FAKULTA
APLIKOVANÝCH VĚD
ZÁPADOČESKÉ
UNIVERZITY
V PLZNI

Quality assessment of manoeuvering object tracking by radar multi-lateration

Diploma thesis

Author: Bc. Marek Siřiště
Supervisor: doc. Ing. Ondřej Straka, Ph.D.
Consultant: Ing. Jan Krejčí

Pilsen, 2024

Rozsah diplomové práce: **40-50 stránek A4**
Rozsah grafických prací:
Forma zpracování diplomové práce: **tištěná/elektronická**
Jazyk zpracování: **Angličtina**

Seznam doporučené literatury:

1. Paul D. Groves, "Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems", second edition, Artech House, 2013
2. Yaakov Bar-Shalom and Xiao-Rong Li, "Estimation and Tracking: Principles, Techniques, and Software", Artech House, 1993

Vedoucí diplomové práce: **Doc. Ing. Ondřej Straka, Ph.D.**
Katedra kybernetiky

Datum zadání diplomové práce: **2. října 2023**
Termín odevzdání diplomové práce: **20. května 2024**

Západočeská univerzita v Plzni
L.S.
Fakulta aplikovaných věd

Doc. Ing. Miloš Železný, Ph.D.
děkan

Doc. Dr. Ing. Vlasta Radová
vedoucí katedry

V Plzni dne 2. října 2023

ZÁPADOČESKÁ UNIVERZITA V PLZNI
Fakulta aplikovaných věd
Akademický rok: 2023/2024

ZADÁNÍ DIPLOMOVÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení:	**Bc. Marek SIŘIŠTĚ**
Osobní číslo:	**A22N0100P**
Studijní program:	**N0714A150011 Kybernetika a řídicí technika**
Specializace:	**Automatické řízení a robotika**
Téma práce:	**Vyhodnocení kvality sledování manévrujícího objektu radarovou multi-laterací**
Zadávající katedra:	**Katedra kybernetiky**

Zásady pro vypracování

1. Seznamte se s problémem sledování manévrujícího objektu s využitím radarové multi-laterace.
2. Implementujte algoritmus sledování s možnostmi:

- volby poloh radarových přijímačů, jejich dostupnosti a charakteru měření (TDOA, AoA),
- množiny možných modelů pohybu objektu.

3. Vyhodnoťte kvalitu sledování objektu za pomoci Monte Carlo simulace a proveďte zobrazení výsledku ve formě mapy.

Čestné prohlášení

Předkládám tímto k posouzení a obhajobě diplomovou práci zpracovanou na závěr studia na Fakultě aplikovaných věd Západočeské univerzity v Plzni.

Prohlašuji, že jsem diplomovou práci vypracoval samostatně a výhradně s použitím odborné literatury a pramenů, jejichž úplný seznam je její součástí.

V Plzni dne .....

Abstrakt

Cílem této diplomové práce je analýza kvality odhadu polohy objektů v předem definované mapě. Pro tuto analýzu byl implementován algoritmus pro sledování objektů s využitím radarové multilaterace, který umožňuje modulární přístup ke konfiguraci jednotlivých přijímačů z hlediska jejich počtu, rozmístění a dostupnosti měření. Algoritmus využívá čtyři dynamické pohybové modely pro účely odhadu polohy. Přesnost poskytnutých odhadů je vyhodnocena pomocí Monte Carlo simulací, a údaje o přesnosti jsou následně prezentovány prostřednictvím mapy kvality odhadu. Výsledná simulace poskytuje celkový obraz o kvalitě odhadu v rámci definovaného prostoru v závislosti na různých podmínkách.

Klíčová slova: odhad stavu, Kalmanův filtr, rozšířený Kalmanův filtr, interakce mezi více modely, dynamické modely pohybu, měření vzdálenosti, čas příchodu signálů, časový rozdíl příchodu signálů, úhel příchodu signálů

Abstract

The aim of this thesis is to analyze the quality of object position estimation in a predefined map. For this analysis, an algorithm for object tracking using radar multilateration has been implemented, which allows a modular approach to the configuration of individual receivers in terms of their number, placement and availability of measurements. The algorithm uses four dynamic motion models for position estimation purposes. The accuracy of the provided estimates is evaluated using Monte Carlo simulations, and the accuracy is then presented via an estimation quality map. The resulting simulation provides an overall picture of the estimation quality within the defined space depending on various conditions.

