\sigma^2_{el,i}$, correspondingly.

Since all $\tilde{\varphi}_i$ and $\tilde{\theta}_i$ are independent, we have covariance matrix of the noise vector $\boldsymbol{n}$:

$$\boldsymbol{C} = E\{\boldsymbol{n}\boldsymbol{n}^T\} = \text{diag}\{\sigma^2_{az,1}, \ldots, \sigma^2_{az,N}, \sigma^2_{el,1}, \ldots, \sigma^2_{el,N}\} \tag{4.6}$$

where $\boldsymbol{n} = [\tilde{\varphi}_1 \;\; \tilde{\varphi}_2 \;\; \ldots \;\; \tilde{\varphi}_N \;\; \tilde{\theta}_1 \;\; \tilde{\theta}_2 \;\; \ldots \;\; \tilde{\theta}_N]^T$ and $N$ is the number of base stations.

## 4.3 Probability analysis

### 4.3.1 Azimuth angle

We evaluate the effect of the phase jump corrections on localization. A phase jump correction is meaningful when it is non-zero. Let $\rho(\varphi_i)$ be the probability of that event, at the argument $\varphi_i$ . We have

$$\begin{aligned}\rho(\varphi_i) &= p(\tilde{\varphi}_i < -\varphi_i) + p(\tilde{\varphi}_i \geq 2\pi - \varphi_i) \\ &= \Phi\left(\frac{-\varphi_i}{\sigma_{az,i}}\right) + 1 - \Phi\left(\frac{2\pi - \varphi_i}{\sigma_{az,i}}\right)\end{aligned} \tag{4.7}$$

where $\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_0^x e^{-t^2/2}\, dt$

Since $\varphi_i$ is in $[0, 2\pi)$, the probability $p_{az,i}$ that the phase jump correction is non-zero is

$$p_{az,i} = \frac{1}{2\pi} \int_0^{2\pi} \rho(\varphi_i)\, d\varphi_i \tag{4.8}$$

In [38], it is approximated that $\Phi(x) \approx \dfrac{e^{2\sqrt{\frac{2}{\pi}}x}}{1 + e^{2\sqrt{\frac{2}{\pi}}x}}$ when $x$ is very large.

Thus, when $\sigma_{az,i}$ is small enough

$$p_{az,i} \approx \frac{1}{2\pi} \int_0^{2\pi} \left( \frac{e^{2\sqrt{\frac{2}{\pi}}\frac{-\varphi_i}{\sigma_{az,i}}}}{1 + e^{2\sqrt{\frac{2}{\pi}}\frac{-\varphi_i}{\sigma_{az,i}}}} + 1 - \frac{e^{2\sqrt{\frac{2}{\pi}}\frac{2\pi-\varphi_i}{\sigma_{az,i}}}}{1 + e^{2\sqrt{\frac{2}{\pi}}\frac{2\pi-\varphi_i}{\sigma_{az,i}}}} \right) d\varphi_i \tag{4.9}$$

$$p_{az,i} \approx \sigma_{az,i} \frac{\ln 2 - \ln\left(e^{\frac{4\sqrt{2\pi}}{\sigma_{az,i}}} + 1\right)}{2\sqrt{2\pi}} + 2 \tag{4.10}$$

Since $\sigma_{az,i}$ is small, we approximate

$$\ln\left(e^{\frac{4\sqrt{2\pi}}{\sigma_{az,i}}} + 1\right) \approx \ln\left(e^{\frac{4\sqrt{2\pi}}{\sigma_{az,i}}}\right) = \frac{4\sqrt{2\pi}}{\sigma_{az,i}} \tag{4.11}$$

As a result, (4.10) and (4.11) give an approximation:

$$p_{az,i} \approx \sigma_{az,i} \frac{\ln 2 - \dfrac{4\sqrt{2\pi}}{\sigma_{az,i}}}{2\sqrt{2\pi}} + 2 = \sigma_{az,i} \frac{\ln 2}{2\sqrt{2\pi}} \tag{4.12}$$

Therefore, when $\sigma_{az,i}$ is small enough, $p_{az,i}$ is proportional to $\sigma_{az,i}$ with the coefficient of $\frac{\ln 2}{2\sqrt{2\pi}}$. The network has multiple base stations, so the probability that **at least** one phase jump correction is non-zero is much higher in practical localization.

### 4.3.2 Elevation angle

We compute the probability that $\hat{\theta}_i < -\pi/2$ or $\hat{\theta}_i > \pi/2$. It is equivalent to $\tilde{\theta}_i < -\pi/2 - \theta_i$ or $\tilde{\theta}_i > \pi/2 - \theta_i$. Let $\rho(\theta_i)$ be the probability of that event, at the argument $\theta_i$ . We have

$$\begin{aligned}\rho(\theta_i) &= p(\tilde{\theta}_i < -\pi/2 - \theta_i) + p(\tilde{\theta}_i > \pi/2 - \theta_i) \\ &= \Phi\left(\frac{-\pi/2 - \theta_i}{\sigma_{el,i}}\right) + 1 - \Phi\left(\frac{\pi/2 - \theta_i}{\sigma_{el,i}}\right)\end{aligned} \tag{4.13}$$

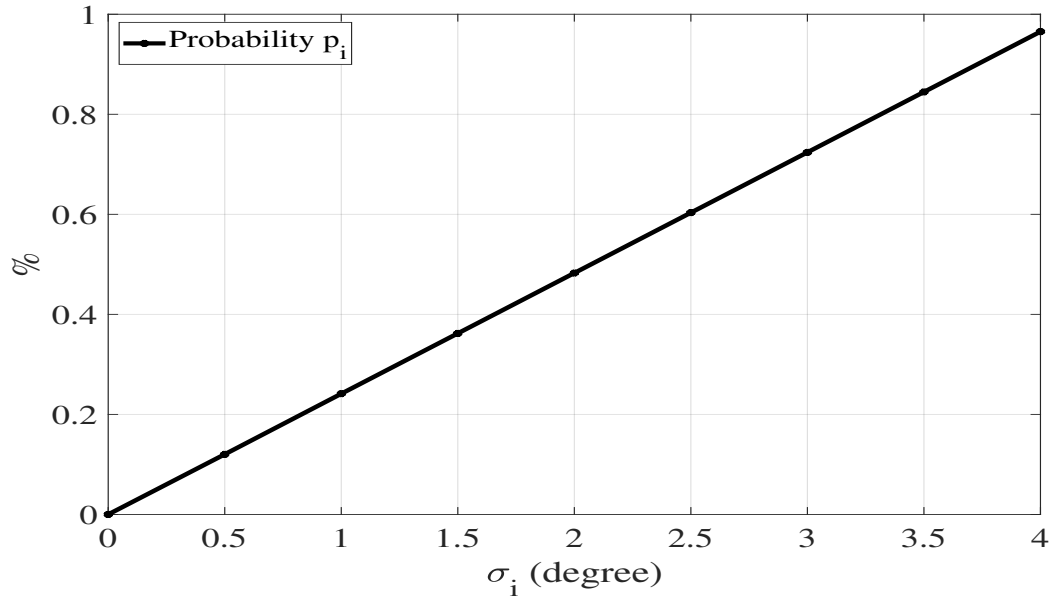



Figure 4.4: Probability (%) that a phase jump correction is non-zero, in terms of $\sigma_{az,i}$, when $\sigma_{az,i}$ is small.

When $\sigma_{el,i}$ is small enough, it is approximated that

$$p_{el,i} \approx \frac{1}{\pi} \int_{-a}^{a} \left( \frac{e^{2\sqrt{\frac{2}{\pi}} \frac{-\pi/2-\theta_i}{\sigma_{el,i}}}}{1+e^{2\sqrt{\frac{2}{\pi}} \frac{-\pi/2-\theta_i}{\sigma_{el,i}}}} + 1 - \frac{e^{2\sqrt{\frac{2}{\pi}} \frac{\pi/2-\theta_i}{\sigma_{el,i}}}}{1+e^{2\sqrt{\frac{2}{\pi}} \frac{\pi/2-\varphi_i}{\sigma_{el,i}}}} \right) d\theta_i \tag{4.14}$$

$$p_{el,i} \approx \sigma_{el,i} \frac{\ln\left(1+e^{\frac{-2\sqrt{2}a}{\sqrt{\pi}\sigma_{el,i}} + \frac{\sqrt{2\pi}}{\sigma_{el,i}}}\right) - \ln\left(e^{\frac{(2a+\pi)\sqrt{2}}{\sqrt{\pi}\sigma_{el,i}}} + 1\right)}{\sqrt{2\pi}} + \frac{4a}{\pi} \tag{4.15}$$

Since $\sigma_{az,i}$ is small, we approximate

$$\ln\left(e^{\frac{(2a+\pi)\sqrt{2}}{\sqrt{\pi}\sigma_{el,i}}} + 1\right) \approx \ln\left(e^{\frac{(2a+\pi)\sqrt{2}}{\sqrt{\pi}\sigma_{el,i}}}\right) = \frac{(2a+\pi)\sqrt{2}}{\sqrt{\pi}\sigma_{el,i}} \tag{4.16}$$

As a result, (4.15) and (4.16) give an approximation:

$$p_{el,i} \approx \sigma_{el,i} \frac{\ln\left(1+e^{\frac{-2\sqrt{2}a}{\sqrt{\pi}\sigma_{el,i}} + \frac{\sqrt{2\pi}}{\sigma_{el,i}}}\right) - \frac{(2a+\pi)\sqrt{2}}{\sqrt{\pi}\sigma_{el,i}}}{\sqrt{2\pi}} + \frac{4a}{\pi} \tag{4.17}$$

$$p_{el,i} \approx \frac{2a}{\pi} - 1 + \frac{\sigma_{el,i}}{\sqrt{2\pi}} \ln\left(1+e^{\frac{-2\sqrt{2}a}{\sqrt{\pi}\sigma_{el,i}} + \frac{\sqrt{2\pi}}{\sigma_{el,i}}}\right) \tag{4.18}$$

$$p_{el,i} \approx \frac{\sigma_{el,i}}{\sqrt{2\pi}} \ln \frac{1 + e^{\frac{-2\sqrt{2}a}{\sqrt{\pi}\sigma_{el,i}} + \frac{\sqrt{2\pi}}{\sigma_{el,i}}}}{e^{\frac{-2\sqrt{2}a}{\sqrt{\pi}\sigma_{el,i}} + \frac{\sqrt{2\pi}}{\sigma_{el,i}}}} \tag{4.19}$$

Considering that $\sigma_{el,i} \leq 5\pi/180$
$p_{el,i} \leq 10^{-6} \iff a \leq \pi/2.8$

## 4.4 Estimating position by Least Squares method

From equation (4.1), we have

$$\tan\varphi_i = \frac{y - y_i}{x - x_i} \tag{4.20}$$

$$x\sin\varphi_i - y\cos\varphi_i = x_i\sin\varphi_i - y_i\cos\varphi_i \tag{4.21}$$

As $\tilde{\varphi}_i$ is very small, we approximate that $\sin\tilde{\varphi}_i \approx 0$ and $\cos\tilde{\varphi}_i \approx 1$. Thus

$$\sin\varphi_i = \sin(\hat{\varphi}_i - \tilde{\varphi}_i - k_i 2\pi) = \sin(\hat{\varphi}_i - \tilde{\varphi}_i) \approx \sin\hat{\varphi}_i \tag{4.22}$$

$$\cos\varphi_i = \cos(\hat{\varphi}_i - \tilde{\varphi}_i - k_i 2\pi) = \cos(\hat{\varphi}_i - \tilde{\varphi}_i) \approx \cos\hat{\varphi}_i \tag{4.23}$$

Hence, from (4.21), it is approximated that

$$x\sin\hat{\varphi}_i - y\cos\hat{\varphi}_i = x_i\sin\hat{\varphi}_i - y_i\cos\hat{\varphi}_i \tag{4.24}$$

From equation (4.4), we have

$$\tan\theta_i = \frac{z - z_i}{\sqrt{(x - x_i)^2 + (y - y_i)^2}} = \frac{(z - z_i)\cos\varphi_i}{x - x_i} \tag{4.25}$$

As $\tilde{\theta}_i$ is very small, we approximate that $\sin\tilde{\theta}_i \approx 0$ and $\cos\tilde{\varphi}_i \approx 1$, so $\tan\tilde{\theta}_i \approx 0$. Thus $\tan\theta_i \approx \tan\hat{\theta}_i$

We have the approximation

$$\tan\hat{\theta}_i = \frac{(z - z_i)\cos\hat{\varphi}_i}{x - x_i} \tag{4.26}$$

$$x\tan\hat{\theta}_i - z\cos\hat{\varphi}_i = x_i\tan\hat{\theta}_i - z_i\cos\hat{\varphi}_i \tag{4.27}$$

In matrix formulation, we define

$$\hat{\boldsymbol{A}} = \begin{bmatrix} \sin\hat{\varphi}_1 & -\cos\hat{\varphi}_1 & 0 \\ \sin\hat{\varphi}_2 & -\cos\hat{\varphi}_2 & 0 \\ \cdots & \cdots & \cdots \\ \sin\hat{\varphi}_N & -\cos\hat{\varphi}_N & 0 \\ \tan\hat{\theta}_1 & 0 & -\cos\hat{\varphi}_1 \\ \tan\hat{\theta}_2 & 0 & -\cos\hat{\varphi}_2 \\ \cdots & \cdots & \cdots \\ \tan\hat{\theta}_N & 0 & -\cos\hat{\varphi}_N \end{bmatrix} ; \quad \hat{\boldsymbol{b}} = \begin{bmatrix} x_1\sin\hat{\varphi}_1 - y_1\cos\hat{\varphi}_1 \\ x_2\sin\hat{\varphi}_2 - y_2\cos\hat{\varphi}_2 \\ \cdots \\ x_N\sin\hat{\varphi}_N - y_N\cos\hat{\varphi}_N \\ x_1\tan\hat{\theta}_1 - z_1\cos\hat{\varphi}_1 \\ x_2\tan\hat{\theta}_2 - z_2\cos\hat{\varphi}_2 \\ \cdots \\ x_N\tan\hat{\theta}_N - z_N\cos\hat{\varphi}_N \end{bmatrix}$$

$$\boldsymbol{x} = \begin{bmatrix} x & y & z \end{bmatrix}^T$$

We then have the equation of approximation

$$\hat{\boldsymbol{A}}\,\boldsymbol{x} = \hat{\boldsymbol{b}} \tag{4.28}$$

Therefore, the estimate of $\boldsymbol{x}$ is

$$\hat{\boldsymbol{x}} = \min_{\boldsymbol{x}} \|\hat{\boldsymbol{A}}\boldsymbol{x} - \hat{\boldsymbol{b}}\|^2 \tag{4.29}$$

$\hat{\boldsymbol{x}}$ is calculated by Least-Square estimation of $\boldsymbol{x}$

$$\hat{\boldsymbol{x}} = \hat{\boldsymbol{A}}^{\dagger}\hat{\boldsymbol{b}} \tag{4.30}$$

where $\boldsymbol{A}^{\dagger} = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T$ is the Moore-Penrose pseudo inverse of matrix $\boldsymbol{A}$.

For a more accurate estimation of the mobile's position, this estimate is taken as the initialization of an iterative procedure, which will be discussed in the following section.

## 4.5 Optimizing position estimation by an approximate Maximum Likelihood estimator

Unlike Least Squares method, in this approximate estimator, we assume that the effect of approximately Gaussian DoA estimation errors $\tilde{\varphi}_i$ and $\tilde{\theta}_i$ can be limited to first-order perturbations, so we have the approximations: $\sin\tilde{\varphi}_i \approx \tilde{\varphi}_i$; $\cos\tilde{\varphi}_i \approx 1$ and $\tan\tilde{\theta}_i \approx \tilde{\theta}_i$. As a result, we have:

$$\sin\varphi_i = \sin(\hat{\varphi}_i - \tilde{\varphi}_i - k_i 2\pi) = \sin(\hat{\varphi}_i - \tilde{\varphi}_i) \approx \sin\hat{\varphi}_i - \tilde{\varphi}_i\cos\hat{\varphi}_i \tag{4.31}$$

$$\cos\varphi_i = \cos(\hat{\varphi}_i - \tilde{\varphi}_i - k_i 2\pi) = \cos(\hat{\varphi}_i - \tilde{\varphi}_i) \approx \cos\hat{\varphi}_i + \tilde{\varphi}_i\sin\hat{\varphi}_i \tag{4.32}$$

Therefore, from (4.21), we have the following approximations:

$$(x - x_i)(\sin\hat{\varphi}_i - \tilde{\varphi}_i\cos\hat{\varphi}_i) - (y - y_i)(\cos\hat{\varphi}_i + \tilde{\varphi}_i\sin\hat{\varphi}_i) \approx 0 \tag{4.33}$$

$$\tilde{\varphi}_i \approx \frac{-(x-x_i)\sin\hat{\varphi}_i + (y-y_i)\cos\hat{\varphi}_i}{-(x-x_i)\cos\hat{\varphi}_i - (y-y_i)\sin\hat{\varphi}_i} \tag{4.34}$$

$$\begin{aligned}
&\cos\varphi_i \approx \cos\hat{\varphi}_i + \tilde{\varphi}_i \sin\hat{\varphi}_i \\
&= \frac{-(x-x_i)\cos^2\hat{\varphi}_i - (y-y_i)\sin\hat{\varphi}_i\cos\hat{\varphi}_i}{-(x-x_i)\cos\hat{\varphi}_i - (y-y_i)\sin\hat{\varphi}_i} + \frac{-(x-x_i)\sin^2\hat{\varphi}_i + (y-y_i)\sin\hat{\varphi}_i\cos\hat{\varphi}_i}{-(x-x_i)\cos\hat{\varphi}_i - (y-y_i)\sin\hat{\varphi}_i} \\
&= \frac{-(x-x_i)}{-(x-x_i)\cos\hat{\varphi}_i - (y-y_i)\sin\hat{\varphi}_i} = \frac{x-x_i}{(x-x_i)\cos\hat{\varphi}_i + (y-y_i)\sin\hat{\varphi}_i}
\end{aligned} \tag{4.35}$$

$$\tan\theta_i = \frac{(z-z_i)\cos\varphi_i}{x-x_i} \approx \frac{z-z_i}{(x-x_i)\cos\hat{\varphi}_i + (y-y_i)\sin\hat{\varphi}_i} \tag{4.36}$$

$$\tilde{\theta}_i \approx \tan\tilde{\theta}_i = \tan(\hat{\theta}_i - \theta_i) = \frac{\tan\hat{\theta}_i - \dfrac{z-z_i}{(x-x_i)\cos\hat{\varphi}_i + (y-y_i)\sin\hat{\varphi}_i}}{1 + \tan\hat{\theta}_i \dfrac{z-z_i}{(x-x_i)\cos\hat{\varphi}_i + (y-y_i)\sin\hat{\varphi}_i}} \tag{4.37}$$

$$\tilde{\theta}_i \approx \frac{(x-x_i)\cos\hat{\varphi}_i\tan\hat{\theta}_i + (y-y_i)\sin\hat{\varphi}_i\tan\hat{\theta}_i - (z-z_i)}{(x-x_i)\cos\hat{\varphi}_i + (y-y_i)\sin\hat{\varphi}_i + (z-z_i)\tan\hat{\theta}_i} \tag{4.38}$$

The Cost Function of this approximate ML estimator is expressed as

$$L = \sum_{i=1}^{N}\left(\frac{\tilde{\varphi}_i^2}{\sigma_{az,i}^2} + \frac{\tilde{\theta}_i^2}{\sigma_{el,i}^2}\right) \tag{4.39}$$

where $\tilde{\varphi}_i$ and $\tilde{\theta}_i$ are approximated in (4.34) and (4.38), respectively.

Our task is to find an estimate $\hat{\boldsymbol{x}}$ that minimizes the Cost Function, or

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} L \tag{4.40}$$

Finding $\hat{\boldsymbol{x}} = [\hat{x}\ \ \hat{y}\ \ \hat{z}]^T$ is really a challenging task. A solution is to use a local search, which is an iterative algorithm requiring an initial position estimate. We consider Gauss Newton algorithm [22] for $\hat{\boldsymbol{x}}$. At the iteration $(u+1)$:

$$\hat{\boldsymbol{x}}^{(u+1)} = \hat{\boldsymbol{x}}^{(u)} - (\boldsymbol{G}_n^T(\hat{\boldsymbol{x}}^{(u)})\boldsymbol{G}_n(\hat{\boldsymbol{x}}^{(u)}))^{-1}\boldsymbol{G}_n^T(\hat{\boldsymbol{x}}^{(u)})\boldsymbol{n}^T \tag{4.41}$$

where $\hat{\boldsymbol{x}}^{(u)}$ is the estimated coordinate vector of the mobile at the $u$-th iteration. $\boldsymbol{G}_n(\boldsymbol{x})$ is the following Jacobian matrix:

$$\boldsymbol{G}_n(\boldsymbol{x}) = \frac{\partial \boldsymbol{n}}{\partial \boldsymbol{x}^T} \tag{4.42}$$

The procedure is expected to terminate when $\|\hat{\boldsymbol{x}}^{(u+1)} - \hat{\boldsymbol{x}}^{(u)}\|_2 < \varepsilon_1$, for the stopping criterion $\varepsilon_1$ sufficiently small. Then, the final position of the procedure is considered to be the coordinates of the mobile device in the $xy$ plane. However, the iterative procedures do not always converge. The possible outcomes of an iterative procedure and how the estimated position of the mobile device is taken from that procedure are carefully analyzed in section 3.3.4.

The Algorithm 2 illustrates the Gauss-Newton iterative procedure of the approximate ML estimator.

**Algorithm 2:** Proposed Approximate ML Estimator

Take the measured Direction of Arrival $\hat{\varphi}_i$.
Assign $u = 1$ and $\varepsilon_1$ sufficiently small.
Assign the coordinate vector computed by (4.30) as the first estimated coordinate vector $\hat{\boldsymbol{x}}^{(1)}$ of the mobile device.
**repeat**
    Compute the following estimated coordinate vector $\hat{\boldsymbol{x}}^{(u+1)}$ of the mobile device by (4.41).
    $u = u + 1$;
**until** $\|\hat{\boldsymbol{x}}^{(u+1)} - \hat{\boldsymbol{x}}^{(u)}\|_2 < \varepsilon$ *or* $u > 1000$ *or* $\|\hat{\boldsymbol{x}}^{(u+1)}\|_2 = \pm\infty$;
**if** $u > 1000$ *or* $\|\hat{\boldsymbol{x}}^{(u+1)}\|_2 = \pm\infty$ **then**
    $\hat{\boldsymbol{x}}^{(1)}$ is the estimated position of the mobile device;
**else**
    $\hat{\boldsymbol{x}}^{(u)}$ is the estimated position of the mobile device;

## 4.6 Optimizing position estimation by the true Maximum Likelihood estimator

In this section, we apply an iterative Maximum Likelihood estimator, to optimize $\hat{\boldsymbol{x}}$ obtained in (4.30),

In vector form, we denote

$$\hat{\boldsymbol{\varphi}} = \begin{bmatrix} \hat{\varphi}_1 & \dots & \hat{\varphi}_N & \hat{\theta}_1 & \dots & \hat{\theta}_N \end{bmatrix}^T \tag{4.43}$$

$$\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{k}) = \begin{bmatrix} \varphi_1(\boldsymbol{x}) + k_1 2\pi \\ \varphi_2(\boldsymbol{x}) + k_2 2\pi \\ \dots \\ \varphi_N(\boldsymbol{x}) + k_N 2\pi \\ \theta_1(\boldsymbol{x}) \\ \theta_2(\boldsymbol{x}) \\ \dots \\ \theta_N(\boldsymbol{x}) \end{bmatrix} \tag{4.44}$$

where $\boldsymbol{k} = [k_1 \ k_2 \ \cdots \ k_N]^T$; $\varphi_i(\boldsymbol{x})$ and $\theta_i(\boldsymbol{x})$ are the estimated azimuth and elevation angles, respectively, depending on $\boldsymbol{x} = [x \ \ y \ \ z]^T$ and are computed by

$$\varphi_i(\boldsymbol{x}) = \mathrm{mod}(\mathrm{atan2}(x - x_i, \ y - y_i), 2\pi) \tag{4.45}$$

$$\theta_i(\boldsymbol{x}) = \arctan \frac{z - z_i}{\sqrt{(x - x_i)^2 + (y - y_i)^2}} \tag{4.46}$$

Treating the phase shift vector $\boldsymbol{k}$ as unknown parameters and ignoring their dependence on the noise, the measurement vector $\hat{\boldsymbol{\varphi}}$ is Gaussian with mean vector of $\boldsymbol{f}$ and covariance

matrix $\boldsymbol{C}$, we have the probability density function (pdf) [21]:

$$\mathrm{p}(\hat{\boldsymbol{\varphi}}|\boldsymbol{x},\boldsymbol{k}) = \frac{(2\pi)^{-N}}{|\boldsymbol{C}|^{1/2}} \exp\left[\tfrac{-1}{2}(\hat{\boldsymbol{\varphi}}-\boldsymbol{f})^T\boldsymbol{C}^{-1}(\hat{\boldsymbol{\varphi}}-\boldsymbol{f})\right] \tag{4.47}$$

Maximizing the pdf in (4.47) is equivalent to finding

$$\hat{\boldsymbol{x}},\widehat{\boldsymbol{k}} = \arg\min_{\boldsymbol{x},\boldsymbol{k}}(\hat{\boldsymbol{\varphi}} - \boldsymbol{f}(\boldsymbol{x},\boldsymbol{k}))^T\boldsymbol{C}^{-1}(\hat{\boldsymbol{\varphi}} - \boldsymbol{f}(\boldsymbol{x},\boldsymbol{k})) \tag{4.48}$$

which we shall perform alternatingly. We consider Gauss Newton [22] for $\hat{x}$. At the iteration $(u+1)$:

$$\hat{\boldsymbol{x}}^{(u+1)} = \hat{\boldsymbol{x}}^{(u)} + (\boldsymbol{G}^T\boldsymbol{C}^{-1}\boldsymbol{G})^{-1}\boldsymbol{G}^T\boldsymbol{C}^{-1}(\hat{\boldsymbol{\varphi}} - \boldsymbol{f}(\hat{\boldsymbol{x}}^{(u)},\boldsymbol{k}^{(u+1)})) \tag{4.49}$$

where $\boldsymbol{G}$ is the Jacobian matrix.

$$\boldsymbol{G} = \boldsymbol{G}\left(\hat{\boldsymbol{x}}^{(u)},\boldsymbol{k}^{(u+1)}\right),\ \boldsymbol{G}(\boldsymbol{x},\boldsymbol{k}) = \frac{\partial \boldsymbol{f}(\boldsymbol{x},\boldsymbol{k})}{\partial \boldsymbol{x}^T}. \tag{4.50}$$

At this point, it is important to determine the value of $k_i$. As the additive noise in each DoA measurement is unclear, $k_i$ cannot be determined by equation (4.3). From (4.2), we have

$$|\tilde{\varphi}_i| = |\hat{\varphi}_i - \varphi_i(\boldsymbol{x}) - k_i 2\pi| \tag{4.51}$$

We assume $\tilde{\varphi}_i$ small enough, so $|\tilde{\varphi}_i| < \pi$ with the probability almost 1. Thus $\widehat{k}_i$ can be estimated by

$$\hat{k}_i^{(u+1)} = \arg\min_{k_i\in\{0;\pm1\}} |\hat{\varphi}_i - \varphi_i(\hat{\boldsymbol{x}}^{(u)}) - k_i 2\pi| \tag{4.52}$$

where $\hat{\boldsymbol{x}}^{(u)}$ is the estimated coordinate vector of the mobile device at the $u$-th iteration.