Key words: state estimation, Kalman filter, extended Kalman filter, Interacting Multiple Models, dynamic motion models, ranging, time of arrival, time difference of arrival, angle of arrival

Poděkování

Rád bych poděkoval vedoucímu mé diplomové práce panu doc. Ing. Ondřeji Strakovi, Ph.D za odborné rady, veškerou pomoc, vřelý přístup a poskytnutý čas. Dále bych velice rád poděkoval svému konzultantovi panu Ing. Janu krejčímu za poskytnutou pomoc, odborné rady, čas a taktéž za podporu, kterou mi během práce poskytl. V neposlední řadě bych také velice rád poděkoval panu Ing. Pavlovi Kulmonovi, Ph.D za poskytnutí příležitosti a taktéž za odborné konzultace.

Contents

1 Introduction **1**
- 1.1 Motivation . . . 1
- 1.2 Thesis structure . . . 2

2 Ranging **4**
- 2.1 Ranging Techniques: Methods and Applications . . . 4
  - 2.1.1 Geometric principles of positioning by ranging . . . 4
  - 2.1.2 Passive ranging - time of arrival (ToA) . . . 6
  - 2.1.3 Positioning from Time Difference Of Arrival . . . 6
  - 2.1.4 Positioning from Angle of Arrival . . . 8
  - 2.1.5 Transformation of TDoA and AoA measurements . . . 9

3 Dynamic Object Motion Models **13**
- 3.1 Dynamic models in navigation . . . 13
  - 3.1.1 Nearly constant velocity . . . 13
  - 3.1.2 Singer acceleration model . . . 14
  - 3.1.3 Discrete white noise acceleration model . . . 15
  - 3.1.4 Discrete Wiener process acceleration model . . . 16
- 3.2 Modeling for several spatial dimentions . . . 17
- 3.3 Observation model . . . 18

4 State Estimation **19**
- 4.1 Kalman filter . . . 19
  - 4.1.1 KF Algorithm . . . 20
- 4.2 Extended Kalman filter . . . 21
- 4.3 Multiple model approach . . . 22
  - 4.3.1 Formulation . . . 22
  - 4.3.2 The Multiple Model Approach for fixed modes . . . 22
  - 4.3.3 The Multiple Model Approach for switching modes . . . 24
- 4.4 The Interacting Multiple Model . . . 25
  - 4.4.1 Algorithm description . . . 28

5 Architectural Overview of the Tracking Assessement Framework **31**
- 5.1 Configuration module . . . 32
- 5.2 Simulations of true values and measurements . . . 35
- 5.3 Estimation framework . . . 36
  - 5.3.1 Estimation using the IMM algorithm . . . 37
- 5.4 Visualization module . . . 39

6 Simulation Experiments **44**
6.1 Part I: Frequent measurements 47
6.2 Part II: Scarse measurements 51
6.3 Experiment conclusion 52

7 Conclusion **53**

References **55**

List of abbreviations

- **AoA** - Angle of Arrival
- **ANEES** - Average Normalized Estimation Error Squared
- **DWNA** - Discrete White Noise Acceleration
- **DWPA** - Discrete Wiener Process Acceleration
- **HMM** - Hidden Markov Model
- **IMM** - Interacting Multiple Model
- **KF** - Kalman Filter
- **LoP** - Line of Position
- **LoS** - Line Of Sight
- **MMA** - Multiple Model Approach
- **NCV** - Nearly Constant Velocity
- **PDF** - Probability Density Function
- **RMSE** - Root Mean Square Error
- **TDoA** - Time Difference of Arrival
- **ToA** - Time of Arrival
- **ToF** - Time of Flight
- **UTC** - Coordinated Universal Time

List of Figures

1 Object detection using radar multilateration 1
2 Block diagram of software implementation structure 2
3 Line of position for a single receiver 5
4 Locus of emitter's position for two receivers 5
5 Hyperbolic LoP from TDoA measurements in 2D using three receivers . . . 7
6 AOA - azimuth and elevation 9
7 Comparison of converted TDoA/AoA measurements with true emitter positions in 3D 11
8 Mode probability update 23
9 IMM - interaction step [17] 27
10 IMM - Filtering step [17] 27
11 IMM - Combination step [17] 27
12 Architecture of code 31
13 Star layout of the receivers 34
14 Example evaluation map 34
15 Generated trajectory using HMM with Singer and DWPA models 35
16 Active model over time 36
17 IMM algorithm estimation 38
18 Visualization of the estimation quality within the evaluation map 40
19 Layout of the receivers for the specific example 40
20 Visualization of the estimate quality for one trajectory 41
21 Visualization of the poor estimation quality for one trajectory 41
22 Visualization of estimation quality in 2D for specific layer 42
23 Visualization of estimate quality in 2D for specific layers of z and y 42
24 RMSE metric for each axis for the example trajectory 43
25 ANEES metric for the example trajectory 43
26 Experiment: layout of receivers 45
27 Scenario 1: KF 47
28 Scenario 4: EKF 47
29 NCV and DWNA models - scenario 1 and 4 (three TDoA, no AoA) 47
30 Scenario 2: KF 48
31 Scenario 5: EKF 48
32 NCV and DWNA models - scenario 2 and 5 (three TDoA, four AoA) . . . 48
33 Scenario 3: KF 48
34 Scenario 6: EKF 48
35 NCV and DWNA models - scenario 3 and 6 (two TDoA, three AoA) . . . 48
36 Scenario 1 and 4: the quality across all z-levels 48

37 Scenario 2 and 5: the quality across all z-levels 49
38 Scenario 3 and 6: the quality across all z-levels 49
39 Scenario 1: KF 50
40 Scenario 4: EKF 50
41 Singer and DWPA models - scenario 3 and 6: (three TDoA, no AoA) . . . 50
42 Scenario 1 and 4: the quality across all z-levels 50
43 Scenario 1: KF 51
44 Scenario 4: EKF 51
45 NCV and DWNA models - scenario 1 and 4 (every tenth measurement is used) 51
46 Scenario 1 and 4: the quality across all z-levels (every tenth measurement is used) 51

1 Introduction

Accurate estimation of aircraft trajectories is essential for the coordination of aircraft movements at airports and aerospace, which is a critical aspect for safety. Radar sensors, which are commonly used as data source for estimation, are noisy and their processing (i.e., the estimation) relies on inaccurate motion models. This may lead to compromised estimation accuracy, calling for further investigation. The main objective of this work is to develop a methodology and software capable of the assessing accuracy of aircraft position estimation depending on the chosen model configuration and data processing approach [1].

1.1 Motivation

The motivation for this thesis is to assess the accuracy and reliability of object position estimates depending on several conditions. Multilateration, a method that determines an object position by utilizing range measurements such as angle of arrival (AoA), time of arrival (ToA) and time difference of arrival (TDoA) from multiple receivers with known positions, is illustrated in Figure 1. The reliability of multilateration can be compromised for objects that perform manoeuvres due to the dynamic nature of their motions and measurement uncertainties. [2].

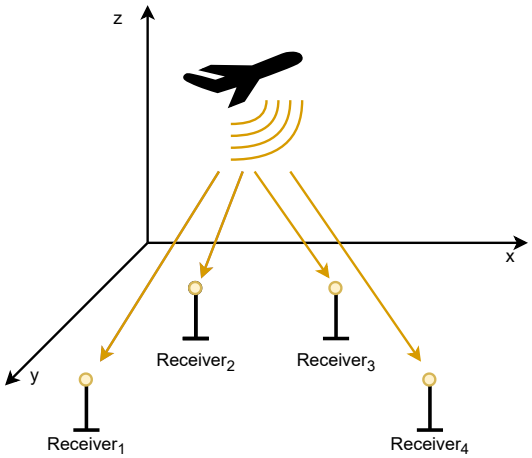


Figure 1: *Object detection using radar multilateration*

Multilateration techniques are increasingly utilized in airport surveillance systems to enhance the monitoring of object movements. The multilateration sensors provide precise positioniong data to complete the capabilities of primarily used tracking systems. Multilateration system has the potential to provide the same or even better results at a reduced cost compared to traditional surveillance radar systems [3]. This potential is particularly important in challenging areas where it is crucial to ensure high estimation accuracy. Furthermore, receivers layouts and availability of measurements can enhance the overall efficiency of tracking systems. The detailed analysis of estimation quality within a predefined space under specific conditions can lead to strategic decision on the placement of receivers. Such strategic placement ensures that the system provides the most accurate and reliable estimate of object locations.

1.2 Thesis structure

This thesis deals with the problem of tracking maneuvering objects using radar multilateration. It is used to estimate the position of individual objects moving in the area of, e.g., an airport. The key component of this thesis is to implement a tracking algorithm that allows modular selection of individual receivers, in terms of their number but also their layout. In addition, it is possible to choose availability of individual measurements provided by each receiver. The algorithm utilizes a predefined set of four models for estimation purposes. The estimate accuracy is evaluated by Monte Carlo simulations and is presented through a map of estimation quality.