The procedure is expected to terminate when $\|\hat{\boldsymbol{x}}^{(u)} - \hat{\boldsymbol{x}}^{(u-1)}\|_2 < \varepsilon$, for the stopping value $\varepsilon$ sufficiently small. However, iterative procedures do not always converge. The possible outcomes of an iterative procedure and how the estimated position of the mobile device is taken from that procedure are carefully analyzed in section 3.3.4.

In summary, the Algorithm 3, a Gauss-Newton iterative procedure of Maximum Likelihood estimator, is proposed.

## 4.7 Simulations and Results

In this section, we simulate the theories in 2D and 3D scenarios. In 2D scenarios, all the data related to elevation elements are removed. The definition of DoA is the azimuth expression.

**Algorithm 3:** Proposed Maximum Likelihood estimator with estimation of $\hat{\boldsymbol{k}}$

Take the measured Direction of Arrival: azimuth $\hat{\varphi}_i$ and elevation $\hat{\theta}_i$.
Assign $u = 1$ and $\varepsilon$ sufficiently small.
Assign a vector as the first estimated coordinate vector $\hat{\boldsymbol{x}}^{(1)}$ of the mobile device. This vector can be the estimate of Least Squares in equation (4.30) **or** the the result of approximate ML by Algorithm 2 in section 4.5.
**repeat**
    Compute the estimated azimuth $\hat{\varphi}_i$ and elevation $\hat{\theta}_i$ by (4.45) and (4.46), respectively.
    **if** $|\varphi_i(\hat{\boldsymbol{x}}^{(u)}) - \hat{\varphi}_i| \geq \pi$ **then**
        $\widehat{k}_i = \text{sign}(\varphi_i(\hat{\boldsymbol{x}}^{(u)}) - \hat{\varphi}_i)$
    **else**
        $\widehat{k}_i = 0$ ;
    Compute the following estimated coordinate vector $\hat{\boldsymbol{x}}^{(u+1)}$ of the mobile device by (4.49).
    $u = u + 1$;
**until** $\|\boldsymbol{x}^{(u)} - \boldsymbol{x}^{(u-1)}\|_2 < \varepsilon$ *or* $u > 1000$ *or* $\|\boldsymbol{x}^{(u)}\|_2 = \pm\infty$;
**if** $u > 1000$ *or* $\|\hat{\boldsymbol{x}}^{(u)}\|_2 = \pm\infty$ **then**
    $\hat{\boldsymbol{x}}^{(1)}$ is the estimated position of the mobile device;
**else**
    $\hat{\boldsymbol{x}}^{(u)}$ is the estimated position of the mobile device;

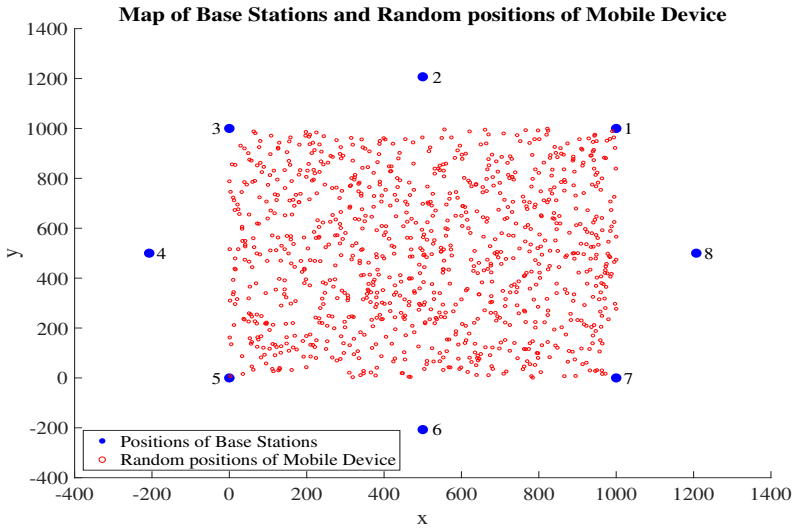



Figure 4.5: Map of base stations' network, random points of the mobile device in 2D.

### 4.7.1 Cramer Rao Bound (CRB) analysis

The Cramer-Rao Bound (CRB) is computed for the quaility evaluation of the algorithm. The Fisher Information Matrix (FIM) is calculated by

$$\boldsymbol{I}(\boldsymbol{x}) = \boldsymbol{G}^T(\boldsymbol{x})\boldsymbol{C}^{-1}\boldsymbol{G}(\boldsymbol{x}) \tag{4.53}$$

The CRB is the trace of the inverse of FIM:

$$\text{CRB} = \text{tr}(\boldsymbol{I}^{-1}) \tag{4.54}$$

To compare the quality among the algorithms and CRB, we use Root Mean Square Position Error (RMSE) which is defined by

$$\text{RMSE} = \sqrt{\text{E}\left(\|\boldsymbol{x} - \boldsymbol{x}\|^2\right)} \tag{4.55}$$

where $\boldsymbol{x}$ is the true position of the mobile device and $\hat{\boldsymbol{x}}$ is its estimate.

### 4.7.2 2D scenarios

#### Simulations

In the $xy$ plane, RMSE averaging is over 1000 mobile positions picked randomly in a square of 1000m x 1000m centered in the BS circle. The network of 8 Base stations (numbered from 1 to 8) forms the circumscribed circle of this square. All the related points are shown in Fig. 4.5. Stopping criteria for the approximate ML and true ML estimators are $\varepsilon_1 = \varepsilon_2 = 0.01$.

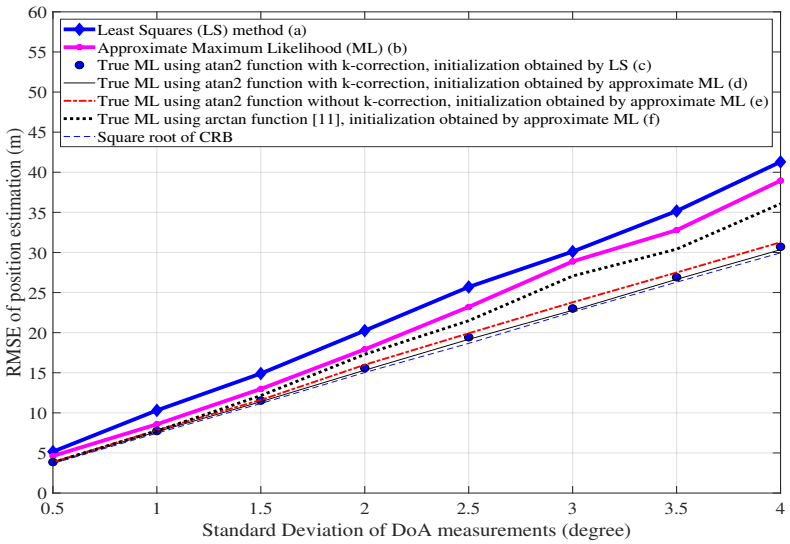
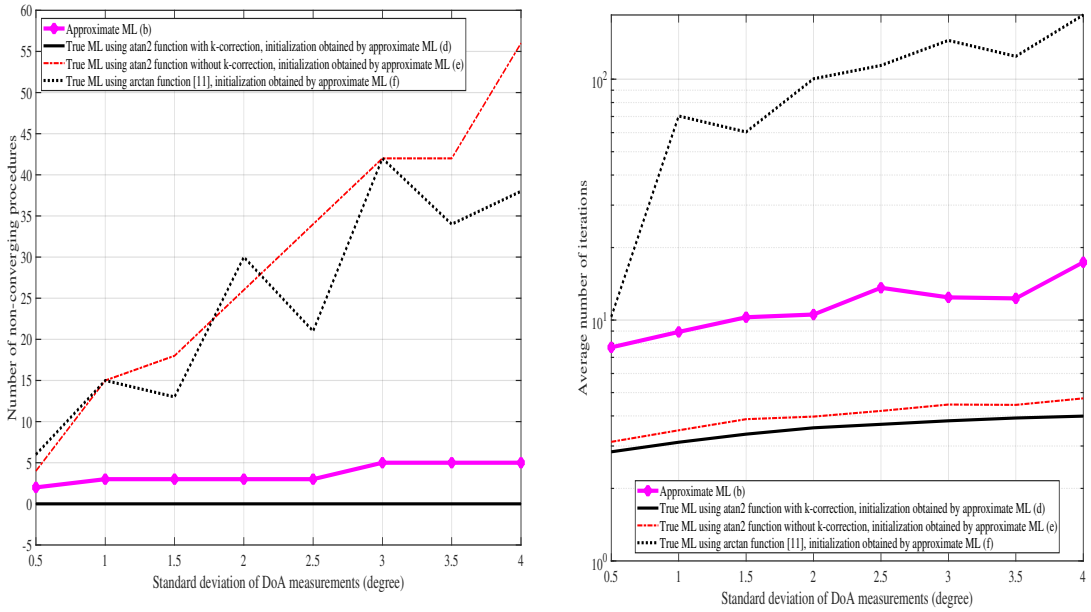



Figure 4.6: Comparisons of RMSEs among the algorithms when $N = 8$ BSs, the standard deviation of DOA measurements varies from 0.5° to 4°.




(a) Number of non-converging procedures out of 1000 testing ones

(b) Average number of iterations

Figure 4.7: Comparisons among the iterative procedures of the algorithms when $N = 8$ BSs, the standard deviation of DOA measurements varies from 0.5° to 4°.

**Results**

Instead of comparing the MSEs to the CRB, we compare their square roots: The Root Mean Square Error (RMSE) = $\sqrt{\text{MSE}}$ and square root of CRB ($\sqrt{\text{CRB}}$). In each scheme, a common standard deviation is assumed for all the DoA measurements. More comprehensively, Fig. 4.6 compares the RMSEs of the 6 positioning algorithms:

(a) Least Squares method shown in section 4.4.

(b) Approximate ML estimator shown in section 4.5, with the initialization obtained by Least Squares method.

(c) True ML estimator (section 4.6), **with** k-correction; the DoA definition using atan2 function. The initialization is obtained by Least Squares method.

(d) True ML estimator (section 4.6), **with** k-correction; the DoA definition using atan2 function. The initialization is obtained by Approximate ML.

(e) True ML estimator (section 4.6), **without** k-correction; the DoA definition using atan2 function. The initialization is obtained by by Approximate ML.

(f) True ML estimator (section 4.6), with the DoA definition using arctan function [36]. The initialization is obtained by the Approximate ML.

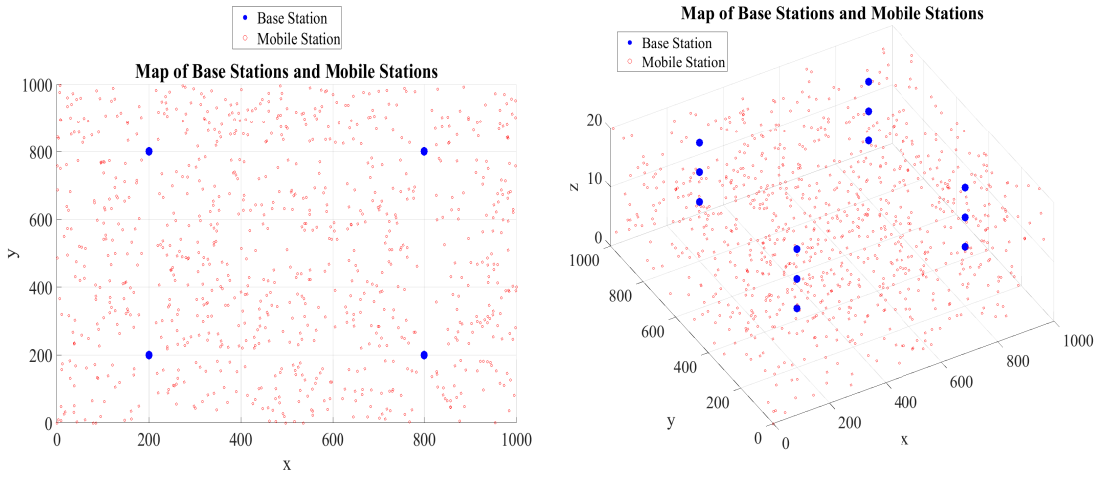
The $\sqrt{\text{CRB}}$ is also added to validate their performances.

Section 3.3.4 introduces 3 possible outcomes of an iterative procedure. We compare the algorithms in terms of the accuracy (evaluated by the RMSEs) and the time delay (evaluated by the average number of iterations). When a procedure is **diverging** or **oscillating**, the initial position is taken as its estimate, which increases the RMSE and thus makes the location less exact. Moreover, an **oscillating** procedure raises remarkably the number of iterations, which makes the localization processes slower. As a result, the figures, which compare the RMSEs, the number of **non-converging** procedures and the average number of iterations among the algorithms, are demonstrated.

Fig. 4.6 compares the RMSEs of the 6 algorithms above in the scenario of 8 base stations and the standard deviation of DoA measurements varies from 0.5° to 4°. RMSE stands for the accuracy in estimation. Furthermore, Fig. 4.7 gives us an overview on how efficient the iterative procedures are in approximate ML and true ML estimators. More specifically, Fig 4.7a illustrates the number of non-converging procedures out of 1000 testing procedures. **Non-converging** procedures are the combination of **diverging** procedures and **oscillating** procedures defined in section 4.5. Our proposed algorithm for true ML estimator has zero non-converging procedure, or in other words, all the procedures converge to local minima. Fig. 4.7b presents the average number of iterations. Our proposed algorithm has the fewest average number of iterations, which reduces the time delay for localization processes.

From the results above, it is obvious that

- The true ML estimator gives the best accurate position estimation, compared to approximate ML estimator and Least Squares method.




(a) View from top

(b) View from one side

Figure 4.8: Map of base stations and random positions of the mobile device in 3D

- In the true ML estimator, the initialization obtained by approximate ML gives a little smaller RMSE than that has initialization obtained by Least Squares. On the other hand, this RMSE is still larger than the $\sqrt{\text{CRB}}$, which assures the unbiased property of the estimator.

- The positioning algorithm with the k-correction and the definition of DoA using atan2 function has the best performance, in both accuracy and time delay, compared to the algorithms, of which the DoA is defined by arctan function, or by atan2 function but no k-correction for the iterative procedures. With this proposed algorithm, the RMSE is lowest but still higher than the $\sqrt{\text{CRB}}$, and the average number of iterations is noticeably smaller than the 2 other algorithms. This is the most important contribution of this section.

In essence, the true ML likelihood estimator, of which the DoA is defined with atan2 function and an addition k-correction is used, is the best estimator. The initialization by an approximate ML can enhance the performance of the true ML.

### 4.7.3 3D scenarios

**Simulations**

We consider an area of 1000m x 1000m with the height of 20m. RMSE averaging is over 1000 mobile positions picked randomly in this space (Fig. 4.8). The center of this space is at the coordinates (500; 500; 10). At the height of 10m, 4 base stations of the coordinates (200; 200; 10), (800; 200; 10), (200; 800; 10) and (800; 800; 10) are placed, which forms a square. To enhance the localization in 3D, two similar sets of base stations

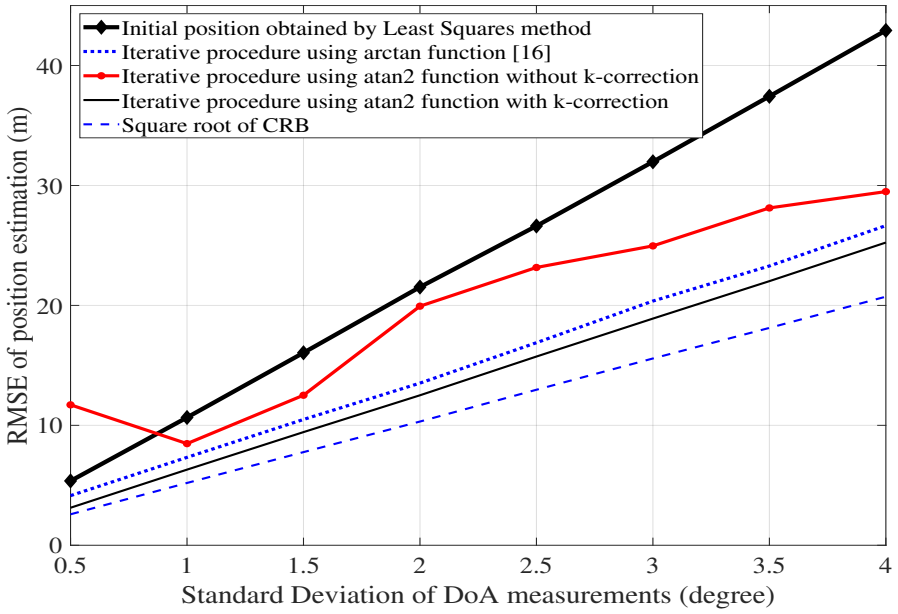



Figure 4.9: DoA-based localization at network of base stations: Comparison of RMSE when the standard deviation of DoA measurements varies from 0.5° to 4°

are installed at the height of 15m and 20m, respectively. As a result, there are totally 12 base stations in our network.

The positions of the base stations, as well as the space where the mobile device is arbitrarily placed, are illustrated with a view from top (Fig. 4.8a) and a view from one side (Fig. 4.8b). In Fig. 4.8a, the x-coordinate and y-coordinate of any random position of the mobile device are set up to be far enough from those coordinates of all the base stations, so that the true elevation angles are far from the 2 boundaries $-\pi/2$ and $\pi/2$.

The value $\varepsilon_2$ for the stopping criterion is 0.01.

In the simulations, we assume that all the estimations of azimuth and elevation angles have the same standard deviation: $\sigma_{az,1} = \sigma_{el,1} = \cdots = \sigma_{az,N} = \sigma_{el,N} = \sigma$.

**Results**

Fig. 4.9 and Fig. 4.10 illustrate the results when the common standard deviation of DoA estimations ($\sigma$) varies from 0.5° to 4°. Specifically, Fig. 4.9 compares the RMSEs of the 4 algorithms:

(a) The initial position obtained by Least Squares method shown in section 4.4.

(b) Maximum Likelihood estimator with the definition of azimuth angle using arctan function [39].

(c) Maximum Likelihood estimator **without** k-correction; the definition of azimuth angle using atan2 function without k-correction.

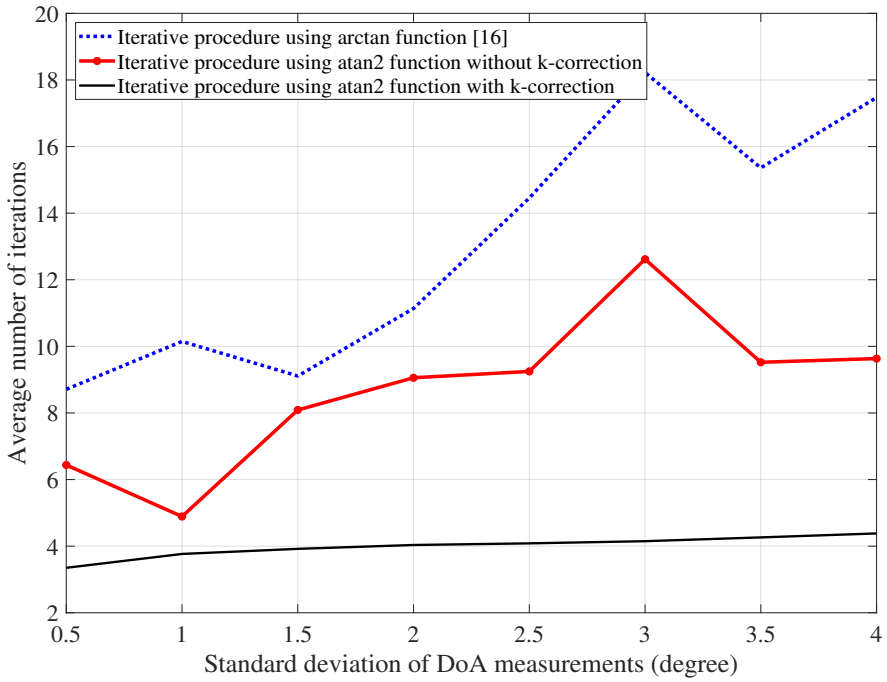



Figure 4.10: DoA-based localization at network of base stations: Comparison of average number of iterations when the standard deviation of DoA measurements varies from 0.5° to 4°

(d) Maximum Likelihood estimator **with** k-correction; the definition of azimuth angle using atan2 function (our proposed algorithm).

To validate the performances of the algorithms, we added the $\sqrt{\text{CRB}}$. Fig. 4.10 compares the average number of iterations of the three algorithms (b), (c) and (d).

Section 4.6 introduces 3 possible outcomes of an iterative procedure. Table 6.1 compares their results on RMSE and number of iterations.

In Fig. 4.9, the RMSE of our proposed algorithm (d) is much smaller than the "initial point" and higher than the $\sqrt{\text{CRB}}$, which shows that the algorithm (d) is efficient and unbiased. Compared to the algorithms (b) and (c), the algorithm (d) has the lowest RMSE so its positioning results are the most accurate. Furthermore, in Fig. 4.10, the average number of iterations of our proposed algorithm (d) is the lowest, which proves that this algorithm has the most converging procedures and the fewest combinations of diverging and oscillating procedures. As a result, it reduces remarkably the time delay for localization.

# Chapter 5

# Direction-based localization at mobile device in 2D scenarios

## 5.1 Introduction

DoA-based localization computes the coordinates of the mobile device based on the direction of incident waves to base stations. The numerical expression of this direction is the trigonometric angle between the $x$-direction and the signal wave (Fig. 5.2). To avoid confusion in measuring angles, all the angles' values must be defined in an interval whose length is $2\pi$. Furthermore, at the boundaries of the interval, the DoA is very sensitive to noise. For instance, on the condition that an angle's set of definition is $[0; 2\pi)$, when the true value of the angle is $\varepsilon_a$, a small noise can make the angle's value $-\varepsilon_b$ ($\varepsilon_a$ and $\varepsilon_b$ are very small positive value). However, as the set of definition is $[0; 2\pi)$, the estimated value of this angle is supposed to be $2\pi - \varepsilon_b$, which is very different from the true value.