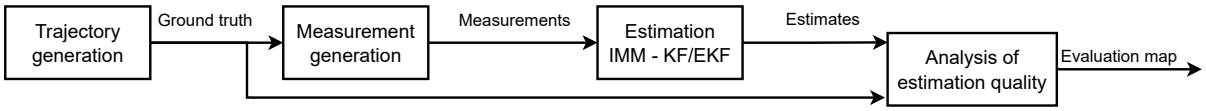


Figure 2: *Block diagram of software implementation structure*

A block diagram describing the structure of the software implementation is illustrated in Figure 2. The thesis is organised as follows: Section 2 discusses geometric principles of positioning, including explanation of the concept such as AoA, ToA and TDoA. It consists of mathematical formulas essential for modeling and simulating the measurements. This part is represented in the block diagram by the block **Measurement generation**. Subsequently, Section 3 describes motion models for tracking position and velocity, including the nearly constant velocity (NCV) and discrete white noise acceleration (DWNA) models. It also covers more comprehensive models like the Singer model and the discrete Wiener process acceleration (DWPA), which model position, velocity, and acceleration dynamics. In the block diagram, this part is included in the **Trajectory generation**, where the models are used to generate groud-truth values, and in the **Estimation** block, where the motion models are used in the IMM algorithm. Next, the principles of multiple model approach are expressed in Section 4.3 including the interacting multiple models (IMM) algorithm, which is essential for the efficient state estimation processes discussed in this thesis [4]–[6]. Later, Section 5 is devoted to the implementation. This includes generating of ground-truth values using hidden Markov model (HMM) with previously mentioned models. The ground truth is utilized to generate TDoA and AoA measurements, which are transformed into position vector for the Kalman filter (KF) or used as raw values for the extended Kalman filter (EKF). The IMM algorithm is implemented for state estimation across several scenarios, including different combination of models, receiver's layout and availability of measurements. The outcomes of the IMM algorithm are further analyzed in the evaluation part of the program (the block **Analysis of estimation quality** in the diagram), where the estimation results are compared with the true values. The results are illustrated using a map, which is colored based on the quality

of estimates allowing users to visualy assess estimation quality under the chosen conditions. The software implementation is structured with distinct, modular components, each featuring specific inputs and outputs that allows easy substitution.

Finally, simulation experiments are conducted in Section 6 to demonstrate the flexibility in selecting individual conditions for evaluating the quality of the estimation. Several scenarios under different conditions are presented, showcasing their outcomes and comparisons. This will serve as a demonstration of the analysis capabilities and plot the result on a sample case.

2 Ranging

This section covers the basic concept of ranging, including ranging techniques specifically focusing on time of flight methods such as time of arrival, time difference of arrival, and angle of arrival. Each technique's methodology, applications, and mathematical foundations are explored to provide a comprehensive understanding.

2.1 Ranging Techniques: Methods and Applications

Positioning by ranging is a process of determining the location of an object emitting a radio signal (emitter) by indirectly measuring its distance from several objects (receivers) with known positions. Positioning by ranging can also be referred to as lateration, trilateration, or multilateration depending on the number of receivers, and which signals are used for obtaining the object position. It is one of the most common method used in radio navigation.

Range is typically determined by measuring the signals's time of flight (ToF), but it can also be calculated from the received signal strength. For the purpose of this thesis, however only ToF methods will be discussed. ToF is based on the time signal takes to travel from the emitter to the receiver [7].

There are several types of ToF-based methods to obtain position of an object. For the purpose of this thesis, the following methods will be further described:

- Passive ranging – time of arrival (ToA) measurements are used.
- Hyperbolic ranging – time difference of arrival (TDoA) measurements are used.
- Angular positioning – angle of arrival (AoA) measurements are used.
- Combination of hyperbolic ranging and angular positioning.

2.1.1 Geometric principles of positioning by ranging

Before describing the individual ranging methods, it is necessary to establish geometric principles of positioning by ranging. In this section, it is assumed that the emitters and receivers are located in a 2-D plane for simplicity and the ranges i.e., distances between the emitter and the receiver, are available. In the case of a single receiver, the emitter can be positioned anywhere on the circle with the receiver at its center. The radius of the circle is equal to the distance between the emitter and receiver. The circle is called **line of position** (LoP). Generally, LoP is considered as locus of candidate positions, as shown in Figure 3, where (x_r, y_r) represent the receiver coordinates and d is the range.

To compute the geometric range, denote the emitter position as (x_e, y_e), where x_e and y_e signify the emitter coordinates within the 2-D planar coordinate frame. (x_r, y_r)

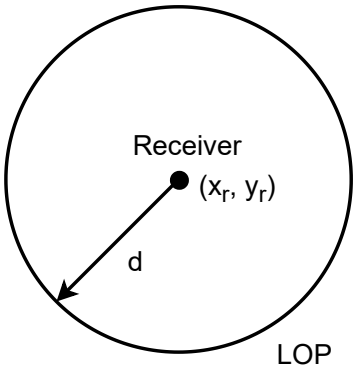


Figure 3: *Line of position for a single receiver*

represent the receiver coordinates. The subscripts r and e refer to receiver and emitter respectively. In the case of the considered 2-D setting, the coresponding geometric range can be computed as follows:

$$d_{er} = \sqrt{(x_r - x_e)^2 + (y_r - y_e)^2}, \tag{2.1}$$

Note that, the geometric range (2.1) is independent of direction, i.e., d_{er} is equal to d_{re}.

Considering a second receiver, the locus of the emitter position is defined by the intersection of the two coresponding LoPs. Such a scenario is illustrated in Figure 4, where the circles intersect at two points. That is, to satisfy the LoP constrains from both

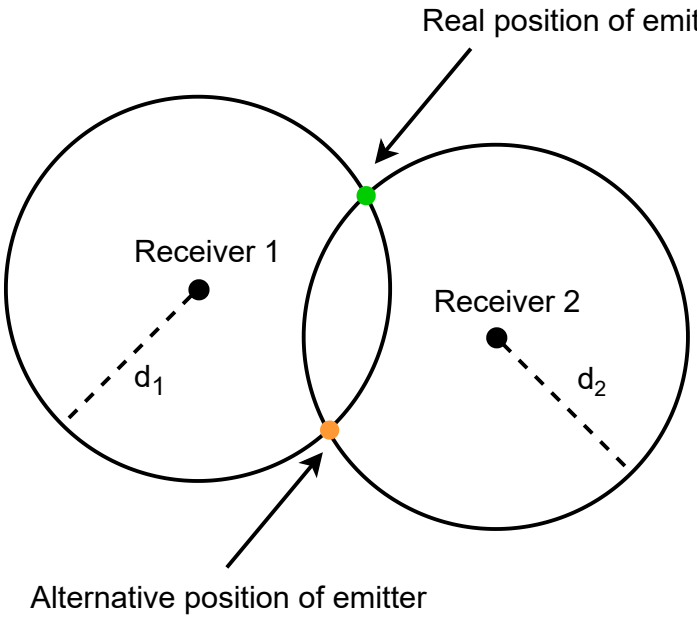


Figure 4: *Locus of emitter's position for two receivers*

receivers, the object position should lie at an intersection. This leads to ambiguity of solution, which can be resolved by obtaining measurements from a third receiver or by

considering additional information about the object.

2.1.2 Passive ranging - time of arrival (ToA)

Given the geometric aspects, the methodology of passive ranging using ToA, ie., for a single receiver, can be stated. In passive ranging, the receiver r records the time t_r of detecting a known signal that was sent from an emmiter e at a known time t_e. The time t_e is referred to as the time of emission. Consider that there are N receivers with known positions $\mathcal{X}_i = [x_i, y_i, z_i]^\intercal$, $i = 1, 2, ..., N$ and one emitter with unknown position $\mathcal{X}_e = [x_e, y_e, z_e]^\intercal$. The time t_{ri} at which a signal emmited from emitter reaches the receiver i can be expressed as:

$$t_{ri} = t_e + \frac{\|\mathcal{X}_e - \mathcal{X}_i\|}{c} + v_i, \tag{2.2}$$

where c is speed of light and v_i is the arrival time measurement noise. In this thesis, v_i is assumed to be zero-mean Gaussian process uncorrelated with other receivers for simplicity. The time of arrival (ToA) is measured by the receiver clock while the signal transmission time is determined by the emitter clock. Note, that there are four unknown values in Equation (2.2): the position of the emitter $\mathcal{X}_e = [x_e, y_e, z_e]$ and time of emission t_e. Consequently, a minimum of four ToA measurements is necessary to solve for X_e and t_e. Time of emission can be removed from (2.2) by a different approach. One of the possible solution is the positioning from time difference of arrival (TDoA) [7]–[9].