DoA-based positioning is only feasible for **Network-Positioning** because the orientation of each base station is fixed and known. However, as for **Self-Positioning**, since the orientation of a mobile device is unknown, it cannot refer to the $x$-direction to calculate the angle of arrival. Consequently, Direction Difference of Arrival (DDoA) is proposed. In this technique, only the difference in directions of arrival of incident waves from a pair of base stations is required (Fig. 5.1). Mathematically, a DDoA is calculated by subtracting the 2 DoAs concerned. Recent achievements in DoA estimations at mobile devices [40–43] make the DDoA-based positioning algorithms potential and promising.

In [9], the very first ideas about DDoA are introduced. A DoA-based positioning method for mobile devices, in which the prior knowledge of the $x$-direction is not required, is also presented in [44]. Nevertheless, the authors use arctan function in the definition of DoA, which cannot cover all the possible values of an angle. In [45, 46] the authors also demonstrate DDoA in different ways of explication. In general, they do not consider the sensitivity to noise of an angle's measured value.

This chapter gives a clear analysis of localization based on the DDoAs of the incident wave from the mobile device to the network of base stations. In the definition of DoA, the atan2 function is utilized instead of arctan function. Subtraction of two DoAs returns

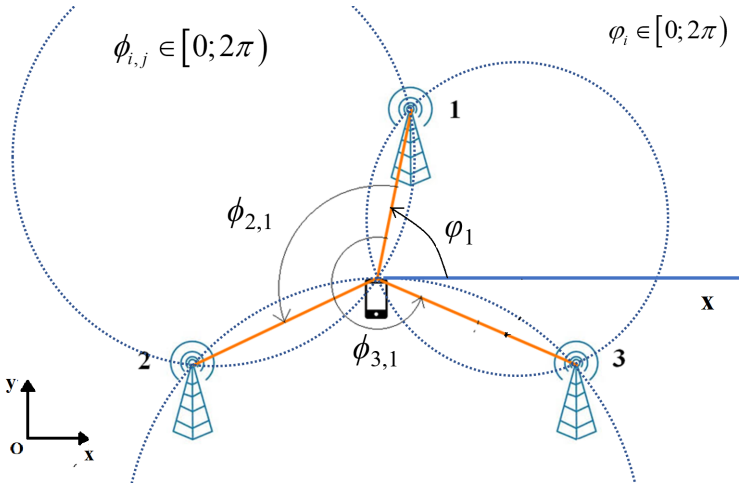



Figure 5.1: DDoA approach for mobile-based localization

a value in the range of $(-2\pi; 2\pi)$, so a modulo operation with a divisor of $2\pi$ is applied. For this reason, the codomain of DDoA computations is $[0; 2\pi)$. This interval is also the set of definition of DDoA measured values. Furthermore, an additional correction is added to the subtraction of two DoAs to avoid possible huge computing errors caused by small mistakes in practical measurements. Compared to [9], we formulate a Maximum Likelihood estimator with the fitting criterion at the level of DDoAs instead of their tangents, allowing the original introduction of phase corrections (to offset the modulo operations).

## 5.2 Problem Formulation

Let $\theta$ be the angle demonstrating the orientation of the mobile device (Fig. 5.2). We assume that $-\pi < \theta \leq \pi$.

Let $\psi_i$ be the DoA compared to the $x$-axis, $\varphi_i$ is the DoA measured at the antenna arrays of the mobile device.

We have:

$$\psi_i = \text{mod}(\text{atan2}(y_i - y, x_i - x), 2\pi) \tag{5.1}$$

$$\varphi_i = \text{mod}(\varphi_i - \theta, 2\pi) \tag{5.2}$$

At the mobile device, we can only get the estimate of $\varphi_i$, which is not directly linked to the related coordinates. We have two solutions to solve this self-positioning problem:

- Based on the estimates of $\varphi_i$, we calculate the coordinates of the mobile device with the simultaneous estimation of orientation.
- Compute the estimated Direction Difference of Arrival (DDoA) from the estimated DoA, then localize the mobile device.

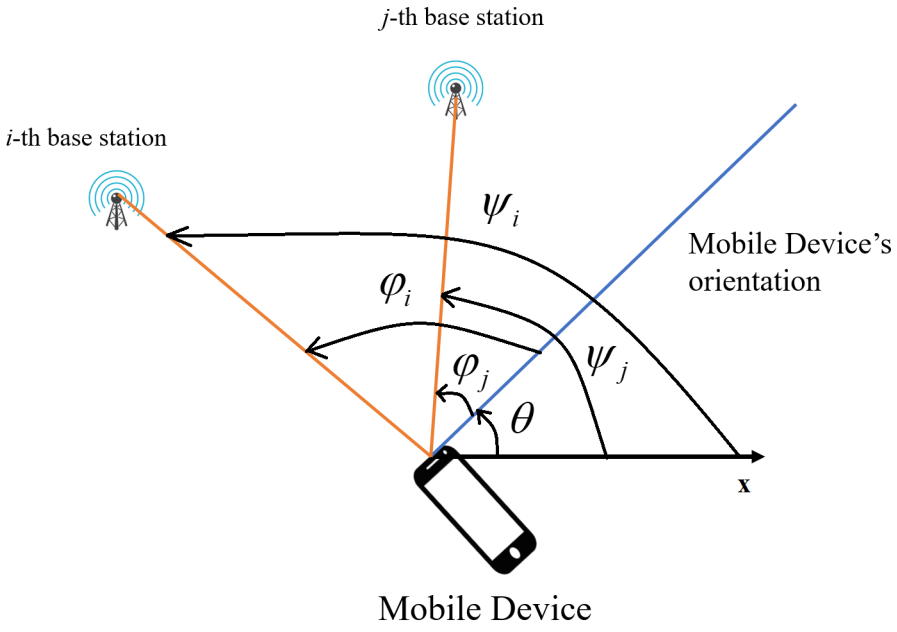



Figure 5.2: Illustration of 2D Self-Positioning

## 5.3 Localization with joint estimation of mobile orientation

DoA estimation can be expressed as:

$$\hat{\psi}_i = \psi_i + n_{\text{DoA},i} + k_{\text{DoA},i} 2\pi \tag{5.3}$$

where $n_{\text{DoA},i}$ is the error in DoA estimation. It is assumed that $n_{\text{DoA},i}$ is Gaussian distributed with zero-mean and variance of $\sigma^2_{\text{DoA}}$.

We name the action of adding the phase jump correction of $k_i 2\pi$ as **k-correction**. We then have the expression of $k_i$.

$$k_{\text{DoA},i} = \begin{cases} 1 & , \psi_i + n_{\text{DoA},i} < 0 \\ -1 & , \psi_i + n_{\text{DoA},i} \geq 2\pi \\ 0 & \text{otherwise.} \end{cases} \tag{5.4}$$

Thanks to the **k-correction**, the estimated DoA is also in the interval: $0 \leq \hat{\psi}_i < 2\pi$.

### 5.3.1 Least Squares method

We have the relation between the DoA and the coordinates:

$$\tan \psi_i = \tan(\varphi_i + \theta) = \frac{y_i - y}{x_i - x} \tag{5.5}$$

From the Appendix C, we have the following equation:

$$\tan\varphi_i(x + y\tan\theta) - (y - x\tan\theta) - (x_i + y_i\tan\varphi_i)\tan\theta = x_i\tan\varphi_i - y_i \tag{5.6}$$

In matrix formulation, we denote

$$\hat{\boldsymbol{A}}_{\text{DoA}} = \begin{bmatrix} \tan\hat{\varphi}_1 & -1 & -(x_1 + y_1\tan\hat{\varphi}_1) \\ \tan\hat{\varphi}_2 & -1 & -(x_2 + y_2\tan\hat{\varphi}_2) \\ \cdots & \cdots & \cdots \\ \tan\hat{\varphi}_N & -1 & -(x_N + y_N\tan\hat{\varphi}_N) \end{bmatrix}$$

$$\boldsymbol{\omega}_{\text{DoA}} = \begin{bmatrix} x + y\tan\theta \\ y - x\tan\theta \\ \tan\theta \end{bmatrix}; \quad \hat{\boldsymbol{b}}_{\text{DoA}} = \begin{bmatrix} x_1\tan\hat{\varphi}_1 - y_1 \\ x_2\tan\hat{\varphi}_2 - y_2 \\ \cdots \\ x_N\tan\hat{\varphi}_N - y_N \end{bmatrix}$$

We have $\hat{\boldsymbol{A}}_{\text{DoA}}\boldsymbol{\omega}_{\text{DoA}} = \hat{\boldsymbol{b}}_{\text{DoA}}$ so

$$\hat{\boldsymbol{\omega}}_{\text{DoA}} = \hat{\boldsymbol{A}}^{\dagger}_{\text{DoA}}\hat{\boldsymbol{b}}_{\text{DoA}} \tag{5.7}$$

The estimates of coordinates and orientation:

$$\hat{x} = \frac{[\hat{\boldsymbol{\omega}}_{\text{DoA}}]_1 - [\hat{\boldsymbol{\omega}}_{\text{DoA}}]_2[\hat{\boldsymbol{\omega}}_{\text{DoA}}]_3}{1 + [\hat{\boldsymbol{\omega}}_{\text{DoA}}]_3^2} \tag{5.8}$$

$$\hat{y} = \frac{[\hat{\boldsymbol{\omega}}_{\text{DoA}}]_2 + [\hat{\boldsymbol{\omega}}_{\text{DoA}}]_1[\hat{\boldsymbol{\omega}}_{\text{DoA}}]_3}{1 + [\hat{\boldsymbol{\omega}}_{\text{DoA}}]_3^2} \tag{5.9}$$

$$\tan\hat{\theta} = [\hat{\boldsymbol{\omega}}_{\text{DoA}}]_3 \tag{5.10}$$

Only the tangent of the mobile orientation is computed. We consider that the estimated mobile orientation is

$$\hat{\theta} = \arctan[\hat{\boldsymbol{\omega}}_{\text{DoA}}]_3 \tag{5.11}$$

### 5.3.2 Maximum Likelihood estimator

The estimated DoA is computed by

$$\hat{\psi}_i = \text{mod}(\hat{\varphi}_i + \hat{\theta}_i, 2\pi) \tag{5.12}$$

where $\hat{\theta}_i$ is the estimated orientation of the mobile device and computed by (5.11).

In vector form, we denote

$$\hat{\boldsymbol{\psi}} = \begin{bmatrix} \hat{\psi}_1 & \hat{\psi}_2 & \dots & \hat{\psi}_N \end{bmatrix}^{\text{T}} \tag{5.13}$$

$$\boldsymbol{f}_{\mathrm{DoA}}(\hat{\boldsymbol{x}}, \boldsymbol{k}_{\mathrm{DoA}}) = \begin{bmatrix} \hat{\psi}_1(\hat{\boldsymbol{x}}) + k_{\mathrm{DoA},1} 2\pi \\ \hat{\psi}_2(\hat{\boldsymbol{x}}) + k_{\mathrm{DoA},2} 2\pi \\ \cdots \\ \hat{\psi}_N(\hat{\boldsymbol{x}}) + k_{\mathrm{DoA},N} 2\pi \end{bmatrix} \tag{5.14}$$

where $\boldsymbol{k}_{\mathrm{DoA}} = [k_{\mathrm{DoA},1} \;\; k_{\mathrm{DoA},2} \;\; \cdots \;\; k_{\mathrm{DoA},N}]^T$, $\hat{\varphi}_i(\hat{\boldsymbol{x}})$ is the estimated DoA depending on $\hat{\boldsymbol{x}} = [\hat{x} \;\; \hat{y}]$ and computed by:

$$\hat{\psi}_i(\hat{\boldsymbol{x}}) = \mathrm{mod}(\mathrm{atan2}(y_i - \hat{y},\; x_i - \hat{x}), 2\pi) \tag{5.15}$$

Treating the phase shift vector $\boldsymbol{k}_{\mathrm{DoA}}$ as unknown parameters and ignoring their dependence on the noise, the measurement vector $\hat{\boldsymbol{\varphi}}$ is Gaussian with mean vector of $\boldsymbol{f}$ and covariance matrix $\boldsymbol{C}_{\mathrm{DoA}}$, we have the probability density function (pdf) [21]:

$$\mathrm{p}\left(\hat{\boldsymbol{\psi}} | \boldsymbol{x}, \boldsymbol{k}_{\mathrm{DoA}}\right) = \frac{(2\pi)^{-\frac{N}{2}}}{|\boldsymbol{C}_{\mathrm{DoA}}|^{\frac{1}{2}}} \exp\left[\frac{-1}{2}(\hat{\boldsymbol{\psi}} - \boldsymbol{f}_{\mathrm{DoA}})^T \boldsymbol{C}_{\mathrm{DoA}}^{-1}(\hat{\boldsymbol{\psi}} - \boldsymbol{f}_{\mathrm{DoA}})\right] \tag{5.16}$$

where $\boldsymbol{C}_{\mathrm{DoA}} = \mathrm{diag}(\sigma^2_{\mathrm{DoA},1}, \sigma^2_{\mathrm{DoA},2}, \ldots, \sigma^2_{\mathrm{DoA},N})$.

Maximizing the pdf in (5.16) is equivalent to

$$\hat{\boldsymbol{x}}, \widehat{\boldsymbol{k}} = \arg\min_{\boldsymbol{x},\boldsymbol{k}} (\hat{\boldsymbol{\psi}} - \boldsymbol{f}_{\mathrm{DoA}}(\boldsymbol{x}, \boldsymbol{k}_{\mathrm{DoA}}))^T \boldsymbol{C}_{\mathrm{DoA}}^{-1} (\hat{\boldsymbol{\psi}} - \boldsymbol{f}_{\mathrm{DoA}}(\boldsymbol{x}, \boldsymbol{k}_{\mathrm{DoA}})) \tag{5.17}$$

which we shall perform alternatingly.

The Gauss Newton algorithm [22] is applied for $\hat{\boldsymbol{x}}$. At the iteration $(u+1)$:

$$\hat{\boldsymbol{x}}^{(u+1)} = \hat{\boldsymbol{x}}^{(u)} + \left(\boldsymbol{G}_{\mathrm{DoA}}^T \boldsymbol{C}_{\mathrm{DoA}}^{-1} \boldsymbol{G}_{\mathrm{DoA}}\right)^{-1} \boldsymbol{G}_{\mathrm{DoA}}^T \boldsymbol{C}_{\mathrm{DoA}}^{-1} \left(\hat{\boldsymbol{\psi}} - \boldsymbol{f}\left(\hat{\boldsymbol{x}}^{(u)}, \boldsymbol{k}_{\mathrm{DoA}}^{(u+1)}\right)\right) \tag{5.18}$$

where $\boldsymbol{G}_{\mathrm{DoA}}$ is the Jacobian matrix.

$$\boldsymbol{G}_{\mathrm{DoA}} = \boldsymbol{G}\left(\hat{\boldsymbol{x}}^{(u)}, \boldsymbol{k}_{\mathrm{DoA}}^{(u+1)}\right), \; \boldsymbol{G}(\boldsymbol{x}, \boldsymbol{k}) = \frac{\partial \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{k})}{\partial \boldsymbol{x}^T}. \tag{5.19}$$

At this point, it is important to determine the value of $k_i$. As we do not know the additive noise in each DoA measurement, $k_i$ cannot be determined by equation (5.4). From (5.3), we have

$$|n_{\mathrm{DoA},i}| = |\hat{\psi}_i - \psi_i - k_{\mathrm{DoA},i} 2\pi| \tag{5.20}$$

We assume $n_{\mathrm{DoA},i}$ small enough, $|n_{\mathrm{DoA},i}| < \pi$. Thus $\widehat{k}_{\mathrm{DoA},i}$ is estimated by

$$\hat{k}_{\mathrm{DoA},i}^{(u+1)} = \arg \min_{k_{\mathrm{DoA},i} \in \{0;\pm 1\}} |\hat{\psi}_i\left(\hat{\boldsymbol{x}}^{(u)}\right) - \hat{\psi}_i - k_{\mathrm{DoA},i} 2\pi| \tag{5.21}$$

where $\hat{\boldsymbol{x}}^{(u)} = [\hat{x}^{(u)} \;\; \hat{y}^{(u)}]$ is the estimated coordinate vector of the mobile device at the $u$-th iteration.

In a nutshell, we propose the Algorithm 4, a Gauss-Newton iterative solution for the Maximum Likelihood estimator.

**Algorithm 4:** Proposed Maximum Likelihood estimator with phase correction $\widehat{\boldsymbol{k}}$

Take the measured DoA $\hat{\varphi}_i$ as the trigonometric angle of the incident wave from $i$-th base station to the mobile device.
Assign $u = 1$ and $\varepsilon_{\mathrm{DoA}}$ sufficiently small.
Compute the estimation $\hat{\boldsymbol{\omega}}_{\mathrm{DoA}}$ by (5.12), then assign $\hat{\boldsymbol{x}}^{(1)} = \left[[\hat{\boldsymbol{\omega}}_{\mathrm{DoA}}]_1 \quad [\hat{\boldsymbol{\omega}}_{\mathrm{DoA}}]_2\right]^T$ as the first estimated coordinates of the mobile device.
**repeat**
    Compute the estimated Direction of Arrival $\hat{\psi}_i$ by (5.3).
    **if** $|\hat{\psi}_i\left(\hat{\boldsymbol{x}}^{(u)}\right) - \hat{\psi}_i \geq \pi$ **then**
        $\hat{k}_{\mathrm{DoA},i} = \mathrm{sign}\left(\hat{\psi}_i\left(\hat{\boldsymbol{x}}^{(u)}\right) - \hat{\psi}_i\right)$
    **else**
        $\hat{k}_{\mathrm{DoA},i} = 0$ ;
    Compute $\hat{\boldsymbol{x}}^{(u+1)}$ by (5.18).
    $u = u + 1$;
**until** $\|\hat{\boldsymbol{x}}^{(u)} - \hat{\boldsymbol{x}}^{(u-1)}\|_2 < \varepsilon_{DoA}$ *or* $u > 1000$ *or* $\|\hat{\boldsymbol{x}}^{(u)}\|_2 = \pm\infty$;
**if** $u > 1000$ *or* $\|\hat{\boldsymbol{x}}^{(u)}\|_2 = \pm\infty$ **then**
    $\hat{\boldsymbol{x}}^{(1)}$ is the estimated position of the mobile device;
**else**
    $\hat{\boldsymbol{x}}^{(u)}$ is the estimated position of the mobile device;

## 5.4 Localization based on DDoA

### 5.4.1 Direction of Arrival

We define $\varphi_i$ to be the **trigonometric angle** between the $x$ axis and the signal ray received at the mobile station. Let $(x, y)$ be the coordinates of the mobile station and $(x_i, y_i)$ be the coordinates of the $i$-th base station. We then have the real DoA of the signal from the $i$-th base station:

$$\psi_i = \operatorname{atan2}(y_i - y, x_i - x)\,. \tag{5.22}$$

In the presence of estimation errors, the measured value of $i$-th DoA shall be:

$$\hat{\psi}_i = \psi_i + n_{\mathrm{DoA},i} = \operatorname{atan2}(y_i - y, x_i - x) + n_{\mathrm{DoA},i} \tag{5.23}$$

where $n_{\mathrm{DoA},i}$ is the error in DoA estimation. The authors of [47] illustrates that when there is Gaussian noise in received signal, the error in estimation is asymptotically Gaussian distributed with zero-mean. As a result, we can assume that $n_{\mathrm{DoA},i}$ is Gaussian distributed with zero-mean and variance of $\sigma^2_{\mathrm{DoA},i}$.

### 5.4.2 Direction Difference of Arrival

As $\psi_i \in [0; 2\pi)$, we have the difference $d\psi_{i,j} = \psi_i - \psi_j \in (-2\pi; 2\pi)$. However, the value of an angle must be predefined in a $2\pi$-long range. Therefore, we state the Direction Difference of Arrival (DDoA) between signal ray from $i$-th base station and signal ray from $j$-th base station (where $i$ from 1 to $N$, $j$ from 1 to $N$, $i \neq j$, $N$ is the number of base stations) as follows:

$$\begin{aligned}\phi_{i,j} &= \operatorname{mod}(\psi_i - \psi_j, 2\pi) = \operatorname{mod}(d\psi_{i,j}, 2\pi) \\ &= \operatorname{mod}(\operatorname{atan2}(y_i - y, x_i - x) - \operatorname{atan2}(y_j - y, x_j - x), 2\pi)\end{aligned} \tag{5.24}$$

where for the last equality and below we assume the absence of errors. In terms of the arctan function, we get with (2)

$$\begin{aligned}\phi_{i,j} &= \arctan\frac{y_i - y}{x_i - x} - \arctan\frac{y_j - y}{x_j - x} + m_1\pi \\ &= \arctan\frac{(y_i - y)(x_j - x) - (x_j - x)(y_j - y)}{(y_i - y)(y_j - y) + (x_i - x)(x_j - x)} + m_2\pi\end{aligned} \tag{5.25}$$

where $m_1$ and $m_2$ are integers. Hence we get

$$\tan\phi_{i,j} = \frac{(y_i - y)(x_j - x) - (x_i - x)(y_j - y)}{(y_i - y)(y_j - y) + (x_i - x)(x_j - x)}\,. \tag{5.26}$$

$$\begin{aligned}&x[-y_i + y_j + (x_i + x_j)\tan(\phi_{i,j})] + y[x_i - x_j + (y_i + y_j)\tan(\phi_{i,j})] - (x^2 + y^2)\tan(\phi_{i,j}) \\ &= -x_j y_i + x_i y_j + (x_i x_j + y_i y_j)\tan(\phi_{i,j})\,.\end{aligned} \tag{5.27}$$

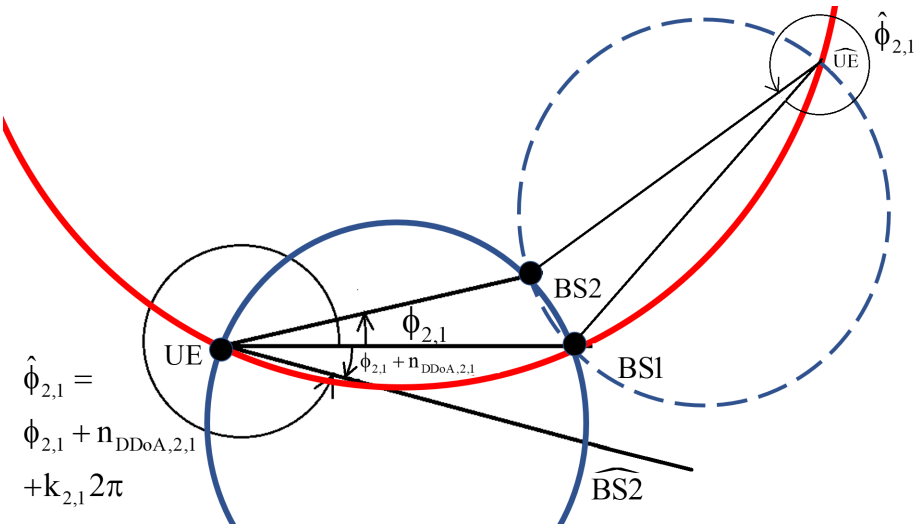



Figure 5.3: Sensitivity to noise of an angle's measured value

In the presence of errors, the DDoA measurements, which are the practical estimated values of the DDoAs, also in range of $[0;\ 2\pi)$ and denoted by $\{\hat{\phi}_{i,j}\}$, are modeled as

$$\hat{\phi}_{i,j} = \mathrm{mod}\left(\hat{\psi}_i - \hat{\psi}_j, 2\pi\right) = \phi_{i,j} + k_{i,j}2\pi + n_{\mathrm{DDoA},i,j} \tag{5.28}$$

where $n_{\mathrm{DDoA},i,j} = n_{\mathrm{DoA},i} - n_{\mathrm{DoA},j} \sim N(0, \sigma^2_{\mathrm{DoA},i} + \sigma^2_{\mathrm{DoA},j})$ and the modulo induced noise term $k_i$ is defined as:

$$k_{\mathrm{DDoA},i,j} = \begin{cases} 1 & , \begin{cases} d\varphi_{i,j} \geq 0 \text{ and } d\varphi_{i,j} + n_{\mathrm{DDoA},i,j} < 0 \\ \textbf{or } d\varphi_{i,j} + n_{\mathrm{DDoA},i,j} < -2\pi \end{cases} \\ -1 & , \begin{cases} d\varphi_{i,j} < 0 \text{ and } d\varphi_{i,j} + n_{\mathrm{DDoA},i,j} \geq 0 \\ \textbf{or } d\varphi_{i,j} + n_{\mathrm{DDoA},i,j} \geq 2\pi \end{cases} \\ 0 & \text{otherwise.} \end{cases} \tag{5.29}$$

### 5.4.3 Estimating position by Least Squares method

Regardless of $k_{\mathrm{DDoA},i,j}$, for small enough $n_{\mathrm{DDoA},i,j}$ we get

$$\tan\hat{\phi}_{i,j} = \tan(\phi_{i,j} + k_{\mathrm{DDoA},i,j}2\pi + n_{\mathrm{DDoA},i,j}) \approx \tan\phi_{i,j} + n_{\mathrm{DDoA},i,j}\,. \tag{5.30}$$

(5.27) is a 2-variable quadratic equation, which is satisfied by $(x_i, y_i)$, $(x_j, y_j)$ and $(x, y)$. In other words, (5.27) is the equation of a circle passing through the positions of $i$-th BS, $j$-th BS and the mobile. With 2 base stations and a DDoA, the locus of all the possible positions of the mobile device is a circle and the DDoA is an inscribed angle. When $\phi_{i,j}$ is close to 0, the noise term $n_{\mathrm{DDoA},i,j}$ can make a huge change to the circle. For example, in Fig. 5.3, with two base stations BS1 and BS2, the dash blue circle is

the estimate of the solid blue one in noisy scenario. Thus, the estimated position of the mobile device ($\widehat{\text{UE}}$) is very far from its true position (UE).

With $N$ BSs, we get $N(N-1)/2$ circles.

In matrix notations, we define $\boldsymbol{\omega}_{\text{DDoA}} = [x \;\; y \;\; x^2+y^2]^T$. In addition, $\hat{\boldsymbol{A}}$ and $\hat{\boldsymbol{b}}$ are defined by (5.31) and (5.32) respectively, which are illustrated in the beginning of the following page.

$$\hat{\boldsymbol{A}}_{\text{DDoA}} = \begin{bmatrix} -y_2+y_1+(x_2+x_1) & x_2-x_1-(y_2+y_1) & -\tan(\hat{\phi}_{2,1}) \\ -y_3+y_1+(x_3+x_1) & x_3-x_1-(y_3+y_1) & -\tan(\hat{\phi}_{3,1}) \\ \multicolumn{3}{c}{\dots} \\ -y_N+y_{N-1}+(x_N+x_{N-1}) & x_N-x_{N-1}-(y_N+y_{N-1}) & -\tan(\hat{\phi}_{N,N-1}) \end{bmatrix} \tag{5.31}$$

$$\hat{\boldsymbol{b}}_{\text{DDoA}} = \begin{bmatrix} -x_1y_2+x_2y_1+(x_2x_1+y_2y_1)\tan(\hat{\phi}_{2,1}) \\ -x_1y_3+x_3y_1+(x_3x_1+y_3y_1)\tan(\hat{\phi}_{3,1}) \\ \dots \\ -x_{N-1}y_N+x_Ny_{N-1}+(x_Nx_{N-1}+y_Ny_{N-1})\tan(\hat{\phi}_{N,N-1}) \end{bmatrix} \tag{5.32}$$

We have

$$\hat{\boldsymbol{A}}_{\text{DDoA}}\boldsymbol{\omega}_{\text{DDoA}} = \hat{\boldsymbol{b}}_{\text{DDoA}} \tag{5.33}$$

Therefore, the estimate of $\boldsymbol{\omega}$ is calculated by

$$\hat{\boldsymbol{\omega}}_{\text{DDoA}} = \hat{\boldsymbol{A}}^{\dagger}_{\text{DDoA}}\hat{\boldsymbol{b}}_{\text{DDoA}} \tag{5.34}$$

where $\boldsymbol{A}^{\dagger} = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T$ is the Moore-Penrose pseudo inverse of matrix $\boldsymbol{A}$.

The estimated coordinates of the mobile device are the two first elements of $\hat{\boldsymbol{\omega}}$:

$$\hat{\boldsymbol{x}} = \begin{bmatrix}[\hat{\boldsymbol{\omega}}_{\text{DDoA}}]_1 & [\hat{\boldsymbol{\omega}}_{\text{DDoA}}]_2\end{bmatrix}^T \tag{5.35}$$

To further optimize the estimation of the mobile's position, this can be taken as initialization of an iterative procedure discussed next.

### 5.4.4 Iterative Maximum Likelihood Procedure

To optimize $\hat{\boldsymbol{x}}$ obtained in (5.35), an iterative Maximum Likelihood estimator is applied.

In vector form, we denote

$$\hat{\boldsymbol{\phi}} = \begin{bmatrix}\hat{\phi}_{2,1} & \hat{\phi}_{3,1} & \dots \hat{\phi}_{N,1}\end{bmatrix}^T \tag{5.36}$$

$$\boldsymbol{f}_{\text{DDoA}}(\boldsymbol{x}, \boldsymbol{k}_{\text{DDoA}}) = \begin{bmatrix} \phi_{2,1}(\boldsymbol{x}) + k_{\text{DDoA},2,1}2\pi \\ \phi_{3,1}(\boldsymbol{x}) + k_{\text{DDoA},3,1}2\pi \\ \dots \\ \phi_{N,1}(\boldsymbol{x}) + k_{\text{DDoA},N,1}2\pi \end{bmatrix} \tag{5.37}$$

where $\boldsymbol{k}_{\text{DDoA}} = [k_{\text{DDoA},2,1}\ k_{\text{DDoA},3,1} \cdots k_{\text{DDoA},N,1}]^T$, $\boldsymbol{x} = [x \quad y]^T$ and $\phi_{i,1}(\boldsymbol{x})$ is the estimated DDoA between the 1st and the $i$-th incident waves ($i \geq 2$) and computed by

$$\phi_{i,j}(\boldsymbol{x}) = \text{mod}\left(\text{atan2}(y_i - y, x_i - x) - \text{atan2}(y_1 - y, x_1 - x), 2\pi\right) \tag{5.38}$$

We denote the vector of error estimation as $\boldsymbol{n}_{\text{DDoA}}$

$$\boldsymbol{n}_{\text{DDoA}} = \begin{bmatrix} n_{\text{DDoA},2,1} & n_{\text{DDoA},3,1} & \cdots & n_{\text{DDoA},N,1} \end{bmatrix} \tag{5.39}$$

The covariance matrix of all the additive errors is

$$\boldsymbol{C}_{\text{DDoA}} = \text{E}(\boldsymbol{n}_{\text{DDoA}} \boldsymbol{n}_{\text{DDoA}}^T) = \sigma_{\text{DoA},1}^2 \bullet \mathbf{1} \bullet \mathbf{1}^T + \text{diag}(\sigma_{\text{DoA},2}^2, \sigma_{\text{DoA},3}^2, \ldots, \sigma_{\text{DoA},N}^2) \tag{5.40}$$

where $\mathbf{1} = \begin{bmatrix} 1 & 1 & \ldots & 1 \end{bmatrix}^T$ is the all-one vector.

Treating the phase shift vector $\boldsymbol{k}$ as unknown parameters and ignoring their dependence on the noise, the measurement vector $\hat{\boldsymbol{\phi}}$ is Gaussian with mean vector of $\boldsymbol{f}$ and covariance matrix $\boldsymbol{C}_{\text{DDoA}}$, we have the probability density function (pdf) [21]:

$$\text{p}\left(\hat{\boldsymbol{\phi}} | \boldsymbol{x}, \boldsymbol{k}_{\text{DDoA}}\right) = \frac{(2\pi)^{-\frac{N}{2}}}{|\boldsymbol{C}_{\text{DooA}}|^{\frac{1}{2}}} \exp\left[\frac{-1}{2}(\hat{\boldsymbol{\phi}} - \boldsymbol{f}_{\text{DDoA}})^T \boldsymbol{C}_{\text{DDoA}}^{-1} (\hat{\boldsymbol{\phi}} - \boldsymbol{f}_{\text{DDoA}})\right] \tag{5.41}$$

Maximizing the pdf in (5.41) is equivalent to finding

$$\hat{\boldsymbol{x}}, \widehat{\boldsymbol{k}}_{\text{DDoA}} = \arg \min_{\boldsymbol{x}, \boldsymbol{k}_{\text{DDoA}}} (\hat{\boldsymbol{\phi}} - \boldsymbol{f}_{\text{DDoA}}(\boldsymbol{x}, \boldsymbol{k}_{\text{DDoA}}))^T \boldsymbol{C}_{\text{DDoA}}^{-1} (\hat{\boldsymbol{\phi}} - \boldsymbol{f}_{\text{DDoA}}(\boldsymbol{x}, \boldsymbol{k}_{\text{DDoA}})) \tag{5.42}$$

which we shall perform alternatingly. We consider Gauss Newton [22] for $\boldsymbol{x}$. At iteration $(u+1)$

$$\hat{\boldsymbol{x}}^{(u+1)} = \hat{\boldsymbol{x}}^{(u)} + \left(\boldsymbol{G}_{\text{DDoA}}^T \boldsymbol{C}_{\text{DDoA}}^{-1} \boldsymbol{G}_{\text{DDoA}}\right)^{-1} \boldsymbol{G}_{\text{DDoA}}^T \boldsymbol{C}_{\text{DDoA}}^{-1} \left(\hat{\boldsymbol{\phi}} - \boldsymbol{f}_{\text{DDoA}}\left(\hat{\boldsymbol{x}}^{(u)}, \boldsymbol{k}_{\text{DDoA}}^{(u+1)}\right)\right) \tag{5.43}$$

where $\boldsymbol{G}_{\text{DDoA}}$ is the Jacobian matrix of $\boldsymbol{f}_{\text{DDoA}}(\boldsymbol{x})$

$$\boldsymbol{G}_{\text{DDoA}} = \boldsymbol{G}\left(\hat{\boldsymbol{x}}^{(u)}, \boldsymbol{k}_{\text{DDoA}}^{(u+1)}\right),\ \boldsymbol{G}(\boldsymbol{x}, \boldsymbol{k}) = \frac{\partial \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{k})}{\partial \boldsymbol{x}^T}. \tag{5.44}$$

At this point, it is important to determine the value of $\boldsymbol{k}_{\text{DDoA}}$. As we do not know the additive noise in each DoA measurement, $k_{\text{DDoA},i,1}$ cannot be determined by equation (5.29). From (5.28), we have

$$|n_{\text{DDoA},i,1}| = |\hat{\phi}_{i,1}(\hat{\boldsymbol{x}}) - \phi_{i,1} - k_{\text{DDoA},i,1} 2\pi| \tag{5.45}$$

We assume that $n_{\text{DDoA},i,1}$ is small enough, so $|n_{\text{DDoA},i,1}| < \pi$ with the probability almost 1. Thus $\hat{k}_{i,1}$ can be estimated by

$$\hat{k}_{i,1}^{(u+1)} = \arg \min_{k_{i,1} \in \{0; \pm 1\}} \left| \hat{\phi}_{i,1} - \phi_{i,1}\left(\hat{\boldsymbol{x}}^{(u)}\right) - k_{i,1} 2\pi \right| \tag{5.46}$$

where $\hat{\phi}_{i,1}\left(\hat{\boldsymbol{x}}^{(u)}\right)$ is the estimated value of $\phi_{i,1}$ at the $u$-th iteration.

A procedure is expected to terminate when $\|\hat{\boldsymbol{x}}^{(u)} - \hat{\boldsymbol{x}}^{(u-1)}\|_2 < \varepsilon_{\text{DDoA}}$, for the stopping value $\varepsilon_{\text{DDoA}}$ sufficiently small. Then, the final position of the procedure is considered to be the coordinates of the mobile device in the $xy$ plane.

However, the iterative procedures do not always converge. The possible outcomes of an iterative procedure and how the estimated position of the mobile device is taken from that procedure are carefully analyzed in section 3.3.4.

In a nutshell, we propose the Algorithm 5, a Gauss-Newton iterative solution for the Maximum Likelihood estimator.

**Algorithm 5:** Proposed Maximum Likelihood estimator with phase correction $\widehat{\boldsymbol{k}}$

Take the measured DDoA $\hat{\phi}_{i,j}$ as the trigonometric angle of the incident wave from $i$-th base station and the incident wave from $j$-th base station.
Assign $u = 1$ and $\varepsilon_{\text{DDoA}}$ sufficiently small.
Compute the estimation $\hat{\boldsymbol{\omega}}_{\text{DDoA}}$ by (5.34), then assign $\hat{\boldsymbol{x}}^{(1)} = \begin{bmatrix}[\hat{\boldsymbol{\omega}}_{\text{DDoA}}]_1 & [\hat{\boldsymbol{\omega}}_{\text{DDoA}}]_2\end{bmatrix}^T$ as the first estimated coordinates of the mobile device.
**repeat**
    Compute the estimated Direction Difference of Arrival $\hat{\phi}_{i,1}$ by (5.38).
    **if** $|\hat{\phi}_{i,1}(\hat{\boldsymbol{x}}^{(u)}) - \hat{\phi}_{i,1} \geq \pi$ **then**
        $\hat{k}_{\text{DDoA},i,1} = \text{sign}(\hat{\phi}_{i,1}(\hat{\boldsymbol{x}}^{(u)}) - \hat{\phi}_{i,1})$
    **else**
        $\hat{k}_{\text{DDoA},i,1} = 0$ ;
    Compute $\hat{\boldsymbol{x}}^{(u+1)}$ by (5.43).
    $u = u + 1$;
**until** $\|\hat{\boldsymbol{x}}^{(u)} - \hat{\boldsymbol{x}}^{(u-1)}\|_2 < \varepsilon_{DDoA}$ *or* $u > 1000$ *or* $\|\hat{\boldsymbol{x}}^{(u)}\|_2 = \pm\infty$;
**if** $u > 1000$ *or* $\|\hat{\boldsymbol{x}}^{(u)}\|_2 = \pm\infty$ **then**
    $\hat{\boldsymbol{x}}^{(1)}$ is the estimated position of the mobile device;
**else**
    $\hat{\boldsymbol{x}}^{(u)}$ is the estimated position of the mobile device;

## 5.5 Simulation Results

To compare the quality among of algorithms and CRB, we use Root Mean Square Position Error (RMSE) which is defined by

$$\text{RMSE} = \sqrt{\text{E}(\|\hat{\boldsymbol{x}} - \bar{\boldsymbol{x}}\|^2)} \tag{5.47}$$

where $\bar{\boldsymbol{x}}$ is the true position of the mobile device and $\hat{X}$ is its estimate. In the $xy$ plane, RMSE averaging is over 1000 mobile positions picked randomly in a square of 1000m x 1000m centered in the circle of BSs. The network of 8 Base stations (numbered from

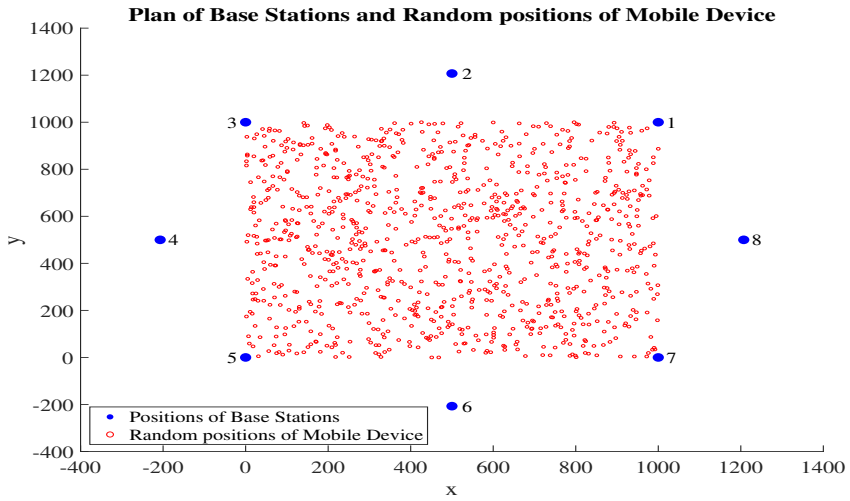



Figure 5.4: Map of base stations' network and random positions of mobile device in 2D

1 to 8) forms the circumscribed circle of this square (Fig. 5.4). The value for $\varepsilon$ in the stopping criterion of the ML estimator is 0.01.

In the figures, "initial point" refers to the position found by Least Squares method in section 5.3.1, whereas "iterative procedure" refers to the pseudo Maximum Likelihood algorithm.

Instead of comparing the MSEs to the CRB, we compare their square roots: The Root Mean Square Error (RMSE) = $\sqrt{\text{MSE}}$ and square root of CRB ($\sqrt{\text{CRB}}$). In the simulations, all the DoA estimations are assumed to have the same standard deviation: $\sigma_{\text{DoA},1} = \sigma_{\text{DoA},2} = \cdots = \sigma_{\text{DoA},N} = \sigma_{\text{DoA}}$.

### 5.5.1 DDoA-based localization

Fig. 5.5 illustrates the results when the common standard deviation of DoA estimations ($\sigma_{\text{DoA}}$) varies from 0.5° to 4°. More comprehensively, Fig. 5.5a compares the RMSEs of the 4 algorithms:

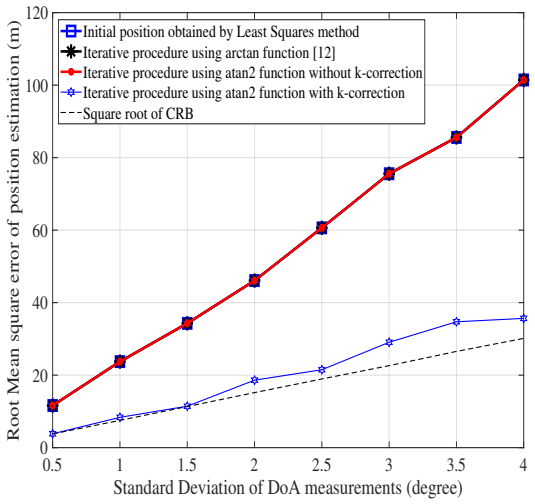
(a) The initial point obtained by Least Squares method shown in section 5.3.1.

(b) Gauss-Newton iterative procedures using arctan function [9], by replacing atan2 by arctan in (5.38) (this corresponds to (4.7), (4.12) and (4.20) in [9]).

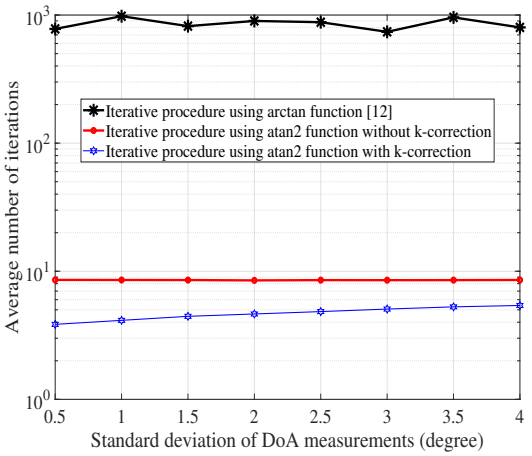
(c) Gauss-Newton iterative procedures using atan2 function **without** k-correction.

(d) Gauss-Newton iterative procedures using atan2 function **with** k-correction.

The $\sqrt{\text{CRB}}$ is also added to validate their performances. Fig. 5.5b compares the average number of iterations of the three algorithms (b), (c) and (d).




(a) RMSEs and $\sqrt{\text{CRB}}$




(b) Average number of iterations

Figure 5.5: Comparisons among the algorithms when $N = 8$ BSs, standard deviation of DOA measurements varies from $0.5^0$ to $4^0$

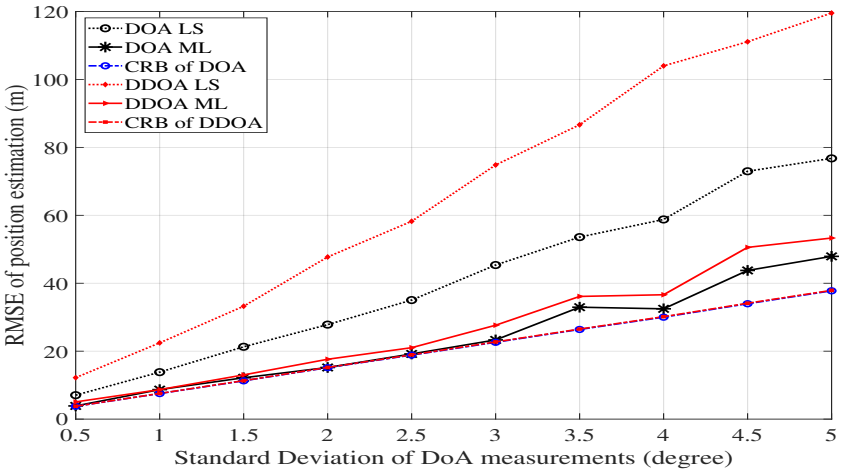
In Fig. 5.5a, there is no difference among the RMSEs of (a), (b) and (c). Undoubtedly, all the iterative procedures with algorithm (b) and (c) diverge or oscillate so that the ML estimator has no effect. On the otherhand, the RMSE of (d) is much smaller than the common RMSE of (a), (b) and (c), which means ML estimator plays an important role in position optimization. The RMSE of (d) is still higher than the $\sqrt{\text{CRB}}$, which demonstrates that this estimator is unbiased.

In Fig. 5.5b, the average number of iterations of (b) is about 900, which means that 90% of the procedures oscillate and only 10% of them diverge. As for (c), this average number is smaller than 10, because most of the procedures diverge. Our proposed algorithm (d) has the smallest average number of iterations, so it can give a remarkable reduction of the time delay for localization processes.

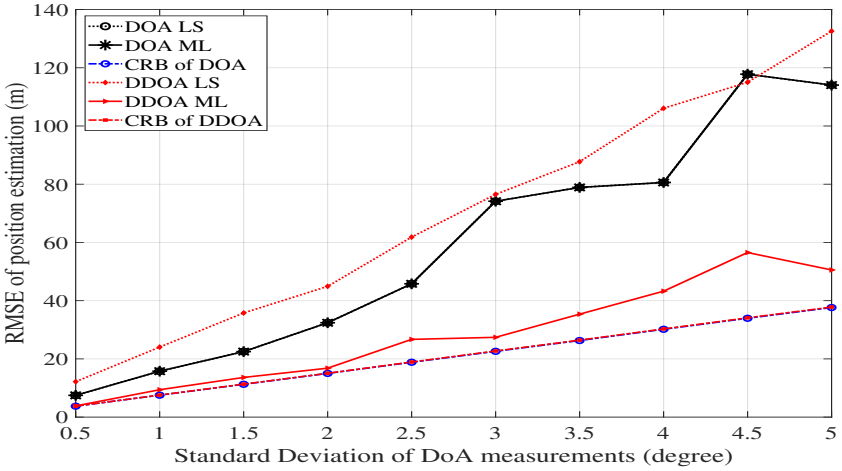
### 5.5.2 Localization with orientation estimation

We compare DDoA-based approach to the DoA-based approach in different mobile orientations. Fig. 5.6 demonstrates the RMSEs of Least Squared method and Maximum Likelihood estimator of the two approaches. It is clear that

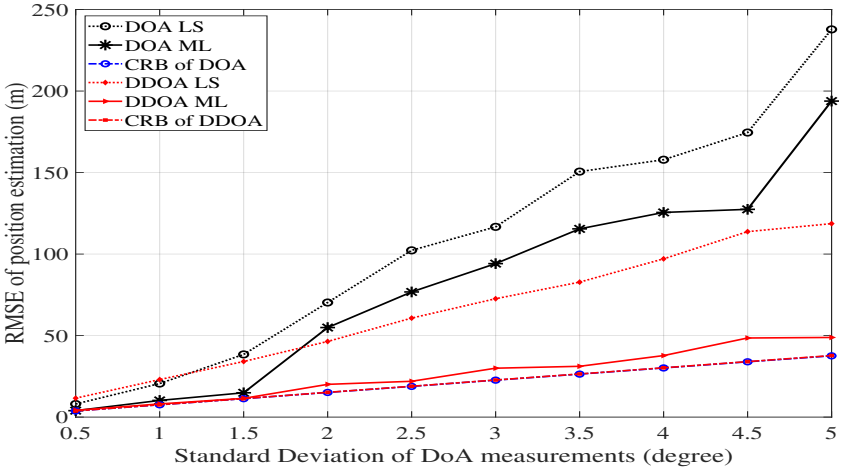
- When the absolute value of the orientation is small (like $\pi/18$ or $10°$), the DoA approach gives a more accurate position location than the DDoA approach, because the error of orientation estimation is not considerable.
- When the absolute value of the orientation is large (like $-4\pi/9$ or $-80°$), the DDoA-based localization is more precise, because there is significant error in orientation estimation, which noticeably affects the position estimation of DoA approach.




(a) Mobile orientation $\theta = 10°$




(b) Mobile orientation $\theta = 120°$




(c) Mobile orientation $\theta = -80°$

Figure 5.6: Self-positioning with joint orientation estimation

- When the mobile orientation is not in the interval $[-\pi/2; \pi/2]$, the orientation estimation is not correct so the ML estimator of the DDoA approach cannot have a correct iterative equation. As a result, its iterations do not converge at all. The DDoA-based RMSE is thus much lower than the DoA-based RMSE.

## 5.6 Conclusions

This chapter thoroughly analyzes self-positioning in 2D scenarios with 2 approaches: DDoA-based localization and DoA-based localization with joint mobile orientation estimation.

The DDoA-based positioning algorithms using atan2 function and the k-correction to overcome the troubles caused by noises in angle measurements. The simulations demonstrate the superior properties of our proposed algorithm: unbiasedness (in certain conditions) with the most accurate results and the shortest time delay.