2.1.3 Positioning from Time Difference Of Arrival

In TDoA positioning, the differences in ranging measurements between each pair of receivers are used to eliminate the emission time t_e. To achieve this, receivers need to be synchronised. The formula to compute the TDoA between receivers i and j is formulated as:

$$\Delta t_{ij} = t_{r_i} - t_{r_j} = \frac{\|\mathcal{X}_e - \mathcal{X}_i\|}{c} - \frac{\|\mathcal{X}_e - \mathcal{X}_j\|}{c} + v_{ij}, \tag{2.3}$$

where noise v_{ij} is given as:

$$v_{ij} = v_i - v_j. \tag{2.4}$$

As mentioned in Section 2.1.2, noises v_i and v_j are assumed independent. The location of the emitter $\mathcal{X}_e = [x_e, y_e, z_e]$ can be now obtained by using only three TDoA measurements due to elimination of time of emission. Note, that Δt_{ij} depends on the order of differencing, ie., $\Delta t_{ij} = -\Delta t_{ji}$ (when neglecting noise) and TDoA between receivers i and

j can alternatively be computed by using the TDoA for a third receiver k:

$$\Delta t_{ij} = \Delta t_{ik} + \Delta t_{kj}. \tag{2.5}$$

Considering positioning by TDoA in 2-D, a visual representation of this concept is illustrated in Figure 5, where TDoA measurements from three receivers create hyperbolic Lines of Position (LoPs).

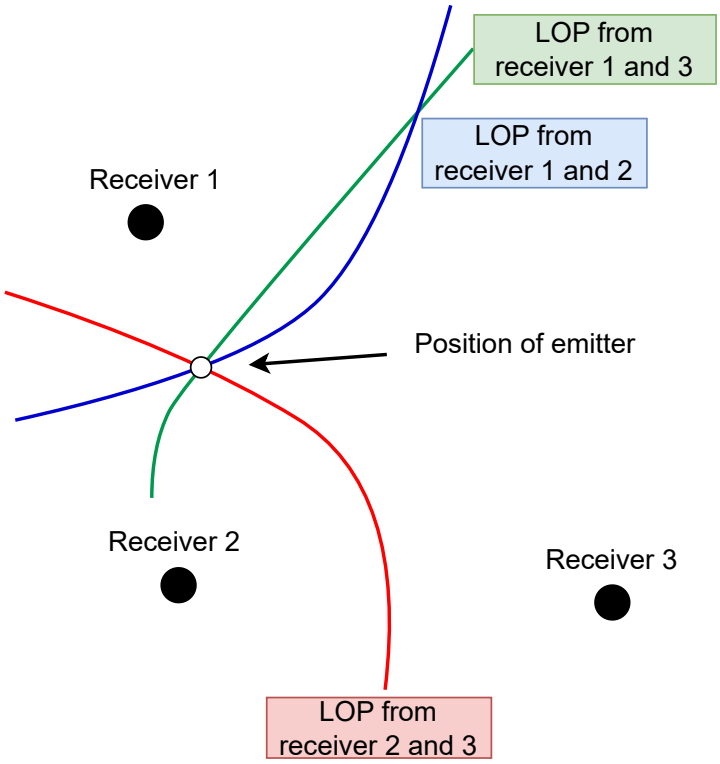


Figure 5: *Hyperbolic LoP from TDoA measurements in 2D using three receivers*

Considering N receivers, there are $N-1$ equations, which are linearly independent. For simplicity, all TDoA measurements are determined relative to a specific receiver (called reference receiver), denoted as $\mathcal{X}_1$. The TDoA between receiver i, where $i = 2, 3, ..., N$, and receiver 1 is given as follows:

$$\Delta t_{i1} = \underbrace{\frac{\|\mathcal{X}_e - \mathcal{X}_i\|}{c} - \frac{\|\mathcal{X}_e - \mathcal{X}_1\|}{c}}_{h_i(\mathcal{X}_e)} + v_{i1}. \tag{2.6}$$

Considering $N-1$ TDoA equations, the formula can be expressed as:

$$\underbrace{\begin{bmatrix}\Delta t_{21}\\ \Delta t_{31}\\ \Delta t_{41}\\ \vdots\\ \Delta t_{N1}\end{bmatrix}}_{\Delta\mathbf{t}} = \underbrace{\begin{bmatrix}h_2(\mathcal{X}_e)\\ h_3(\mathcal{X}_e)\\ h_4(\mathcal{X}_e)\\ \vdots\\ h_N(\mathcal{X}_e)\end{bmatrix}}_{\mathbf{h}(\mathcal{X})_e} + \underbrace{\begin{bmatrix}v_{21}\\ v_{31}\\ v_{41}\\ \vdots\\ v_{N1}\end{bmatrix}}_{\mathbf{v}}. \tag{2.7}$$

As mentioned above, the noise is assumed to be zero-mean Gaussian process. It means that the maximum likelihood estimate can be used for the estimation of $\mathcal{X}_e$:

$$\hat{\mathcal{X}}_e = \arg\min_{\mathcal{X}_e}\Big(\Delta\mathbf{t} - \mathbf{h}(\mathcal{X}_e)\Big)^{\mathsf{T}}\mathbf{Q}^{-1}\Big(\Delta\mathbf{t} - \mathbf{h}(\mathcal{X}_e)\Big), \tag{2.8}$$

where $\mathbf{Q} = E[\mathbf{v}\mathbf{v}^{\mathsf{T}}]$ is the measurements noise covariance matrix. Note that the measurement noise covariance matrix $\mathbf{Q}$ for TDoA is not diagonal, unlike in TOA systems. This is caused because $\mathbf{Q}$ captures covariance between the noise differencies $v_{ij} = v_i - v_j$ from individual receivers, as mentioned above. The issue lies in the challenge of discovering an algorithm that can find the global minimum of the right-hand side of the equation (2.8). The chance to find global minimum can be improved by determining a region, where the optimal solution is located. It is assumed that the emitter is placed within intersection of hyperbolas formed by the time differences of arrival from multiple receivers. The equation (2.8) is valid under the condition that $\mathcal{X}_e \in S_1 \cap S_2 \cap S_N$, where S_i indicates the area where the object is detectable by the receiver r_i [7]–[9].

2.1.4 Positioning from Angle of Arrival

In the angular positioning method, also referred to as Angle of Arrival (AoA), the position of an emitter is obtained through the directions of lines of sight (LoS) from the emitter to two or more known receivers. Each LoS constitutes a line of position. The position of emitter is located at the intersection of these lines of positions [7]. In this thesis, azimuth and elevation are used as AoAs. These two angles, as illustrated in Figure 6, define the 3-D direction from which a signal arrives at a receiver which is essential for determining the position of an emitter in 3-D space.

The azimuth is the angle between true north and the horizontal projection of the line of sight (LoS) from the emitter to each receiver [7]. The azimuth angle between receiver i and emitter can be computed using:

$$Z_i^{AoA,az} = \arctan\left(\frac{y_i - y_e}{x_i - x_e}\right) + v_i^{AoA,az}, \tag{2.9}$$

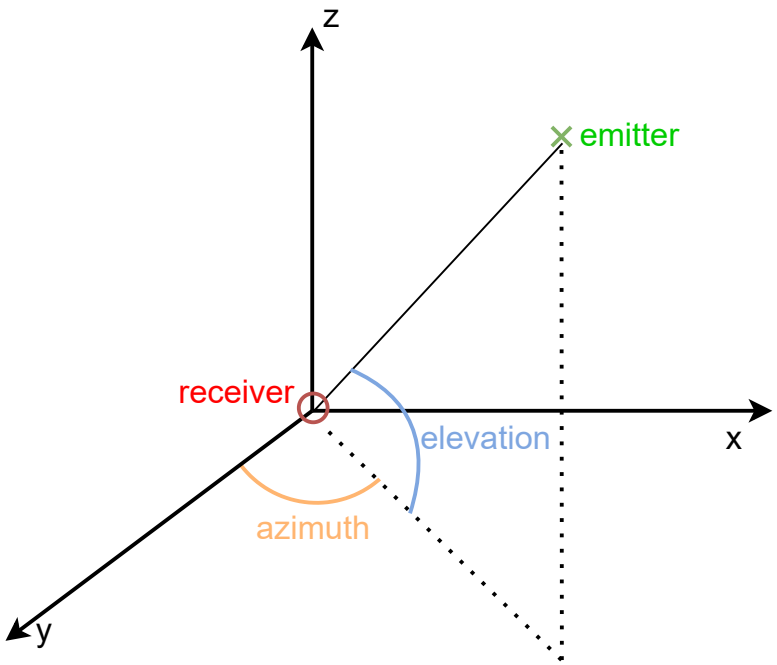


Figure 6: *AOA - azimuth and elevation*

where y_i, x_i and y_e, x_e represent y and x coordinates of receiver i and emitter, respectively. v_i^{AoA} is AoA noise of receiver i.