# Chapter 6

# Direction-based localization at mobile device in 3D scenarios

## 6.1 Introduction

### 6.1.1 Related works

Several papers illustrates their researches and results. In [45], the authors showed an position algorithm using the DDoAs by a gradient iterative procedure. However, they did not show how to get the initial point for the procedure, as well as what to do if the procedure does not converge. The authors in [48] works about DDoA but the tilt of receiver is already given. In [8], a DDoA positioning algorithm is studied. The sensor determines its position by the Visible Light Communications (VLC) emitted from the light-emitting diodes (LEDs) around. Nevertheless, the authors assume that all the LEDs are collinear and the $z$-coordinate of the sensor is always lowers than the common $z$-coordinate. These assumptions reduce the complexity for the problem, but also lose the generality for the solution. In addition, the authors of [9] suggest using the difference between the azimuths and the difference between the elevations to localize the mobile device. However, these differences are not constants when the orientation of the device changes.

### 6.1.2 Our constributions

In this chapter, we form an algorithm using the DDoAs for position estimation by Least Squares method. This algorithm does not require any additional assumption. Afterwards, a Maximum Likelihood estimator is obtained by optimizing a cost function.

## 6.2 Problem Formulation of localization at Mobile device

Direction-based localization at the mobile device is much more complicated than one at the base stations.

The mathematical problems are still challenging to solve.

Fig. 6.1 shows the DoA expression in the true Cartesian coordinate system. The true azimuth angle and elevation angle of the incident wave from the $i$-th base station are $\psi_i$ and $\vartheta_i$, respectively. However, at the relative coordinate system with regard to the mobile device, the relative azimuth angle is $\varphi_i$ and and elevation angle is $\theta_i$ (Fig. 6.2). As the tilt of the mobile device is undefined, it is likely impossible to compute $(\psi_i, \vartheta_i)$ from $(\varphi_i, \theta_i)$. It is essential to find a solution which can estimate the mobile position by $(\varphi_i, \theta_i)$.

Let $(x, y, z)$ be the coordinates of the mobile device and $(x_i, y_i, z_i)$ be the coordinates of the $i$-th base station in the true Cartesian coordinate system. In the rotated relative coordinate system with regard to the mobile device, the coordinates of the mobile device are $(x^r, y^r, z^r)$; meanwhile the coordinates of the $i$-th base station are $(x_i^r, y_i^r, z_i^r)$. We then have the azimuth of DoA of the signal from the $i$-th base station:

$$\varphi_i = \operatorname{atan2}\left(y_i^r - y^r, x_i^r - x^r\right) \tag{6.1}$$

The elevation angle:

$$\theta_i = \arctan \frac{z_i^r - z^r}{\sqrt{\left(x_i^r - x^r\right)^2 + \left(y_i^r - y^r\right)^2}} \tag{6.2}$$

In practical measurements, the measured value of $i$-th azimuth and elevation are:

$$\hat{\varphi}_i = \varphi_i + \tilde{\varphi}_i + k_i 2\pi \tag{6.3}$$

$$\hat{\theta}_i = \theta_i + \tilde{\theta}_i \tag{6.4}$$

where $\tilde{\varphi}_i$ is assumed to be an additive Gaussian noise with zero-mean and variance $\nu^2$ and $\tilde{\theta}_i$ is assumed to be an additive Gaussian noise with zero-mean and variance $\mu^2$. The authors of [49] prove that these noises are zero-mean and Gaussian distributed.

We name the action of adding a correction of $k_i 2\pi$ as **k-correction**. We have the definition of $k_i$

$$k_i = \begin{cases} 1 & , \varphi_i + \tilde{\varphi}_i < -\pi \\ -1 & , \varphi_i + \tilde{\varphi}_i \geq \pi \\ 0 & \text{otherwise.} \end{cases} \tag{6.5}$$

## 6.3 Mobile-based localization by DoA-based algorithm when the orientation is given

### 6.3.1 Analysis of mobile orientation

A common way to express the orientation of a rigid object like a mobile device in 3D is quaternion [50].

Quaternion provides a convenient mathematical notation for representing spatial orientations and rotations of elements in three dimensional space. Specifically, they encode information about an axis-angle rotation about an arbitrary axis.

In 3-dimensional space, according to Euler's rotation theorem, any rotation or sequence of rotations of a rigid body or coordinate system about a fixed point is equivalent to a single rotation by a given angle $\xi$ about a fixed axis (called the Euler axis) that runs through the fixed point. The Euler axis is typically represented by a unit vector $\overrightarrow{v}$ in the picture). Therefore, any rotation in three dimensions can be represented as a combination of a vector $\overrightarrow{v}$ and a scalar $\xi$ (Fig. 6.3).

A rotation of angle $\xi$ around the axis defined by the unit vector. In addition, $\xi$ is supposed to be in $[0; 2\pi)$.

As $\overrightarrow{v} = (v_x, v_y, v_z)$ is a unit vector, $v_x^2 + v_y^2 + v_z^2 = 1$.

The quaternion $\boldsymbol{q} = [q_0 \;\; q_1 \;\; q_2 \;\; q_3]$ is defined, where

$$q_0 = \cos\frac{\xi}{2},\; q_1 = v_x \sin\frac{\xi}{2},\; q_2 = v_y \sin\frac{\xi}{2},\; q_3 = v_z \sin\frac{\xi}{2} \tag{6.6}$$

Then, the rotation matrix described by the quaternion is given as below

$$\boldsymbol{R} = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1 q_2 - q_0 q_3) & 2(q_1 q_3 + q_0 q_2) \\ 2(q_1 q_2 + q_0 q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2 q_3 - q_0 q_1) \\ 2(q_1 q_3 - q_0 q_2) & 2(q_2 q_3 + q_0 q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \tag{6.7}$$

If a position has the coordinates $\boldsymbol{p} = [p_x \;\; p_y \;\; p_z]^T$ in the true coordinate system, and the coordinates in relative rotation coordinate system is $\boldsymbol{p}^r = [p_x^r \;\; p_y^r \;\; p_z^r]^T$, their relation will be:

$$\boldsymbol{p}^r = \boldsymbol{R}\boldsymbol{p} \tag{6.8}$$

$$\boldsymbol{p} = \boldsymbol{R}^{-1}\boldsymbol{p}^r \tag{6.9}$$

### 6.3.2 Estimating position by Least Squares method

We have the relation between the coordinates and the meausured DoA in rotated coordinate system

$$\tan\varphi_i = \frac{y_i^r - y^r}{x_i^r - x^r} \tag{6.10}$$

$$x^r \sin\varphi_i - y^r \cos\varphi_i = x_i^r \sin\varphi_i - y_i^r \cos\varphi_i \tag{6.11}$$

As $\tilde{\varphi}_i$ is very small, we approximate that $\sin\tilde{\varphi}_i \approx 0$ and $\cos\tilde{\varphi}_i \approx 1$. Thus

$$\sin\varphi_i = \sin(\hat{\varphi}_i - \tilde{\varphi}_i - k_i 2\pi) = \sin(\hat{\varphi}_i - \tilde{\varphi}_i) \approx \sin\hat{\varphi}_i \tag{6.12}$$

$$\cos\varphi_i = \cos(\hat{\varphi}_i - \tilde{\varphi}_i - k_i 2\pi) = \cos(\hat{\varphi}_i - \tilde{\varphi}_i) \approx \cos\hat{\varphi}_i \tag{6.13}$$

Hence, from (6.11), it is approximated that

$$x^r \sin\hat{\varphi}_i - y^r \cos\hat{\varphi}_i = x_i^r \sin\hat{\varphi}_i - y_i^r \cos\hat{\varphi}_i \tag{6.14}$$

Moreover, we have

$$\tan\theta_i = \frac{z_i^r - z^r}{\sqrt{(x_i^r - x^r)^2 + (y_i^r - y^r)^2}} = \frac{(z_i^r - z^r)\cos\varphi_i}{x_i^r - x^r} \tag{6.15}$$

As $\tilde{\theta}_i$ is very small, we approximate that $\sin\tilde{\theta}_i \approx 0$ and $\cos\tilde{\varphi}_i \approx 1$, so $\tan\tilde{\theta}_i \approx 0$. Thus $\tan\theta_i \approx \tan\hat{\theta}_i$

We have the approximation

$$\tan\hat{\theta}_i = \frac{(z_i^r - z^r)\cos\hat{\varphi}_i}{x_i^r - x^r} \tag{6.16}$$

$$x^r \tan\hat{\theta}_i - z^r \cos\hat{\varphi}_i = x_i^r \tan\hat{\theta}_i - z_i^r \cos\hat{\varphi}_i \tag{6.17}$$

In matrix approach

$$\hat{\boldsymbol{A}}_1 = \begin{bmatrix} \sin\hat{\varphi}_1 & -\cos\hat{\varphi}_1 & 0 \\ \sin\hat{\varphi}_2 & -\cos\hat{\varphi}_2 & 0 \\ \cdots & \cdots & \cdots \\ \sin\hat{\varphi}_N & -\cos\hat{\varphi}_N & 0 \\ \tan\hat{\theta}_1 & 0 & -\cos\hat{\varphi}_1 \\ \tan\hat{\theta}_2 & 0 & -\cos\hat{\varphi}_2 \\ \cdots & \cdots & \cdots \\ \tan\hat{\theta}_N & 0 & -\cos\hat{\varphi}_N \end{bmatrix}; \ \hat{\boldsymbol{b}}_1 = \begin{bmatrix} x_1^r \sin\hat{\varphi}_1 - y_1^r \cos\hat{\varphi}_1 \\ x_2^r \sin\hat{\varphi}_2 - y_2^r \cos\hat{\varphi}_2 \\ \cdots \\ x_N^r \sin\hat{\varphi}_N - y_N^r \cos\hat{\varphi}_N \\ x_1^r \tan\hat{\theta}_1 - z_1^r \cos\hat{\varphi}_1 \\ x_2^r \tan\hat{\theta}_2 - z_2^r \cos\hat{\varphi}_2 \\ \cdots \\ x_N^r \tan\hat{\theta}_1 - z_N^r \cos\hat{\varphi}_1 \end{bmatrix}$$

$$\boldsymbol{x}^r = \begin{bmatrix} x^r & y^r & z^r \end{bmatrix}^T$$

We have $\hat{\boldsymbol{A}}_1 \boldsymbol{x}^r = \hat{\boldsymbol{b}}_1$ so the estimate of $\boldsymbol{x}^r$ is

$$\widehat{\boldsymbol{x}^r} = \min_{\boldsymbol{x}^r} \|\hat{\boldsymbol{A}}_1 \boldsymbol{x}^r - \hat{\boldsymbol{b}}_1\|^2 \tag{6.18}$$

leading to the estimate of $\boldsymbol{x}^r$ being calculated by Least-Squares estimation of $\boldsymbol{x}^r$

$$\widehat{\boldsymbol{x}^r} = \boldsymbol{A}_1^\dagger \hat{\boldsymbol{b}}_1 \tag{6.19}$$

where $\boldsymbol{A}^\dagger = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T$ is the Moore-Penrose pseudo inverse of matrix $\boldsymbol{A}$.

### 6.3.3 Optimizing position estimation by the true Maximum Likelihood estimator

To optimize $\widehat{\boldsymbol{x}^r}$ obtained in (6.55), an iterative Maximum Likelihood estimator is applied.

In vector form, we denote

$$\hat{\boldsymbol{\varphi}} = \begin{bmatrix} \hat{\varphi}_1 & \dots & \hat{\varphi}_N & \hat{\theta}_1 & \dots & \hat{\theta}_N \end{bmatrix}^T \tag{6.20}$$

$$\boldsymbol{f}(\boldsymbol{x}^r, \boldsymbol{k}) = \begin{bmatrix} \varphi_1(\boldsymbol{x}^r) + k_1 2\pi \\ \varphi_2(\boldsymbol{x}^r) + k_2 2\pi \\ \dots \\ \varphi_N(\boldsymbol{x}^r) + k_N 2\pi \\ \theta_1(\boldsymbol{x}^r) \\ \theta_2(\boldsymbol{x}^r) \\ \dots \\ \theta_N(\boldsymbol{x}^r) \end{bmatrix} \tag{6.21}$$

where $\boldsymbol{k} = [k_1 \ k_2 \ \cdots \ k_N]^T$; $\varphi_i(\boldsymbol{x}^r)$ and $\theta_i(\boldsymbol{x}^r)$ are the estimated azimuth and elevation angles, respectively, depending on $\boldsymbol{x}^r = [x^r \ \ y^r \ \ z^r]^T$ and are computed by

$$\varphi_i(\boldsymbol{x}^r) = \operatorname{atan2}\left(x_i^r - x^r,\ y_i^r - y^r\right) \tag{6.22}$$

$$\theta_i(\boldsymbol{x}^r) = \arctan \frac{z_i^r - z^r}{\sqrt{(x^r - x_i^r)^2 + (y^r - y_i^r)^2}} \tag{6.23}$$

Treating the phase shift vector $\boldsymbol{k}$ as unknown parameters and ignoring their dependence on the noise, the measurement vector $\hat{\boldsymbol{\varphi}}$ is Gaussian with mean vector of $\boldsymbol{f}$ and covariance matrix $\boldsymbol{C}$, we have the probability density function (pdf) [21]:

$$\mathrm{p}(\hat{\boldsymbol{\varphi}}|\boldsymbol{x}^r, \boldsymbol{k}) = \frac{(2\pi)^{-N}}{|\boldsymbol{C}|^{1/2}} \exp\left[\frac{-1}{2}(\hat{\boldsymbol{\varphi}} - \boldsymbol{f})^T \boldsymbol{C}^{-1} (\hat{\boldsymbol{\varphi}} - \boldsymbol{f})\right] \tag{6.24}$$

Maximizing the pdf in (6.24) is equivalent to

$$\widehat{\boldsymbol{x}^r}, \widehat{\boldsymbol{k}} = \arg\min_{\boldsymbol{x},\boldsymbol{k}} (\hat{\boldsymbol{\varphi}} - \boldsymbol{f}(\boldsymbol{x}^r, \boldsymbol{k}))^T \boldsymbol{C}^{-1} (\hat{\boldsymbol{\varphi}} - \boldsymbol{f}(\boldsymbol{x}^r, \boldsymbol{k})) \tag{6.25}$$

which we shall perform alternatingly. We consider Gauss Newton [22] for $\widehat{x^r}$. At the iteration $(u+1)$:

$$\widehat{\boldsymbol{x}^r}^{(u+1)} = \widehat{\boldsymbol{x}^r}^{(u)} + \left(\boldsymbol{G}_1^T \boldsymbol{C}^{-1} \boldsymbol{G}_1\right)^{-1} \boldsymbol{G}_1^T \boldsymbol{C}^{-1} \left(\hat{\boldsymbol{\varphi}} - \boldsymbol{f}\left(\widehat{\boldsymbol{x}^r}^{(u)}, \boldsymbol{k}^{(u+1)}\right)\right) \tag{6.26}$$

where $\boldsymbol{G}_1$ is the Jacobian matrix.

$$\boldsymbol{G}_1 = \boldsymbol{G}\left(\widehat{\boldsymbol{x}^r}^{(u)}, \boldsymbol{k}^{(u+1)}\right),\ \boldsymbol{G}(\boldsymbol{x}, \boldsymbol{k}) = \frac{\partial \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{k})}{\partial \boldsymbol{x}^T}. \tag{6.27}$$

At this point, it is important to determine the value of $k_i$. As we do not know the additive noise in each DoA measurement, $k_i$ cannot be determined by equation (6.5). From (6.3), we have

$$|\tilde{\varphi}_i| = |\hat{\varphi}_i - \varphi_i(\boldsymbol{x}^r) - k_i 2\pi| \tag{6.28}$$

We assume $\tilde{\varphi}_i$ small enough, so $|\tilde{\varphi}_i| < \pi$ with the probability almost 1. Thus $\widehat{k}_i$ can be estimated by

$$\hat{k}_i^{(u+1)} = \arg \min_{k_i \in \{0;\pm 1\}} \left| \hat{\varphi}_i - \varphi_i \left( \widehat{\boldsymbol{x}^r}^{(u)} \right) - k_i 2\pi \right| \tag{6.29}$$

where $\hat{\boldsymbol{x}}^{(u)}$ is the estimated coordinate vector of the mobile device at the $u$-th iteration.

The procedure is expected to terminate when $\|\hat{\boldsymbol{x}}^{(u)} - \hat{\boldsymbol{x}}^{(u-1)}\|_2 < \varepsilon_1$, for the stopping value $\varepsilon$ sufficiently small. Then, the final position of the procedure is considered to be the coordinates of the mobile device. The possible outcomes of an iterative procedure and how the estimated position of the mobile device is taken from that procedure are carefully analyzed in section 3.3.4.

In a nutshell, we propose the Algorithm 6, a Gauss-Newton iterative procedure of Maximum Likelihood estimator.

## 6.4 Mobile-based localization by Downlink DoA

### 6.4.1 Problem Formulation

Direction-based localization at the mobile device is much more complicated than one at the base stations. As the orientation of the mobile device is unknown, the azimuth and elevation angles of the incident wave are not able to be estimated. The DoA estimations at mobile device are more erroneous, because of its limitation in size. However, there are the angles remain unchanged regardless of the mobile device's orientation. The mathematical problems are still challenging to solve.

### 6.4.2 Linking the DDoA to the related azimuth and elevation angles

The azimuth and elevation angles are defined similarly to the section 4.4, but the $(xy)$ plane and the $z$-axis are attached to the mobile device.

We define $\beta_{i,j}$ as the Direction Difference of Arrival (DDoA) between incident waves from $i$-th base station and $j$-th base station, $d_i$ and $d_j$ are the distance from the mobile device to $i$-th base station and $j$-th base station.

To calculate $\beta_{i,j}$, we use scalar product of $\overrightarrow{d_i}$ and $\overrightarrow{d_j}$, the vector demonstrating the incident signal from $i$-th and $j$-th base station, respectively.

We have

$\overrightarrow{d_i} = (d_i \cos\theta_i \cos\varphi_i,\ d_i \cos\theta_i \sin\varphi_i,\ d_i \sin\theta_i)$

$\overrightarrow{d_j} = (d_j \cos\theta_j \cos\varphi_j,\ d_j \cos\theta_j \sin\varphi_j,\ d_j \sin\theta_j)$

**Algorithm 6:** Proposed Maximum Likelihood estimator with estimation of $\boldsymbol{k}$

Take the measured Direction of Arrival: azimuth $\hat{\varphi}_i$ and elevation $\hat{\theta}_i$.
Take the orientation of the mobile device via vector $\overrightarrow{v}$ and angle $\xi$.
Compute the rotation matrix $\boldsymbol{R}$ by (6.7)
Compute the coordinates of all the base station in rotation coordinate system by (6.8)
Assign $u = 1$ and $\varepsilon_1$ sufficiently small.
Assign the coordinate vector computed by (6.55) as the first estimated coordinate vector $\widehat{\boldsymbol{x}^r}^{(1)}$ of the mobile device.
**repeat**
    Compute the estimated azimuth $\hat{\varphi}_i$ and elevation $\hat{\theta}_i$ by (6.22) and (6.23), respectively.
    **if** $|\varphi_i(\hat{\boldsymbol{x}}^{(u)}) - \hat{\varphi}_i| \geq \pi$ **then**
        $\widehat{k}_i = \text{sign}(\varphi_i(\hat{\boldsymbol{x}}^{(u)}) - \hat{\varphi}_i)$
    **else**
        $\widehat{k}_i = 0$ ;
    Compute the following estimated coordinate vector $\hat{\boldsymbol{x}}^{(u+1)}$ of the mobile device by (6.26).
    $u = u + 1$;
**until** $\|\boldsymbol{x}^{(u)} - \boldsymbol{x}^{(u-1)}\|_2 < \varepsilon_1$ *or* $u > 1000$ *or* $\|\boldsymbol{x}^{(u)}\|_2 = \pm\infty$;
**if** $u > 1000$ *or* $\|\hat{\boldsymbol{x}}^{(u)}\|_2 = \pm\infty$ **then**
    $\hat{\boldsymbol{x}}^{(1)}$ is the estimated position of the mobile device;
**else**
    $\hat{\boldsymbol{x}}^{(u)}$ is the estimated position of the mobile device;
Compute the coordinate of the mobile device in true coordinate system by (6.9).

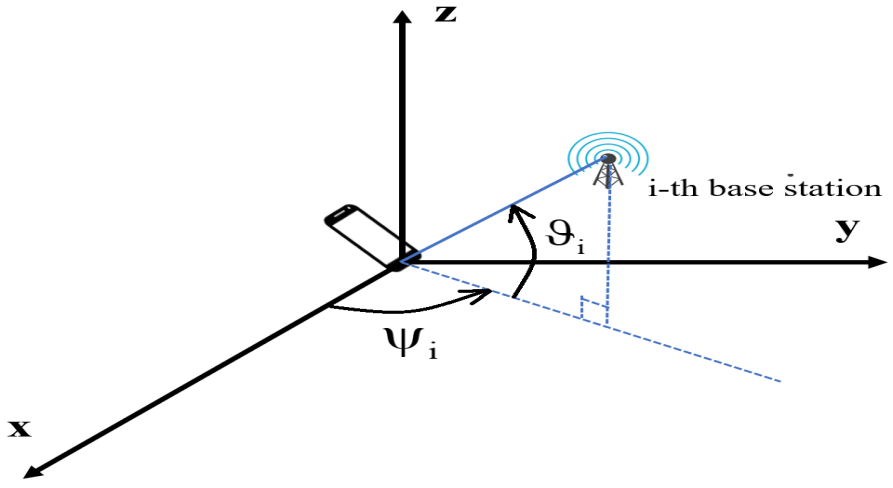



Figure 6.1: Localization at a base station

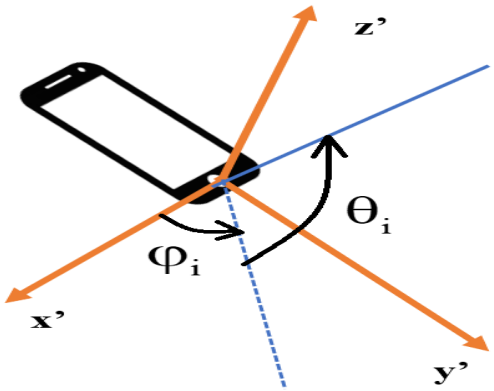



Figure 6.2: Incident wave from $i$-th base station to the mobile device in the relative coordinate system

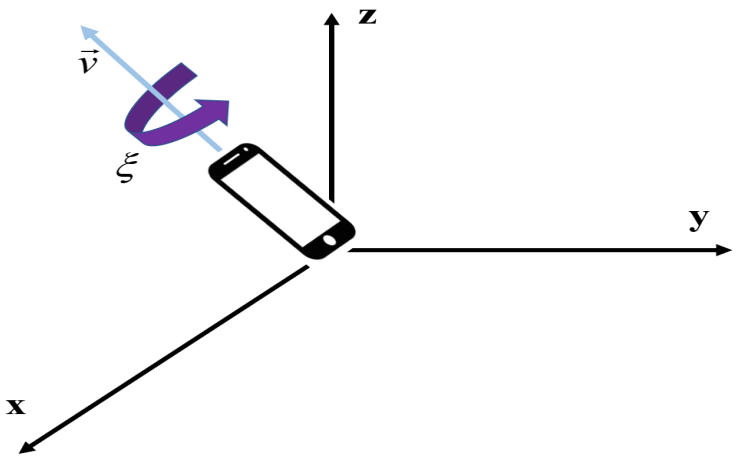



Figure 6.3: Quaternion with a unit vector $\vec{v}$ and a rotation angle $\xi$ to express the orientation of the mobile device in 3D space.

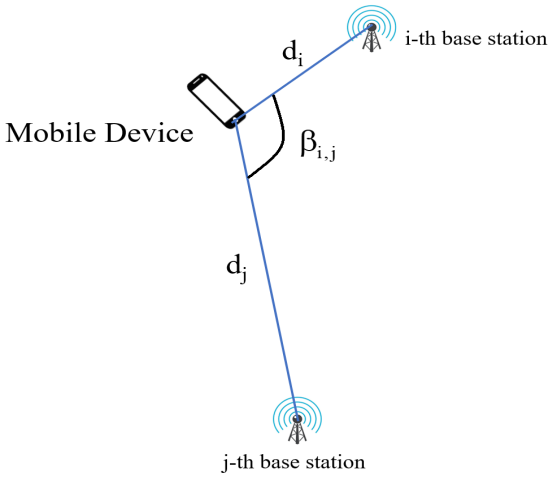



Figure 6.4: Localization at mobile device with Direction Difference of Arrival (DDoA)

Thus

$$\begin{aligned}&\overrightarrow{d_i}.\overrightarrow{d_j}\\&= d_i d_j(\cos\theta_i\cos\varphi_i\cos\theta_j\cos\varphi_j + \cos\theta_i\sin\varphi_i\cos\theta_j\sin\varphi_j + \sin\theta_i\sin\theta_j)\\&= d_i d_j(\cos\theta_i\cos\theta_j\cos(\varphi_j-\varphi_i) + \sin\theta_i\sin\theta_j)\end{aligned} \tag{6.30}$$

where $d_i$ and $d_j$ are the length of two vectors $\overrightarrow{d_i}$ and $\overrightarrow{d_j}$, respectively
The definition of scalar product of two vectors:

$$\overrightarrow{d_i}.\overrightarrow{d_j} = d_i.d_j.\cos\beta_{i,j} \tag{6.31}$$

Thus

$$\cos\beta_{i,j} = \cos\theta_i\cos\theta_j\cos(\varphi_j-\varphi_j) + \sin\theta_i\sin\theta_j \tag{6.32}$$

Considering that $\beta_{i,j} \in [0;\pi]$, we have

$$\beta_{i,j} = \arccos\left(\cos\theta_i\cos\theta_j\cos(\varphi_i-\varphi_j) + \sin\theta_i\sin\theta_j\right) \tag{6.33}$$

Let $\gamma_{i,j} = \cos\beta_{i,j}$
(6.33) reveals the relationship between the measured DOAs $\theta_i$, $\varphi_i$, $\theta_j$, $\varphi_j$ and the DDoA $\beta_{i,j}$
Value of $\beta_{i,j}$ always remains unchanged when the mobile device rotates. [51] proves that the DDoA is unchanged no matter which coordinate system is chosen.
In practice, the estimated $\hat{\gamma}_{i,j}$ is computed by (D.1) in Appendix D.

### 6.4.3 Least Squares method

We have
$\overrightarrow{d_i} = (x_i - x,\ y_i - y,\ z_i - z)$

$\overrightarrow{d_j} = (x_j - x,\ y_j - y,\ z_j - z)$

The DDoA between the incident waves from $i$-th base station and $j$-th base station does not depend on the orientation of the mobile device.

$$\overrightarrow{d_i}.\overrightarrow{d_j} = x^2 - (x_i + x_j)x + x_i x_j + y^2 - (y_i + y_j)y + y_i y_j + z^2 - (z_i + z_j)z + z_i z_j \tag{6.34}$$

$\overrightarrow{d_i}.\overrightarrow{d_j} = d_i d_j \cos\beta_{i,j} = d_i d_j \gamma_{i,j}$

Thus

$$x^2 - (x_i + x_j)x + x_i x_j + y^2 - (y_i + y_j)y + y_i y_j + z^2 - (z_i + z_j)z + z_i z_j = d_i d_j \gamma_{i,j} \tag{6.35}$$

Moreover,

$$d_i = \sqrt{(x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2} \tag{6.36}$$

As a result

$$\begin{aligned} &x^2 - (x_i + x_j)x + x_i x_j + y^2 - (y_i + y_j)y + y_i y_j + z^2 - (z_i + z_j)z + z_i z_j \\ &= \sqrt{(x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2}\sqrt{(x_j - x)^2 + (y_j - y)^2 + (z_j - z)^2}\gamma_{i,j} \end{aligned} \tag{6.37}$$

It is very difficult to solve the equation (6.37) with 3 variables $x, y, z$, but we can have an estimation. We take the square of (6.37). Let

$$\boldsymbol{a}^{(i,j)} = \begin{bmatrix} 1 \\ (x_i + x_j)^2 \\ (y_i + y_j)^2 \\ (z_i + z_j)^2 \\ -2(x_i + x_j) \\ -2(y_i + y_j) \\ -2(z_i + z_j) \\ 2(x_i + x_j)(y_i + y_j) \\ 2(y_i + y_j)(z_i + z_j) \\ 2(z_i + z_j)(x_i + x_j) \\ 2(x_i x_j + y_i y_j + z_i z_j) \\ -2(x_i + x_j)(x_i x_j + y_i y_j + z_i z_j) \\ -2(y_i + y_j)(x_i x_j + y_i y_j + z_i z_j) \\ -2(z_i + z_j)(x_i x_j + y_i y_j + z_i z_j) \end{bmatrix}^T ; \boldsymbol{\omega} = \begin{bmatrix} (x^2 + y^2 + z^2)^2 \\ x^2 \\ y^2 \\ z^2 \\ x(x^2 + y^2 + z^2) \\ y(x^2 + y^2 + z^2) \\ z(x^2 + y^2 + z^2 \\ xy \\ yz \\ zx \\ x^2 + y^2 + z^2 \\ x \\ y \\ z \end{bmatrix}$$

$$\boldsymbol{b}^{(i,j)} = \begin{bmatrix} 1 \\ 4x_ix_j \\ 4y_iy_j \\ 4z_iz_j \\ -2(x_i+x_j) \\ -2(y_i+y_j) \\ -2(z_i+z_j) \\ 4(x_iy_j+y_ix_j) \\ 4(y_iz_j+z_iy_j) \\ 4(z_ix_j+x_iz_j) \\ x_i^2+y_i^2+z_i^2+x_j^2+y_j^2+z_j^2 \\ -2x_i(x_j^2+y_j^2+z_j^2)-2x_j(x_i^2+y_i^2+z_i^2) \\ -2y_i(x_j^2+y_j^2+z_j^2)-2y_j(x_i^2+y_i^2+z_i^2) \\ -2z_i(x_j^2+y_j^2+z_j^2)-2z_j(x_i^2+y_i^2+z_i^2) \end{bmatrix}^T$$

Noting that $\boldsymbol{a}^{(i,j)} = \boldsymbol{a}^{(j,i)}$ and $\boldsymbol{b}^{(i,j)} = \boldsymbol{b}^{(j,i)}$.

Therefore, by taking the square of the left hand side and the right hand side of equation (6.37), we have:

$$\boldsymbol{a}^{(i,j)}\boldsymbol{\omega} + (x_ix_j + y_iy_j + z_iz_j)^2 = \gamma_{i,j}^2\boldsymbol{b}^{(i,j)}\boldsymbol{\omega} + \gamma_{i,j}^2(x_i^2+y_i^2+z_i^2)(x_j^2+y_j^2+z_j^2) \quad (6.38)$$

In matrix formulation, we denote

$$\hat{\boldsymbol{A}} = \begin{bmatrix} \boldsymbol{a}^{(1,2)} - \hat{\gamma}_{1,2}^2\boldsymbol{b}^{(1,2)} \\ \boldsymbol{a}^{(1,3)} - \hat{\gamma}_{1,3}^2\boldsymbol{b}^{(1,3)} \\ \cdots \\ \boldsymbol{a}^{(i,j)} - \hat{\gamma}_{i,j}^2\boldsymbol{b}^{(i,j)} \end{bmatrix} \quad (6.39)$$

$$\hat{\boldsymbol{h}} = \begin{bmatrix} (\hat{\gamma}_{1,2}^2)(x_1^2+y_1^2+z_1^2)(x_2^2+y_2^2+z_2^2) - (x_1x_2+y_1y_2+z_1z_2)^2 \\ (\hat{\gamma}_{1,3}^2)(x_1^2+y_1^2+z_1^2)(x_3^2+y_3^2+z_3^2) - (x_1x_3+y_1y_3+z_1z_3)^2 \\ \cdots \\ (\hat{\gamma}_{i,j}^2)(x_i^2+y_i^2+z_i^2)(x_j^2+y_j^2+z_j^2) - (x_ix_j+y_iy_j+z_iz_j)^2 \end{bmatrix} \quad (6.40)$$

where $i$ from 1 to $N-1$, $j$ from 2 to $N$, $i < j$

We have the equation of approximation

$$\hat{\boldsymbol{A}}\,\boldsymbol{\omega} = \hat{\boldsymbol{h}} \quad (6.41)$$

We have

$$\hat{\boldsymbol{\omega}} = \min_{\boldsymbol{\omega}} \|\hat{\boldsymbol{A}}\boldsymbol{\omega} - \hat{\boldsymbol{h}}\|^2 \quad (6.42)$$

leading to the estimate of $\boldsymbol{\omega}$ being calculated by

Least-Square estimation of $\omega$

$$\hat{\boldsymbol{\omega}} = \boldsymbol{A}^{\dagger}\hat{\boldsymbol{h}} \quad (6.43)$$

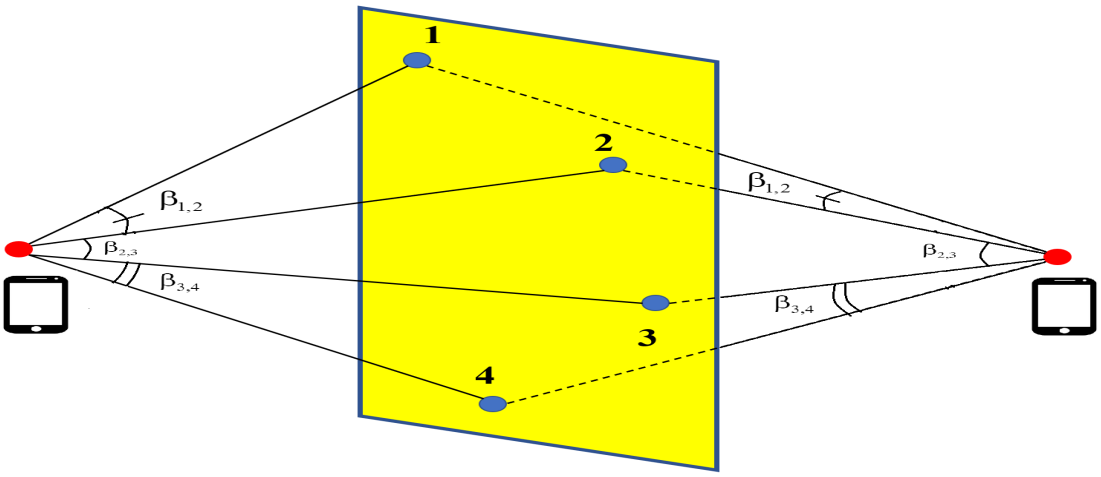



Figure 6.5: Scenario when all the base stations are co-planar. Blue dots stand for base stations. Red dots stand for mobile devices. The yellow plane is the plane containing all the base stations.

where $\boldsymbol{A}^{\dagger} = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T$

Estimated coordinate vector of the mobile device are the 3 last elements (the 12-th, 13-th and 14-th elements) of $\hat{\boldsymbol{\omega}}$

$$\hat{\boldsymbol{x}} = \begin{bmatrix}[\hat{\boldsymbol{\omega}}]_{12} & [\hat{\boldsymbol{\omega}}]_{13} & [\hat{\boldsymbol{\omega}}]_{14}\end{bmatrix}^T \tag{6.44}$$

### 6.4.4 Conditions for DDoA-based positioning

#### Placement of the base stations

All the base stations must not be co-planar.

When all the base stations are co-planar (Fig. 6.5), we take the point which is the symmetry of the mobile device with respect to the plane containing all the base stations, and put another mobile device at that point. All the DDoAs to the new point are exactly the same as the DDoAs to the original point. There is absolutely no method to distinguish the two points just by the set of DDoAs.

Therefore, the first condition is that there must be no plane that can contain all the base stations.

#### Number of base stations

As $\boldsymbol{\omega}$ has 14 elements, at least 14 DDoAs are required to make (6.41) not an underdetermined system. $\beta_{i,j} = \beta_{j,i}$ so with $N$ base stations, we have $N(N-1)/2$ DDoAs. Therefore $N(N-1)/2 \geq 14$ or $N \geq 6$. The minimum number of base stations for the Least Squares method to be feasible is 6.

### 6.4.5 Optimizing position by an iterative Maximum Likelihood procedure

To optimize $\hat{\boldsymbol{x}}$ obtained in (6.44), an iterative Maximum Likelihood estimator is applied.

In vector formulation, we denote

$$\hat{\boldsymbol{\gamma}} = \begin{bmatrix} \hat{\gamma}_{1,2} & \hat{\gamma}_{1,3} & \dots & \hat{\gamma}_{1,N} \end{bmatrix}^T \tag{6.45}$$

$$\boldsymbol{f}_\gamma(\boldsymbol{x}) = \begin{bmatrix} \gamma_{1,2}(\boldsymbol{x}) & \gamma_{1,3}(\boldsymbol{x}) & \dots & \gamma_{1,N}(\boldsymbol{x}) \end{bmatrix}^T \tag{6.46}$$

$$\gamma_{i,j}(\boldsymbol{x}) = \frac{x^2 - (x_i + x_j)x + x_i x_j + y^2 - (y_i + y_j)y + y_i y_j + z^2 - (z_i + z_j)z + z_i z_j}{\sqrt{(x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2}\sqrt{(x_j - x)^2 + (y_j - y)^2 + (z_j - z)^2}} \tag{6.47}$$

where $\boldsymbol{x} = [x \;\; y \;\; z]^T$.

We have

$$\boldsymbol{C}_\gamma = \text{cov}(\hat{\boldsymbol{\gamma}}) = \begin{bmatrix} s^2_{1,2} & s^2_{1,2,3} & \dots & s^2_{1,2,N} \\ s^2_{1,2,3} & s^2_{1,3} & \dots & s^2_{1,3,N} \\ \dots & \dots & \dots & \dots \\ s^2_{1,2,N} & s^2_{1,3,N} & \dots & s^2_{1,N} \end{bmatrix} \tag{6.48}$$

where $s^2_{i,j}$ and $s^2_{i,j,l}$ are expressed in the Appendix D.

The measurement vector $\hat{\boldsymbol{\gamma}}$ is Gaussian distributed with mean vector of $\boldsymbol{f}$ and covariance matrix $\boldsymbol{C}_\gamma$, we have the probability density function (pdf):

$$\text{p}(\hat{\boldsymbol{\gamma}}|\boldsymbol{x}) = \frac{(2\pi)^{-N/2}}{|\boldsymbol{C}_\gamma|^{1/2}} \exp\left[\frac{-1}{2}(\hat{\boldsymbol{\gamma}} - \boldsymbol{f}_\gamma)^T \boldsymbol{C}_\gamma^{-1}(\hat{\boldsymbol{\gamma}} - \boldsymbol{f}_\gamma)\right] \tag{6.49}$$

Maximizing the pdf in (6.49) is equivalent to finding

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} (\hat{\boldsymbol{\gamma}} - \boldsymbol{f}_\gamma(\boldsymbol{x}))^T \boldsymbol{C}_\gamma (\hat{\boldsymbol{\gamma}} - \boldsymbol{f}_\gamma(\boldsymbol{x})) \tag{6.50}$$

which we shall perform alternatingly.

The possible outcomes of an iterative procedure and how the estimated position of the mobile device is taken from that procedure are carefully analyzed in section 3.3.4.

In a nutshell, the Algorithm 7 is proposed for the Gauss-Newton iterative procedure of minimizing the Cost Function.

## 6.5 Hybrid ToA-DDoA localization

In this section, ToA estimation is added to simplify the computation. When $d_i$ and $d_j$ are already estimated by the related ToA, the equation (6.35) can be utilized for estimating position.

**Algorithm 7:** Proposed Maximum Likelihood estimator

Take the measured Direction of Arrival: azimuth $\hat{\varphi}_i$ and elevation $\hat{\theta}_i$.
Compute $\gamma_{i,j}$ by (D.1).
Assign $u = 1$ and $\varepsilon_\gamma$ sufficiently small.
Assign the coordinate computed by (6.44) as the first estimated coordinate vector $\hat{\boldsymbol{x}}^{(1)}$ of the mobile device.
**repeat**
    Compute the estimated DDoA by (6.57)
    Compute the following estimated coordinate vector $\hat{\boldsymbol{x}}^{(u+1)}$ of the mobile device by (6.66).
    $u = u + 1$;
**until** $\|\hat{\boldsymbol{x}}^{(u+1)} - \hat{\boldsymbol{x}}^{(u)}\|_2 < \varepsilon_\gamma$ *or* $u > 1000$ *or* $\|\hat{\boldsymbol{x}}^{(u+1)}\| = \pm\infty$;
**if** $u > 1000$ *or* $\|\hat{\boldsymbol{x}}^{(u+1)}\|_2 = \pm\infty$ **then**
    $\hat{\boldsymbol{x}}^{(1)}$ is the estimated position of the mobile device;
**else**
    $\hat{\boldsymbol{x}}^{(u)}$ is the estimated position of the mobile device;

### 6.5.1 Least Squares method

In matrix formulation, we denote

$$\boldsymbol{A}_{\text{hyb}} = \begin{bmatrix} -(x_1 + x_2) & -(y_1 + y_2) & -(z_1 + z_2) & 1 \\ -(x_1 + x_3) & -(y_1 + y_3) & -(z_1 + z_3) & 1 \\ \cdots & \cdots & \cdots & \cdots \\ -(x_1 + x_N) & -(y_1 + y_N) & -(z_1 + z_N) & 1 \end{bmatrix}$$

$$\boldsymbol{\omega}_{\text{hyb}} = \begin{bmatrix} x \\ y \\ z \\ x^2 + y^2 + z^2 \end{bmatrix} \; ; \; \hat{\boldsymbol{b}}_{\text{hyb}} = \begin{bmatrix} \hat{d}_1\hat{d}_2\hat{\gamma}_{1,2} \\ \hat{d}_1\hat{d}_3\hat{\gamma}_{1,3} \\ \cdots \\ \hat{d}_1\hat{d}_N\hat{\gamma}_{1,N} \end{bmatrix}$$

where $\hat{d}_i = c\hat{t}_i$ and $\hat{\gamma}_{i,j}$ is estimated by (D.1).

We have

$$\hat{d}_i\hat{d}_j\hat{\gamma}_{i,j} = (d_i + \tilde{d}_i)(d_j + \tilde{d}_j)(\gamma_{i,j} + \tilde{\gamma}_{i,j}) \approx d_i d_j \gamma_{i,j} + \tilde{d}_i d_j \gamma_{i,j} + d_i \tilde{d}_j \gamma_{i,j} + d_i d_j \tilde{\gamma}_{i,j} \tag{6.51}$$

As a result, $\hat{\boldsymbol{b}}_{\text{hyb}} = \boldsymbol{b}_{\text{hyb}} + \tilde{\boldsymbol{b}}_{\text{hyb}}$ where

$$\boldsymbol{b}_{\text{hyb}} = \begin{bmatrix} d_1 d_2 \gamma_{1,2} \\ d_1 d_3 \gamma_{1,3} \\ \cdots \\ d_1 d_N \gamma_{1,N} \end{bmatrix} \; ; \; \tilde{\boldsymbol{b}}_{\text{hyb}} \approx \begin{bmatrix} \tilde{d}_1 d_2 \gamma_{1,2} + d_1 \tilde{d}_2 \gamma_{1,2} + d_1 d_2 \tilde{\gamma}_{1,2} \\ \tilde{d}_1 d_3 \gamma_{1,3} + d_1 \tilde{d}_3 \gamma_{1,3} + d_1 d_3 \tilde{\gamma}_{1,3} \\ \cdots \\ \tilde{d}_1 N \gamma_{1,N} + d_1 \tilde{d}_N \gamma_{1,N} + d_1 d_N \tilde{\gamma}_{1,N} \end{bmatrix}$$

The covariance matrix of $\hat{\boldsymbol{b}}_{\text{hyb}}$

$$\mathrm{E}\left(\tilde{\boldsymbol{b}}_{\text{hyb}}\tilde{\boldsymbol{b}}_{\text{hyb}}^T\right) = \begin{bmatrix} \sigma_1^2 d_2^2\gamma_{1,2}^2 + \sigma_2^2 d_1^2\gamma_{1,2}^2 + s_{1,2}^2 d_1^2 d_2^2 & \sigma_1^2 d_2 d_3\gamma_{1,2}\gamma_{1,3} + s_{1,2,3}^2 d_1^2 d_2 d_3 & \dots & \sigma_1^2 d_2 d_N\gamma_{1,2}\gamma_{1,N} + s_{1,2,N}^2 d_1^2 d_2 d_N \\ \sigma_1^2 d_2 d_3\gamma_{1,2}\gamma_{1,3} + s_{1,2,3}^2 d_1^2 d_2 d_3 & \sigma_1^2 d_3^2\gamma_{1,3}^2 + \sigma_3^2 d_1^2\gamma_{1,3}^2 + s_{1,3}^2 d_1^2 d_3^2 & \dots & \sigma_1^2 d_3 d_N\gamma_{1,3}\gamma_{1,N} + s_{1,3,N}^2 d_1^2 d_3 d_N \\ \dots & \dots & \dots & \dots \\ \sigma_1^2 d_2 d_N\gamma_{1,2}\gamma_{1,N} + s_{1,2,N}^2 d_1^2 d_2 d_N & \sigma_1^2 d_3 d_N\gamma_{1,3}\gamma_{1,N} + s_{1,3,N}^2 d_1^2 d_3 d_N & \dots & \sigma_1^2 d_N^2\gamma_{1,N}^2 + \sigma_N^2 d_1^2\gamma_{1,N}^2 + s_{1,N}^2 d_1^2 d_N^2 \end{bmatrix} \tag{6.52}$$

The Weighted Least Square (WLS) cost function

$$J_{WLS} = (\boldsymbol{A}_{\text{hyb}}\boldsymbol{\omega}_{\text{hyb}} - \boldsymbol{b}_{\text{hyb}})^T \boldsymbol{W}_{\text{hyb}} (\boldsymbol{A}_{\text{hyb}}\boldsymbol{\omega}_{\text{hyb}} - \boldsymbol{b}_{\text{hyb}}) \tag{6.53}$$

where $\boldsymbol{W}_{\text{hyb}}$ is a symmetric weighting matrix.

We choose $\boldsymbol{W}_{\text{hyb}} = \left[\mathrm{E}\left(\tilde{\boldsymbol{b}}_{\text{hyb}}\tilde{\boldsymbol{b}}_{\text{hyb}}^T\right)\right]^{-1}$ with $\mathrm{E}\left(\tilde{\boldsymbol{b}}_{\text{hyb}}\tilde{\boldsymbol{b}}_{\text{hyb}}^T\right)$ is expressed by (6.52).

The WLS estimate of $\boldsymbol{\omega}_{\text{hyb}}$ is

$$\hat{\boldsymbol{\omega}}_{\text{hyb}} = \left(\boldsymbol{A}_{\text{hyb}}^T \boldsymbol{W}_{\text{hyb}} \boldsymbol{A}_{\text{hyb}}\right)^{-1} \boldsymbol{A}_{\text{hyb}}^T \boldsymbol{W}_{\text{hyb}} \hat{\boldsymbol{b}}_{\text{hyb}} \tag{6.54}$$

The estimated coordinate vector of the mobile device comprises the 3 first elements of $\hat{\boldsymbol{\omega}}_{\text{hyb}}$:

$$\hat{\boldsymbol{x}} = \begin{bmatrix} [\hat{\boldsymbol{\omega}}_{\text{hyb}}]_1 & [\hat{\boldsymbol{\omega}}_{\text{hyb}}]_2 & [\hat{\boldsymbol{\omega}}_{\text{hyb}}]_3 \end{bmatrix}^T \tag{6.55}$$

### 6.5.2 Iterative Maximum Likelihood Procedure

We propose an iterative Maximum Likelihood estimator, which uses the positioning result in (6.55) as its initialization, to optimize the estimation.

Linking the distance to the coordinates, we have:

$$d_i(\boldsymbol{x}) = \sqrt{(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2} \tag{6.56}$$

From the equation (6.35), we have the relation between the DDoA and the coordinates:

$$\gamma_{i,j}(\boldsymbol{x}) = \cos\beta_{i,j}(\boldsymbol{x}) = \frac{x^2 - (x_i + x_j)x + x_i x_j + y^2 - (y_i + y_j)y + y_i y_j + z^2 - (z_i + z_j)z + z_i z_j}{\sqrt{(x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2}\sqrt{(x_j - x)^2 + (y_j - y)^2 + (z_j - z)^2}} \tag{6.57}$$

with $i$ from 1 to $N - 1$ and $j$ from $i + 1$ to $N$.

Then, we denote

$$\hat{\boldsymbol{d}} = \begin{bmatrix} \hat{d}_1 & \hat{d}_2 & \dots & \hat{d}_N \end{bmatrix}^T \tag{6.58}$$

$$\hat{\boldsymbol{\gamma}} = \begin{bmatrix} \hat{\gamma}_1 & \hat{\gamma}_2 & \dots & \hat{\gamma}_N \end{bmatrix}^T \tag{6.59}$$

$$\hat{\boldsymbol{r}} = \begin{bmatrix} \hat{\boldsymbol{d}} & \hat{\boldsymbol{\gamma}} \end{bmatrix}^T \tag{6.60}$$

$$\boldsymbol{f}_r(\boldsymbol{x}) = \begin{bmatrix} \sqrt{(x-x_1)^2+(y-y_1)^2+(z-z_1)^2} \\ \sqrt{(x-x_2)^2+(y-y_2)^2+(z-z_2)^2} \\ \cdots \\ \sqrt{(x-x_N)^2+(y-y_N)^2+(z-z_N)^2} \\ \gamma_{1,1}(\boldsymbol{x}) \\ \gamma_{1,2}(\boldsymbol{x}) \\ \cdots \\ \gamma_{1,N}(\boldsymbol{x}) \end{bmatrix} \tag{6.61}$$

where $\gamma_{i,j}(\boldsymbol{x})$ is defined in (6.57).

We have $\mathrm{cov}(\hat{\boldsymbol{d}}) = \boldsymbol{C}_\mathrm{d}$ and

$$\boldsymbol{C}_\gamma = \mathrm{cov}(\hat{\boldsymbol{\gamma}}) = \begin{bmatrix} s^2_{1,2} & s^2_{1,2,3} & \cdots & s^2_{1,2,N} \\ s^2_{1,2,3} & s^2_{1,3} & \cdots & s^2_{1,3,N} \\ \cdots & \cdots & \cdots & \cdots \\ s^2_{1,2,N} & s^2_{1,3,N} & \cdots & s^2_{1,N} \end{bmatrix} \tag{6.62}$$

where $s^2_{i,j}$ and $s^2_{i,j,l}$ are expressed in the Appendix D.

The covariance matrix of $\hat{\boldsymbol{r}}$

$$\boldsymbol{C}_r = \mathrm{cov}(\hat{\boldsymbol{r}}) = \begin{bmatrix} \boldsymbol{C}_\mathrm{d} & \boldsymbol{0}_{N\times(N-1)} \\ \boldsymbol{0}_{(N-1)\times N} & \boldsymbol{C}_\gamma \end{bmatrix} \tag{6.63}$$

where $\boldsymbol{0}_{a\times b}$ is the null matrix of the size $a \times b$.

The measurement vector $\hat{\boldsymbol{r}}$ is Gaussian distributed with mean vector of $\boldsymbol{f}$ and covariance matrix $\boldsymbol{C}_r$, we have the probability density function (pdf):

$$\mathrm{p}(\hat{\boldsymbol{r}}|\boldsymbol{x}) = \frac{(2\pi)^{-N/2}}{|\boldsymbol{C}_r|^{1/2}} \exp\left[\frac{-1}{2}(\hat{\boldsymbol{r}}-\boldsymbol{f}_r)^T\boldsymbol{C}_r^{-1}(\hat{\boldsymbol{r}}-\boldsymbol{f}_r)\right] \tag{6.64}$$

Maximizing the pdf in (6.64) is equivalent to finding

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}}(\hat{\boldsymbol{r}} - \boldsymbol{f}_r(\boldsymbol{x}))^T\boldsymbol{C}_r(\hat{\boldsymbol{r}} - \boldsymbol{f}_r(\boldsymbol{x})) \tag{6.65}$$

which we shall perform alternatingly. We consider Gauss-Newton procedure [22] for $\hat{\boldsymbol{x}}$. At iteration $(u+1)$:

$$\hat{\boldsymbol{x}}^{(u+1)} = \hat{\boldsymbol{x}}^{(u)} + \left(\boldsymbol{G}_r^T\boldsymbol{C}_r\boldsymbol{G}_r\right)^{-1}\boldsymbol{G}_r^T\boldsymbol{C}_r\left(\hat{\boldsymbol{r}} - \boldsymbol{f}(\hat{\boldsymbol{x}}^{(u)}\right) \tag{6.66}$$

where $\boldsymbol{G}_r$ is the Jacobian matrix of $\boldsymbol{f}_r(\boldsymbol{x})$

$$\boldsymbol{G}_r = \boldsymbol{G}\left(\hat{\boldsymbol{x}}^{(u)}\right),\ \boldsymbol{G}(\boldsymbol{x}) = \frac{\partial \boldsymbol{f}(\boldsymbol{x})}{\partial \boldsymbol{x}^T}. \tag{6.67}$$

A procedure is expected to terminate when $\|\hat{\boldsymbol{x}}^{(u+1)} - \hat{\boldsymbol{x}}^{(u)}\|_2 < \varepsilon_r$, for the stopping criterion $\varepsilon_r$ sufficiently small. Then, the final position of the procedure is considered to be the coordinates of the mobile device in the $xyz$ space.

In a nutshell, the Algorithm 8 is proposed for the Gauss-Newton iterative procedure of Maximum Likelihood estimator.

**Algorithm 8:** Proposed Maximum Likelihood estimtor with Gauss-Newton procedure

Take all the estimated ToAs and then compute the corresponding $\hat{d}_1, \hat{d}_2, \ldots, \hat{d}_N$.
Take the measured Direction of Arrival: azimuth $\hat{\varphi}_i$ and elevation $\hat{\theta}_i$.
Compute $\hat{\gamma}_{i,j}$ by (D.1).
Assign $u = 1$ and $\varepsilon$ sufficiently small.
Compute the estimation $\hat{\boldsymbol{x}}$ by (6.55) as the first estimated coordinates of the mobile device.
**repeat**
    Compute the following estimated coordinates $\hat{\boldsymbol{x}}^{(u+1)}$ of the mobile device by (6.66).
    $u = u + 1$;
**until** $\|\hat{\boldsymbol{x}}^{(u+1)} - \hat{\boldsymbol{x}}^{(u)}\|_2 < \varepsilon$ *or* $u > 1000$ *or* $\|\hat{\boldsymbol{x}}^{(u+1)}\|_2 = \pm\infty$;
**if** $u > 1000$ *or* $\|\hat{\boldsymbol{x}}^{(u+1)}\|_2 = \pm\infty$ **then**
    $\hat{\boldsymbol{x}}^{(1)}$ is the estimated position of the mobile device;
**else**
    $\hat{\boldsymbol{x}}^{(u)}$ is the estimated position of the mobile device;

## 6.6 Results

We simulate the algorithms in some scenarios, each scenario has a particular mobile orientation. In each scenario, we run simulations when the mobile orientation is known or unknown. If the mobile orientation is known, DoA-based algorithm in section 6.3 will be applied. Otherwise, DDoA-based algorithm in section 6.4 will be considered. It is expected that the DoA-based algorithm always gives a higher accuracy for location, because it requires the knowledge of mobile orientation.

We assume that all the estimations of azimuth and elevation angles have the same standard deviation: $\mu_1 = \mu_2 = \cdots = \mu_N = \nu_1 = \nu_2 = \cdots = \nu_N = \sigma$.

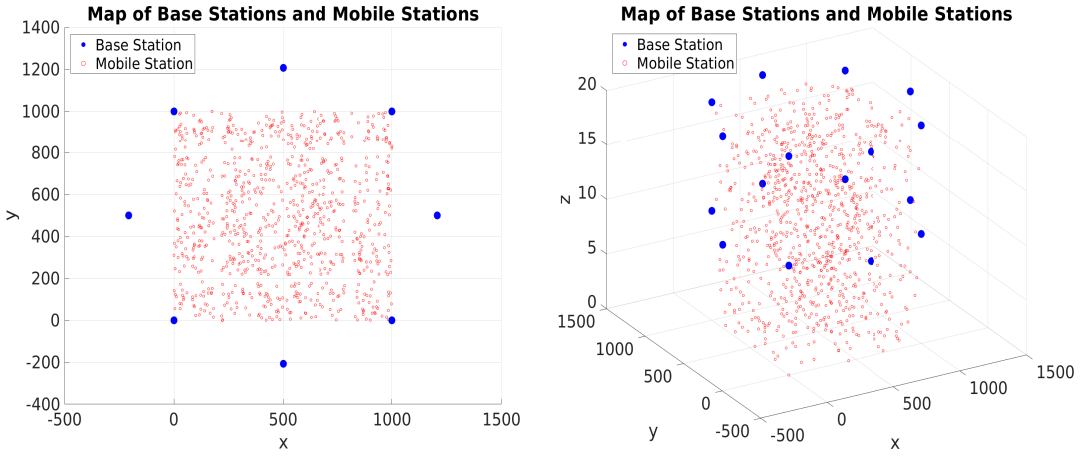
### 6.6.1 DDoA-based localization

Fig 6.7a, 6.7b and 6.7c illustrate the RMSEs of LS method and the ML estimator when the mobile orientation is taken and not taken into account, in 3 different mobile orientations. The results are summarized in Table 6.1.

From the figures, it is obvious that in most of the cases, a prior knowledge of mobile orientation increases noticeably the accuracy of position location. However, in some certain orientation, estimations of DDoA-based positioning algorithm are marginally more precise than those of DoA-based localization.

### 6.6.2 Hybrid DDoA-ToA localization

The standard deviations of all ToA measurements are also assumed to have the same value: $\sigma_{ToA,1} = \sigma_{ToA,2} = \cdots = \sigma_{ToA,N} = \sigma_{ToA}$.




(a) View from top

(b) View from one side

Figure 6.6: Map of base stations and random positions of the mobile device

| **Scenario** | **Unit vector** $\overrightarrow{v}$ | **Rotation angle** $\xi$ | **Results** |
|---|---|---|---|
| 1 | $[1 \;\; 0 \;\; 0]$ | $0°$ | DoA-based algorithm is much more accurate than DDoA-based |
| 2 | $\left[\frac{1}{\sqrt{3}} \;\; \frac{1}{\sqrt{3}} \;\; \frac{1}{\sqrt{3}}\right]$ | $120°$ | DDoA-based results are slighty more accurate the DoA-based results |
| 3 | $\left[\frac{-1}{\sqrt{3}} \;\; \frac{-1}{\sqrt{3}} \;\; \frac{-1}{\sqrt{3}}\right]$ | $60°$ | DoA-based algorithm is much more accurate than DDoA-based |

Table 6.1: Testing scenarios of different mobile orientations

We compare the RMSE of the four different scenarios, where in each scenario, the standard deviation of ToA measurements ($\sigma_{ToA}$) varies from 2.5 ns to 30 ns; the Weighted Least Squares (WLS) method and the Maximum Likelihood (ML) estimator are applied for localization (Fig. 6.8).

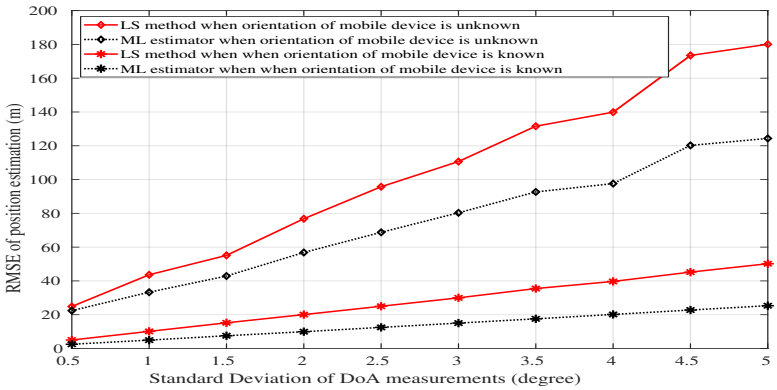
(a) ToA-based positioning algorithm given in [52].

(b) Hybrid positioning algorithm when $\sigma = 0.5°$.

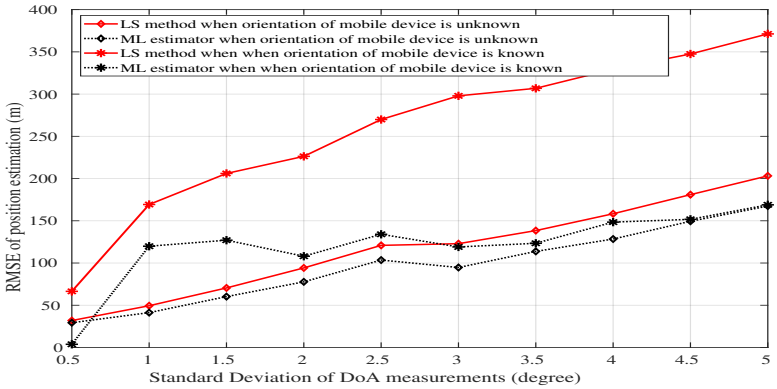
(c) Hybrid positioning algorithm when $\sigma = 1°$.

(d) Hybrid positioning algorithm when $\sigma = 1.5°$.

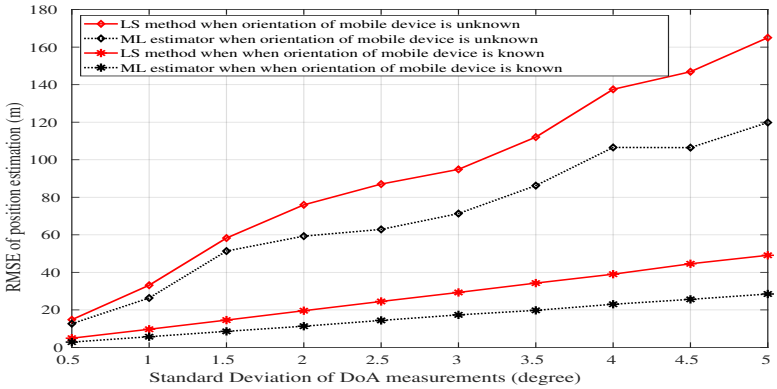
From Fig. 6.8, it is clear that the ML estimator considerably enhance the accuracy of WLS estimations. In addition, the advantage of the proposed hybrid positioning algorithm depends on the accuracy of DoA estimations. The more precise the DoA estimation is, the more useful the hybrid algorithm is, especially when the standard deviation of ToA estimation increases.




(a) $\overrightarrow{v} = [1 \;\; 0 \;\; 0]$, $\xi = 0°$ and $\sigma$ varies from 0.5° to 4°.




(b) $\overrightarrow{v} = \left[\frac{1}{\sqrt{3}} \;\; \frac{1}{\sqrt{3}} \;\; \frac{1}{\sqrt{3}}\right]$, $\xi = 120°$ and $\sigma$ varies from 0.5° to 4°.




(c) $\overrightarrow{v} = \left[\frac{-1}{\sqrt{3}} \;\; \frac{-1}{\sqrt{3}} \;\; \frac{-1}{\sqrt{3}}\right]$, $\xi = 60°$ and $\sigma$ varies from 0.5° to 4°.

Figure 6.7: Comparison of RMSE in DDoA-based localization

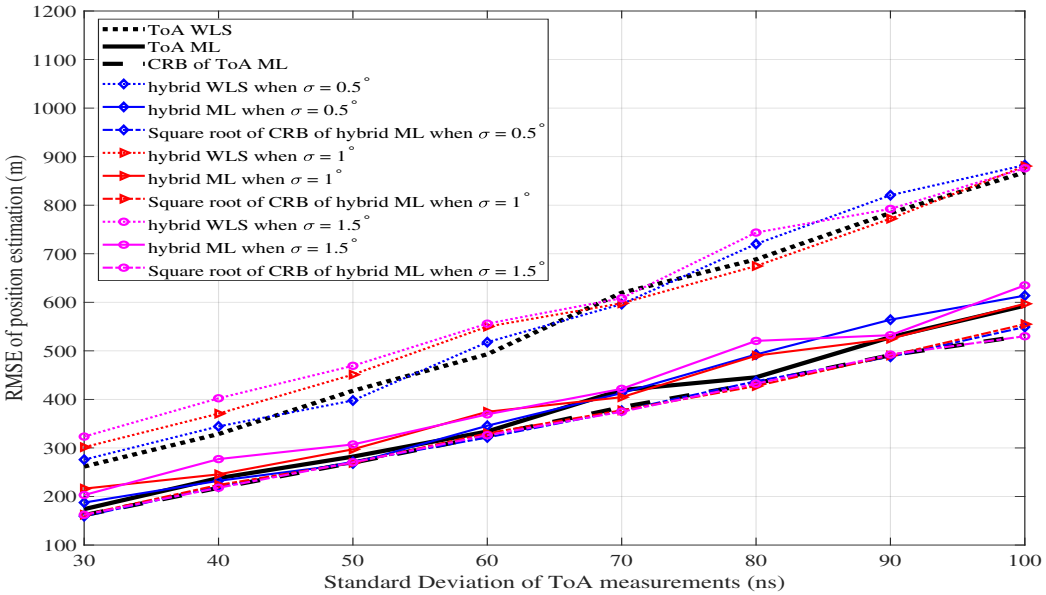



Figure 6.8: Localization at mobile device with hybrid ToA-DDoA

# Chapter 7

# Conclusions and Future works

In this dissertation, we looked at various aspects of positioning algorithms. In this final chapter, we provide some concluding remarks for each types of algorithms of the dissertation. Furthermore, we look at various possible straightforward extensions to the current work here and also some future topics which needs to be explored. We try to highlight both the pros and cons of our various proposed solutions and some possible future directions to circumvent the disadvantages associated with them.

## 7.1 Conclusion

This dissertation studies and analyzes the three main types of positioning algorithms.

### 7.1.1 Trilateration

In chapter 2, Trilateration is quite completely analyzed. In geometric approach, we propose an algorithm that can be applied for all the possible subcases. The matrix formulation with Least Squares method is much more accurate and less complicated than geometric approach.

In the range of trilateration positioning, the chapter proposes a new geometric method that can be applied in all measurement error cases. The numerical results shows, compared to the centroid algorithm and the Fermat Point algorithm, that the proposed approach helps to significantly improve the accuracy in localization and reduce the complexity by avoiding case division. Furthermore, experimental results also demonstrate that integrating path loss estimation can make the position estimation more accurate. Nevertheless, only results in 2D simulation are shown. As for 3D models, algorithms are more complicated and are currently being investigated.

### 7.1.2 Multilateration

In chapter 3, we continue to compare geometric approach and Least Squares method. The Maximum Likelihood estimator is applied with initialization by the both algorithms above. Practice-like simulations are implemented to verify the theory in practice.

This chapter also robustifies the accuracy of TDoA-based localization by a proposed ML estimator, whose performance is also facilitated by a more precise initialization acquired from the LS method. All the results are based on data obtained from realistic ray-tracing simulations, integrated in a MatLab environment.

Practical implementation is currently being carried out, to evaluate all the related positioning algorithms as realistically and objectively as possible.

### 7.1.3 Triangulation

Triangulation is the most complicated type of positioning algorithms which have the most sub-scenarios. We carefully study all the sub-scenarios.

- In chapter 4, the Network-Positioning problems are studied. The azimuth angle is defined in a new way, with the atan2 function and phase jump corrections. This new definition makes the position estimation much more precise, as well as reduce noticeably the time delay for localization processes.

  This chapter thoroughly analyzes a ML estimator with the DoA-based positioning algorithms using atan2 function and the k-correction in 2D scenarios. Moreover, an approximate ML estimator is also proposed. The simulations demonstrate the superior properties of our proposed algorithm: maintaining the unbiased property with the most accurate results and the shortest time delay. The approximate ML gives a better initialization for the true ML, which can augment the accuracy.

  In addition, the simulations demonstrate the superior properties of our proposed algorithm: maintaining the unbiased property with the most accurate results and the shortest time delay.

- In two-dimensional Self-Positioning which is studied in chapter 5, the new definition still brings a large enhancement in localization accuracy. Moreover, the new analysis for Least Squares method is investigated, which gives the ML iterative estimator a suitable initialization to converge.

  This chapter thoroughly analyzes the DDoA-based positioning algorithms using atan2 function and the k-correction to overcome the troubles caused by noises in angle measurements. The simulations demonstrate the superior properties of our proposed algorithm: unbiasedness (in certain conditions) with the most accurate results and the shortest time delay.

- In chapter 6 about Self-Positioning in 3D, we compute the DDoA from all the DoAs concerned, and then link the DDoA to the coordinates. ML estimator is also studied.

  This chapter studies direction-based self-positioning problems at mobile devices, in 2 scenarios: The orientation of the mobile device is known and unknown. When this orientation is undefined, the positioning problem is much more challenging. Consequently, a DDoA-based positioning algorithm is researched, because only the DDoAs do not change when the mobile device rotates. Analytical solution for

Least Squares method is presented. Moreover, a Maximum Likelihood estimator to optimize the position location is also applied.

The results show that Least Squares method is feasible and Maximum Likelihood estimator enhance the position estimation. Generally, a prior knowledge of mobile device's orientation gives more rigorous position estimations. However, in some certain orientations, the localization from DDoA-based algorithms is more accurate.

## 7.2 Future developments

Practical implementations are necessary in general. We believe that there is a great potential in this approach as it shows good generalization capability.

In Trilateration, the positioning algorithms are well studied in general. Practice-like simulations are also implemented to test the Multilateration positioning algorithms. However, practical measurements are still important.

In Triangulation, problems in multi-path and NLoS environment needs to be investigated.

Furthermore, diverse other location methods are also interesting to study: Mapping, Finger-Printing,...; Machine Learning, Deep Learning can also be utilized to improve the accuracy.

# Appendices

# Appendix A

# Vector Calculus

## A.1 Hessian matrix

Suppose $f : \mathbb{R}^n \to \mathbb{R}$ is a function taking as input a vector $\mathbf{x} \in \mathbb{R}^n$ and outputting a scalar $f(\mathbf{x}) \in \mathbb{R}$. If all second partial derivatives of $f$ exist and are continuous over the domain of the function, then the Hessian matrix $\mathbf{H}$ of $f$ is a square $n \times n$ matrix, usually defined and arranged as follows:

$$\mathbf{H}_f = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_N} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_1^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_N} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial^2 f}{\partial x_N \partial x_1} & \frac{\partial^2 f}{\partial x_1 \partial x_n} & \cdots & \frac{\partial^2 f}{\partial x_N^2} \end{bmatrix} \tag{A.1}$$

or, by stating an equation for the coefficients using indices $i$ and $j$:

$$(\mathbf{H}_f)_{i,j} = \frac{\partial f}{\partial x_i \partial x_j} \tag{A.2}$$

## A.2 Jacobian matrix

Suppose $f : \mathbb{R}^n \to \mathbb{R}^m$ is a function taking as input a vector $\mathbf{x} \in \mathbb{R}^n$ and outputting a scalar $f(\mathbf{x}) \in \mathbb{R}$. If all second partial derivatives of $f$ exist and are continuous over the domain of the function, then the Jacobian matrix $\mathbf{J}$ of $f$ is a square $m \times n$ matrix, usually defined and arranged as follows:

$$\mathbf{J}_f = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial f_m}{\partial x_1} & \frac{\partial f_m}{\partial x_2} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix} \tag{A.3}$$

or, by stating an equation for the coefficients using indices $i$ and $j$:

$$(\mathbf{J}_f)_{i,j} = \frac{\partial f_i}{\partial x_j} \tag{A.4}$$

## A.3 Gradient vector

The gradient of a scalar-valued differentiable function $f$ of several variables is the vector field (or vector-valued function) $\nabla f$ whose value at a point $p$ is the vector whose components are the partial derivatives of $f$ at $p$.

That is, for $f\colon \mathbb{R}^n \to \mathbb{R}$, its gradient $\nabla f\colon \mathbb{R}^n \to \mathbb{R}^n$ is defined at the point $p = (x_1, \ldots, x_n)$ in n-dimensional space as the vector:

$$\nabla f(p) = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \cdots \\ \frac{\partial f}{\partial x_n} \end{bmatrix} \tag{A.5}$$

# Appendix B

# Expected value and Variance of $d_i^2$ in RSS-based localization

## B.1 Expected value of $d_i^2$

From (2.20), we have

$$\frac{-(\ln 10)\zeta_{\mathrm{RSS},i}}{5\alpha_i} = 2(\ln 10)(\log d_i) + \frac{-(\ln 10)n_{\mathrm{RSS},i}}{5\alpha_i} = 2(\ln d_i) + \frac{-(\ln 10)n_{\mathrm{RSS},i}}{5\alpha_i} \tag{B.1}$$

As $n_{\mathrm{RSS},i} \sim \mathcal{N}(0, \sigma^2_{\mathrm{RSS},i})$, $\frac{-(\ln 10)\zeta_{\mathrm{RSS},i}}{5\alpha_i}$ has the mean value of $2(\ln d_i)$ and the variance of $\frac{(\ln 10)^2}{25\alpha_i^2}\sigma^2_{\mathrm{RSS},i}$.

It is already proved in [53] that if $q$ is a Gaussian distributed variable, the mean and variance of $q$ will be $\mathrm{E}(e^q) = \exp\left(\mu + \frac{\sigma^2}{2}\right)$ and $\mathrm{var}\,(e^q) = \left(e^{\sigma^2} - 1\right) e^{2\mu+\sigma^2}$. Applying those results, the mean and variance of $\exp\left(\frac{-(\ln 10)\zeta_{\mathrm{RSS},i}}{5\alpha_i}\right)$ are computed as:

$$\mathrm{E}\left(\exp\left(\frac{-(\ln 10)\zeta_{\mathrm{RSS},i}}{5\alpha_i}\right)\right) = \exp\left(2\ln d_i + \frac{(\ln 10)^2\sigma^2_{\mathrm{RSS},i}}{50\alpha_i^2}\right) \tag{B.2}$$

and

$$\mathrm{var}\left(\exp\left(\frac{-(\ln 10)\zeta_{\mathrm{RSS},i}}{5\alpha_i}\right)\right) = \left(\exp\left(\frac{(\ln 10)^2}{25\alpha_i^2}\sigma^2_{\mathrm{RSS},i}\right) - 1\right)\exp\left(4\ln d_i + \frac{(\ln 10)^2}{25\alpha_i^2}\sigma^2_{\mathrm{RSS},i}\right) \tag{B.3}$$

Thus, from (B.2), we get

$$\mathrm{E}\left(\exp\left(\frac{-(\ln 10)\zeta_{\mathrm{RSS},i}}{5\alpha_i}\right)\right) = d_i^2\exp\left(\frac{(\ln 10)^2\sigma^2_{\mathrm{RSS},i}}{50\alpha_i^2}\right) \tag{B.4}$$

Consequently, expected value of $d_i^2$ is

$$\widehat{d_i^2} = \mathrm{E}(d_i^2) = \exp\left(\frac{-(\ln 10)\zeta_{\mathrm{RSS},i}}{5\alpha_i} - \frac{(\ln 10)^2\sigma_{\mathrm{RSS},i}^2}{50\alpha_i^2}\right) \tag{B.5}$$

## B.2 Estimation of var$(d_i^2)$

Furthermore, from (B.3), we have:

$$\mathrm{var}\left(\exp\left(\frac{-(\ln 10)\zeta_{\mathrm{RSS},i}}{5\alpha_i}\right)\right) = d_i^4 \exp\left(\frac{(\ln 10)^2}{25\alpha_i^2}\sigma_{\mathrm{RSS},i}^2\right)\left[\exp\left(\frac{(\ln 10)^2}{25\alpha_i^2}\sigma_{\mathrm{RSS},i}^2\right) - 1\right] \tag{B.6}$$

From (B.1), we have

$$\frac{-(\ln 10)\zeta_{\mathrm{RSS},i}}{2.5\alpha_i} = 4(\ln d_i) + \frac{-(\ln 10)n_{\mathrm{RSS},i}}{2.5\alpha_i} \tag{B.7}$$

Hence, $\frac{-(\ln 10)\zeta_{\mathrm{RSS},i}}{2.5\alpha_i}$ has the mean value of $4(\ln d_i)$ and the variance of $\frac{(\ln 10)^2}{6.25\alpha_i^2}\sigma_{\mathrm{RSS},i}^2$

Similarly to the section B.1, we can compute the mean of $\exp\left(\frac{-(\ln 10)\zeta_{\mathrm{RSS},i}}{2.5\alpha_i}\right)$ by applying the results in [53].

$$\mathrm{E}\left(\exp\left(\frac{-(\ln 10)\zeta_{\mathrm{RSS},i}}{2.5\alpha_i}\right)\right) = \exp\left(4\ln d_i + \frac{(\ln 10)^2\sigma_{\mathrm{RSS},i}^2}{12.5\alpha_i^2}\right) = d_i^4 \exp\left(\frac{(\ln 10)^2\sigma_{\mathrm{RSS},i}^2}{12.5\alpha_i^2}\right) \tag{B.8}$$

Thus, the estimated value of $d_i^4$ is

$$\widehat{d_i^4} = E(d_i^4) = \exp\left(\frac{-(\ln 10)\zeta_{\mathrm{RSS},i}}{2.5\alpha_i} - \frac{(\ln 10)^2\sigma_{\mathrm{RSS},i}^2}{12.5\alpha_i^2}\right) \tag{B.9}$$

(B.5) gives us the unbiased estimate of $d_i^2$, so its variance can be computed by

$$\mathrm{var}(d_i^2) = \mathrm{var}\left(\exp\left(\frac{-(\ln 10)\zeta_{\mathrm{RSS},i}}{5\alpha_i}\right)\right)\exp\left(\frac{-(\ln 10)^2\sigma_{\mathrm{RSS},i}^2}{25\alpha_i^2}\right) \tag{B.10}$$

Using (B.6), we have

$$\begin{aligned}\mathrm{var}(d_i^2) &= d_i^4 \exp\left(\frac{(\ln 10)^2}{25\alpha_i^2}\sigma_{\mathrm{RSS},i}^2\right)\left[\exp\left(\frac{(\ln 10)^2}{25\alpha_i^2}\sigma_{\mathrm{RSS},i}^2\right) - 1\right]\exp\left(\frac{-(\ln 10)^2\sigma_{\mathrm{RSS},i}^2}{25\alpha_i^2}\right) \\ &= d_i^4\left[\exp\left(\frac{(\ln 10)^2}{25\alpha_i^2}\sigma_{\mathrm{RSS},i}^2\right) - 1\right]\end{aligned} \tag{B.11}$$

Since the expected value of $d_i^4$ is given in (B.9), we have the equation of $\mathrm{var}(d_i^2)$:

$$\mathrm{var}(d_i^2) = \exp\left(\frac{-(\ln 10)\zeta_{\mathrm{RSS},i}}{2.5\alpha_i} - \frac{(\ln 10)^2\sigma_{\mathrm{RSS},i}^2}{12.5\alpha_i^2}\right)\left[\exp\left(\frac{(\ln 10)^2}{25\alpha_i^2}\sigma_{\mathrm{RSS},i}^2\right) - 1\right] \quad \text{(B.12)}$$

# Appendix C

# Mathematical derivation of equation (5.5)

Referring to 5.5, we have

$$(x_i - x)\tan(\varphi_i + \theta) = y_i - y \tag{C.1}$$

According to a trigonometric identity, it becomes

$$\frac{\tan\varphi_i + \tan\theta}{1 - \tan\varphi_i \tan\theta}(x_i - x) = y_i - y \tag{C.2}$$

$$(\tan\varphi_i + \tan\theta)(x_i - x) = (y_i - y)(1 - \tan\varphi_i \tan\theta) \tag{C.3}$$

$$x_i \tan\varphi_i + x_i \tan\theta - x\tan\varphi_i - x\tan\theta = y_i - y - y_i \tan\varphi_i \tan\theta + y\tan\varphi_i \tan\theta \tag{C.4}$$

$$\tan\varphi_i(x + y\tan\theta) - (y - x\tan\theta) - (x_i + y_i \tan\varphi_i)\tan\theta = x_i \tan\varphi_i - y_i \tag{C.5}$$

# Appendix D

# Expected value and Variance of $\gamma_{i,j}$

In [54], it is proved that if $x \sim \mathcal{N}(x_0, \varsigma^2)$ then
$\mathrm{E}(\sin x) = e^{-\varsigma^2/2} \sin x_0$
$\mathrm{E}(\cos x) = e^{-\varsigma^2/2} \cos x_0$
$\mathrm{var}(\sin x) = \mathrm{var}(\cos x) = \frac{1}{2}\left(1 - e^{-2\varsigma^2}\right)$
$\mathrm{E}(\sin^2 x) = \frac{1}{2} - \frac{1}{2}e^{-2\varsigma^2} + e^{-2\varsigma^2} \sin^2 x_0$
$\mathrm{E}(\cos^2 x) = \frac{1}{2} - \frac{1}{2}e^{-2\varsigma^2} + e^{-2\varsigma^2} \cos^2 x_0$

## D.1 Expected value of $\gamma_{i,j}$

(6.33) shows the equation of $\gamma_{i,j} = \cos\beta_{i,j}$ in terms of the related DoAs: $\varphi_i, \varphi_j, \theta_i, \theta_j$. Therefore, the estimated value of the $\gamma_{i,j}$ is:

$$\begin{aligned}\hat{\gamma}_{i,j} = \mathrm{E}(\gamma_{i,j}) = \mathrm{E}(\cos\beta_{i,j}) = \left(e^{-\mu_i^2/2}\cos\theta_i\right)\left(e^{-\mu_j^2/2}\cos\theta_j\right)\left(e^{-\left(\nu_i^2+\nu_j^2\right)/2}\cos(\varphi_i - \varphi_j)\right) + \left(e^{-\mu_i^2/2}\sin\theta_i\right)\left(e^{-\mu_j^2/2}\sin\theta_j\right) \\ = e^{-\left(\mu_i^2+\mu_j^2\right)/2}\left(\cos\theta_i\cos\theta_j\right)e^{-\left(\nu_i^2+\nu_j^2\right)/2}\cos\left(\varphi_i - \varphi_j\right) + e^{-(\mu_i^2+\mu_j^2)/2}\sin\theta_i\sin\theta_j\end{aligned} \tag{D.1}$$

## D.2 Variance of $\gamma_{i,j}$

With $i \neq j$, the variance of $\gamma_{i,j}$ is:

$$s_{i,j}^2 = \mathrm{var}(\gamma_{i,j}) = \mathrm{var}(\cos\beta_{i,j}) = \mathrm{E}(\cos^2\beta_{i,j}) - (\mathrm{E}(\cos\beta_{i,j}))^2 \tag{D.2}$$

where $\mathrm{E}(\cos\beta_{i,j})$ is expressed in (D.1) and

$$\begin{aligned}\cos^2\beta_{i,j} = (\cos\theta_i\cos\theta_j\cos(\varphi_i - \varphi_j) + \sin\theta_i\sin\theta_j)^2 \\ = \cos^2\theta_i\cos^2\theta_j\cos^2(\varphi_i - \varphi_j) + \sin^2\theta_i\sin^2\theta_j + \tfrac{1}{4}\sin(2\theta_i)\sin(2\theta_j)\cos(\varphi_i - \varphi_j)\end{aligned} \tag{D.3}$$

As a result,

$$\mathrm{E}(\cos\beta_{i,j}^2) = h_1h_2h_3 + h_4h_5 + \frac{1}{4}h_6h_7h_8 \tag{D.4}$$

where

$$h_1 = \mathrm{E}(\cos^2\theta_i) = \frac{1}{2} - \frac{1}{2}e^{-2\mu_i^2} + e^{-2\mu_i^2}\cos^2\theta_i \; ; \quad h_2 = \mathrm{E}(\cos^2\theta_j) = \frac{1}{2} - \frac{1}{2}e^{-2\mu_j^2} + e^{-2\mu_j^2}\cos^2\theta_j \; ;$$

$$h_3 = \mathrm{E}(\cos^2(\varphi_i - \varphi_j)) = \frac{1}{2} - \frac{1}{2}e^{-2\left(\nu_i^2+\nu_j^2\right)} + e^{-2\left(\nu_i^2+\nu_j^2\right)}\cos^2(\varphi_i - \varphi_j) \; ;$$

$$h_4 = \mathrm{E}(\sin^2\theta_i) = \frac{1}{2} - \frac{1}{2}e^{-2\mu_i^2} + e^{-2\mu_i^2}\sin^2\theta_i \; ;$$

$$h_5 = \mathrm{E}(\sin^2\theta_j) = \frac{1}{2} - \frac{1}{2}e^{-2\mu_j^2} + e^{-2\mu_j^2}\sin^2\theta_j \; ; \quad h_6 = \mathrm{E}(\sin(2\theta_i)) = e^{-2\mu_i^2}\sin(2\theta_i) \; ;$$

$$h_7 = \mathrm{E}(\sin(2\theta_j)) = e^{-2\mu_j^2}\sin(2\theta_j) \; ; \quad h_8 = \mathrm{E}(\cos(\varphi_i - \varphi_j)) = e^{-\left(\nu_i^2+\nu_j^2\right)/2}\cos(\varphi_i - \varphi_j) \; ;$$

## D.3 Covariance of $\gamma_{i,j}$ and $\gamma_{i,l}$

With $i$, $j$ and $l$ are different one by one, the covariance of $\gamma_{i,j}$ and $\gamma_{i,l}$ is

$$s_{i,j,l}^2 = \mathrm{cov}(\gamma_{i,j}, \gamma_{i,l}) = \mathrm{E}(\gamma_{i,j}\gamma_{i,l}) - \mathrm{E}(\gamma_{i,j})\mathrm{E}(\gamma_{i,l}) \tag{D.5}$$

where $\mathrm{E}(\gamma_{i,j})$ is expressed in (D.1) and

$$\begin{aligned}\gamma_{i,j}\gamma_{i,l} &= (\cos\theta_i\cos\theta_j\cos(\varphi_i - \varphi_j) + \sin\theta_i\sin\theta_j)(\cos\theta_i\cos\theta_l\cos(\varphi_i - \varphi_l) + \sin\theta_i\sin\theta_l) \\ &= \tfrac{1}{2}\cos^2\theta_i\cos\theta_j\cos\theta_l(\cos(2\varphi_i - \varphi_j + \varphi_l) + \cos(\varphi_j - \varphi_l)) + \tfrac{1}{2}\sin(2\theta_i)\sin\theta_j\cos\theta_l\cos(\varphi_i - \varphi_l) \\ &+ \tfrac{1}{2}\sin(2\theta_i)\cos\theta_j\sin\theta_l\cos(\varphi_i - \varphi_j) + \sin^2\theta_i\sin\theta_j\sin\theta_l\end{aligned} \tag{D.6}$$

As a result,

$$\mathrm{E}(\gamma_{i,j}\gamma_{i,l}) = \frac{1}{2}m_1m_2m_3(m_4 + m_5) + \frac{1}{2}m_6m_7m_8m_9 + \frac{1}{2}m_{10}m_{11}m_{12}m_{13} + m_{14}m_{15}m_{16} \tag{D.7}$$

where

$$m_1 = \mathrm{E}(\cos^2\theta_i) = h_1 \; ; \quad m_2 = \mathrm{E}(\cos\theta_j) = e^{-\mu_j^2/2}\cos\theta_j \; ; \quad m_3 = \mathrm{E}(\cos\theta_l) = e^{-\mu_l^2/2}\cos\theta_l \; ;$$

$$m_4 = \mathrm{E}(\cos(2\varphi_i - \varphi_j + \varphi_l)) = e^{-\left(4\nu_i^2+\nu_j^2+\nu_l^2\right)/2}\cos(2\varphi_i - \varphi_j + \varphi_l) \; ;$$

$$m_5 = \mathrm{E}(\cos(\varphi_j - \varphi_l)) = e^{-\left(\nu_i^2+\nu_j^2\right)/2}\cos(\varphi_j - \varphi_l) \; ;$$

$$m_6 = \mathrm{E}(\sin(2\theta_i)) = h_6 \; ; \quad m_7 = \mathrm{E}(\cos\theta_l) = e^{-\mu_l^2/2}\cos\theta_l; \quad m_8 = \mathrm{E}(\cos\theta_l) = m_3 \; ;$$

$$m_9 = \mathrm{E}(\cos(\varphi_i - \varphi_l)) = e^{-\left(\nu_i^2+\nu_l^2\right)/2}\cos(\varphi_i - \varphi_l) \; ;$$

$$m_{10} = \mathrm{E}(\sin(2\theta_j)) = h_7 \; ; \quad m_{11} = \mathrm{E}(\cos\theta_j) = m_2 \; ; \quad m_{12} = \mathrm{E}(\sin\theta_l) = e^{-\mu_l^2/2}\sin\theta_l \; ;$$

$$m_{13} = \mathrm{E}(\cos(\varphi_i - \varphi_j)) = h_8;$$

$$m_{14} = \mathrm{E}(\sin^2\theta_i) = h_4 \; ; \quad m_{15} = \mathrm{E}(\sin\theta_j) = e^{-\mu_j^2/2}\sin\theta_j; \quad m_{16} = \mathrm{E}(\sin\theta_l) = m_{12}$$

# Bibliography


[1] C. Mensing, A. Dammann, and S. Sand, *Positioning in Wireless Communication Systems*. Wiley, 2014, ch. 1, pp. 1–2.

[2] R. Zekavat and R. M. Buehrer, *Handbook of Position Location: Theory, Practice, and Advances*. Wiley, 2019, ch. 1, pp. 5–9.

[3] M. Vossiek, L. Wiebking, P. Gulden, J. Wieghardt, and C. H. P. Heide, "Wireless local positioning," *IEEE Microwave Magazine*, vol. 4, 2003.

[4] M. Jamalabdollahi and S. R. Zekavat, "High resolution toa estimation via optimal waveform design," *IEEE Transactions on Communications*, vol. 65, 2017.

[5] M. Jamalabdollahi and S. Zekavat, "Range measurements in non-homogenous, time and frequency dispersive channels via time and direction of arrival merger„" *IEEE Trans. Geosci. Remote Sens.,*, vol. 55, no. 2, pp. 742–752, 2017.

[6] T. Rappaport, J. Reed, and B. Woerner, "Position location using wireless communications on highways of the future," *IEEE Communications Magazine*, 1996.

[7] M. Laoufi, M. Heddebaut, M. Cuvelier, J. Rioult, and J. Rouvaen, "Positioning emergency calls along roads and motorways using a gsm dedicated cellular radio network," *VTC*, 2000.

[8] B. Zhu, J. Cheng, Y. Wang, J. Yan, and J. Wang, "Three-dimensional VLC positioning based on angle difference of arrival with arbitrary tilting angle of receiver," *EEE Journal on Selected Areas in Communications*, vol. 36, no. 1, 2018.

[9] C. Mensing, A. Dammann, and S. Sand, *Positioning in Wireless Communication Systems*. Wiley, 2014, ch. 4, pp. 71–78.

[10] ——, *Positioning in Wireless Communication Systems*. Wiley, 2014, ch. 1, p. 22.

[11] X. Li, "Rss-based location estimation with unknown pathloss model," *IEEE Transactions on Wireless Communications*, vol. 5, 2006.

[12] J. Shirahama and T. Ohtsuki, "Rss-based localization in environments with different path loss exponent for each link," *VTC Spring - IEEE Vehicular Technology Conference*, 2008.

[13] Y. T. Chan, B. H. Lee, R. Inkol, and F. Chan, "Received signal strength localization with an unknown path loss exponent," *24th Canadian Conference on Electrical and Computer Engineering(CCECE)*, 2011.

[14] N. Salman, M. Ghogho, and A. H. Kemp, "On the joint estimation of the rss-based location and path-loss exponent," *IEEE Wireless Communications Letters*, vol. 1, 2012.

[15] N. S. Kodippili and D. Dias, "Integration of fingerprinting and trilateration techniques for improved indoor localization," *Seventh International Conference on Wireless and Optical Communications Networks - (WOCN)*, 2010.

[16] J.-L. Chen, M.-C. Chen, T.-L. Chiang, Y.-C. Chang, and F. Y. Shih, "Distributed fermat-point location estimation for wireless sensor network applications," *IEEE Sarnoff Symposium*, 2007.

[17] M. M. Zaniani, A. M. Shahar, and I. A. Azid, "Trilateration target estimation improvement using new error correction algorithm," *18th Iranian Conference on Electrical Engineering*, 2010.

[18] F. Thomas and L. Ros, "Revisiting trilateration for robot localization," *IEEE Transactions on Robotics*, vol. 21, 2005.

[19] R. A. Johnson, "Advanced euclidean geometry: An elementary treatise on the geometry of the triangle and the circle," 1960.

[20] F. Chan, H. So, J. Zheng, and K. Lui, "Best linear unbiased estimator approach for time-of-arrival based localisation," *IET Signal Proc.*, 2008.

[21] Y. L. Tong, *Fundamental Properties and Sampling Distributions of the Multivariate Normal Distribution.* Ed.Springer, 1989.

[22] A. Bjorck, *Numerical Methods for Least Squares Problems.* SIAM (Society for Industrial and Applied Mathematics), 1996.

[23] D. Wang, G. Hosangadi, P. Monogioudis, and A. Rao, "Mobile device localization in 5G wireless networks," *International Conference on Computing, Networking and Communications (ICNC)*, 2019.

[24] D. Munoz, C. Vargas, R. Enriquez-Caldera, and F. B. Lara, *Position Location Techniques and Applications*, 2009.

[25] S. Wong, R. Jassemi-Zargani, D. Brookes, and B. Kim, "A geometric approach to passive target localization," *S&T Organization*, 2017.

[26] Y. T. Chan and K. C. Ho, "A simple and efficient estimator for hyperbolic location," *IEEE Transactions on Signal Processing*, vol. 42, no. 8, pp. 1905–1915,, 1994.

[27] Y. Huang, J. Benesty, G. W. Elko, and R. M. Mersereau, "Real-time passive source localization: a practical linear-correction least-squares approach," *IEEE Transactions on Speech and Audio Processing*, vol. 9, no. 8, pp. 943–956, 2001.

[28] K. W. Cheung, H. C. So, W. K. Ma, and Y. T. Chan, "A constrained least squares approach to mobile positioning: algorithms and optimality," *Hindawi Publishing Corporation EURASIP Journal on Applied Signal Processing*, 2006.

[29] K. Yang and J. An, "Constrained total least-squares location algorithm using time-difference-of-arrival measurements," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 3, 2010.

[30] S. Dwivedi, R. Shreevastav, F. Munier, J. Nygren, I. Siomina, Y. Lyazidi, D. Shrestha, G. Lindmark, P. Ernström, E. Stare, S. M. Razavi, S. Muruganathan, G. Masini, Åke Busin, and F. Gunnarsson, "Positioning in 5G networks," *arXiv preprint arXiv:2102.03361*, 2021.

[31] A. Omri, M. Shaqfeh, A. Ali, and H. Alnuweiri, "Synchronization procedure in 5G NR systems," *IEEE Access*, vol. 7, 2019.

[32] D. L. Vandegraft, "A boundary delineation system for the bureau of ocean energy management," *OCEANS 2017 - Anchorage*, 2017.

[33] T. L. N. Nguyen and Y. Shin, "A new approach for positioning based on AOA measurements," *International Conference on Computing, Management and Telecommunications (ComManTel)*, 2013.

[34] C.-S. Chen, Y.-J. Chiu, J.-M. Lin, and C.-H. Lee, "Geometrical positioning schemes for MS location estimation," *International Symposium on Computer, Consumer and Control*, 2012.

[35] J. Xu, M. Ma, and C. L. Law, "AOA cooperative position localization," *GLOBECOM*, 2008.

[36] Y. Wang, K. Ho, and G. Wang, "A unified estimator for source positioning and DOA estimation using AOA," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018.

[37] A. Chan and J. Litva, "MUSIC and maximum likelihood techniques on two-dimensional DOA estimation with uniform circular array," *IEE Proc, Radar Sonar Navig*, vol. 142, no. 3, pp. 105–114, 1995.

[38] R. Yerukala and N. K. Boiroju, "Approximations to standard normal distribution function," *International Journal of Scientific & Engineering Research*, vol. 6, 2015.

[39] Y. Wang and K. C. Ho, "An asymptotically efficient estimator in closed-form for 3-D AOA localization using a sensor network," *IEEE Transactions on Wireless Communications*, vol. 14, 2015.

[40] A. I. Oweis, S. A. Alawsh, A. H. Muqaibel, and M. S. Sharawi, "A coprime array of patch antennas for doa estimation in mobile handheld devices," *IEEE International Symposium on Antennas and Propagation & USNC/URSI National Radio Science Meeting*, 2018.

[41] S. Sun, H. Li, and J. Xiong, "Direction of arrival estimation using compact MIMO array for portable devices," *IEEE International Symposium on Antennas and Propagation & USNC/URSI National Radio Science Meeting*, 2017.

[42] G. Li, X. Bao, and Z. Wang, "The design and implementation of a smartphone-based acoustic array system for DOA estimation," *36th Chinese Control Conference (CCC)*, 2017.

[43] H. Li, S. Sun, and J. Wang, "Direction of arrival estimation using amplitude and phase information in low-profile MIMO arrays," *IEEE Transactions on Antennas and Propagation*, vol. 66, 2018.

[44] H.-J. Shao, X.-P. Zhang, and Z. Wang, "Efficient closed-form algorithms for AOA based self-localization of sensor nodes using auxiliary variables," *IEEE Transactions on Signal Processing*, vol. 62, 2014.

[45] Y.-C. Cheng, J.-Y. Lin, C.-W. Yi, Y.-C. Tseng, L.-C. Kuo, Y.-J. Yeh, and C.-W. Lin, "AR-based positioning for mobile devices," *40th International Conference on Parallel Processing Workshops*, 2011.

[46] D. Niculescu and B. Nath, "Ad hoc positioning system (APS) using AOA," *IEEE INFOCOM*, 2003.

[47] P. Stoica and A. Nehorai, "MUSIC, maximum likelihood, and cramer-rao bound," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, 1989.

[48] D. Kim, J. K. Park, and J. T. Kim, "Three-dimensional VLC positioning system model and method considering receiver tilt," *IEEE Access*, vol. 7, 2019.

[49] M. Zhou, X. Zhang, X. Qiu, and C. Wang, "Two-dimensional doa estimation for uniform rectangular array using reduced-dimension propagator method," *International Journal of Antennas and Propagation*, vol. 2015, 2015.

[50] S. Katsuki and N. Sebe, "Rotation matrix optimization with quaternion," *10th Asian Control Conference (ASCC)*, 2015.

[51] B. Zhu, J. Cheng, J. Yan, J. Wang, and Y. Wang, "VLC positioning using cameras with unknown tilting angles," *IEEE Global Communications Conference*, 2017.

[52] Z. Ma and K. Ho, "TOA localization in the presence of random sensor position errors," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2011.

[53] Y. Viniotis, *Probability and Random Processes for Electrical Engineers.* New York: McGraw-Hill, 1998.

[54] J. Gray, "An exact determination of the probability density function under coordinate transformations," *The First IEEE Regional Conference on Aerospace Control Systems,*, 1993.