The elevation refers to the angle formed between the horizontal plane and the (LoS) to the emitter. The elevation angle between receiver i and emitter is calculated by:

$$Z_i^{AoA,el} = \arctan\left(\frac{z_i - z_e}{\sqrt{(y_i - y_e)^2 + (x_i - x_e)^2}}\right) + v_i^{AoA,el}. \tag{2.10}$$

The estimation of position $\mathcal{X}_e$ is calculated numerically similarly to TDoA, i.e. using maximum likelihood estimation as defined in Equation (2.8). The transformation of TDoA and AoA measurements into position vector is further described in the Section 2.1.5.

2.1.5 Transformation of TDoA and AoA measurements

The relation for TDoA and AoA measurements will be used to simulate the measurements from the ground-truth values of object position. In addition, the relations will be used in the estimation algorithm that provide the estimate of emmiter position. As will be discussed in subsequent sections, two distinct methods are employed for handling the measurements in estimation algorithms. These include direct access the ToA and AoA measurements or transforming these values into a position vector. The methodology of transformation is described in this section.

The core of the solution lies in a nonlinear least-squares method implemented in the **lsqnonlin** function in MATLAB. It minimizes the sum of squares of nonlinear functions

to obtain optimal solution. The function **lsqnonlin** solves the problem of form:

$$\min_{\mathcal{X}} \|f(\mathcal{X})\|_2^2 = \min_{\mathcal{X}}(f_1(\mathcal{X})^2 + f_2(\mathcal{X})^2 + \ldots + f_n(\mathcal{X})^2). \tag{2.11}$$

For convenience, the subscript in $\mathcal{X}_e$ will be dropped in the sequel. The function for the nonlinear least-squares solver is defined as an extended version of the minimization formula described in equation 2.8:

$$\hat{\mathcal{X}} = \arg\min_{\mathcal{X}}(\mathcal{Z} - \mathbf{h}_{\text{ext}}(\mathcal{X}))^\intercal \mathbf{Q}^{-1}(\mathcal{Z} - \mathbf{h}_{\text{ext}}(\mathcal{X})), \tag{2.12}$$

where $\mathcal{X}$ is the true position of the emitter, $\mathcal{Z}$ is the vector of TDoA ($\Delta\mathbf{t}$) and AoA (azimuth, elevation) noised measurements:

$$\mathcal{Z} = \begin{bmatrix} \Delta\mathbf{t} \\ Z_1^{AoA,az} \\ Z_1^{AoA,el} \\ \vdots \\ Z_i^{AoA,az} \\ Z_i^{AoA,el} \end{bmatrix}, \tag{2.13}$$

$\mathbf{h}_{\text{ext}}(\mathcal{X})$ is the extended measurement model coresponding to joint TDoA and AoA values evaluated at position of the emitter $\mathcal{X}$:

$$\mathbf{h}_{\text{ext}}(\mathcal{X}) = \begin{bmatrix} \mathbf{h}_{\text{TDoA}}(\mathcal{X}) \\ \mathbf{h}_{\text{AoA}}(\mathcal{X}) \end{bmatrix}, \tag{2.14}$$

and $\mathbf{Q}$ is a diagonal matrix with ToA and AoA noise variances along the diagonal:

$$\mathbf{Q} = diag\Big(\mathbf{Q}_{\text{TDoA}}, \mathbf{Q}_{\text{AoA}}\Big). \tag{2.15}$$

The function to be minimized adjusts the residuals (the difference between measured and predicted values) by the inverse of standard deviation of the measurement noise. It prioritizes the more reliable measurements in the optimization process by giving higher weight to the residuals with smaller expected errors.

In addition to the function $f(\mathcal{X})$, the optimalization process relies on the Jacobian matrix, which represents the partial derivatives of the measurement functions. The Jacobian calculation for TDoA and AoA is explored later in this section.

By dynamically updating the emitter's estimated position based on the feedback from both function $f(\mathcal{X})$ and its Jacobian, the optimalization process refines the emitter location estimation by iteratively minimizing the measurement discrepancies.

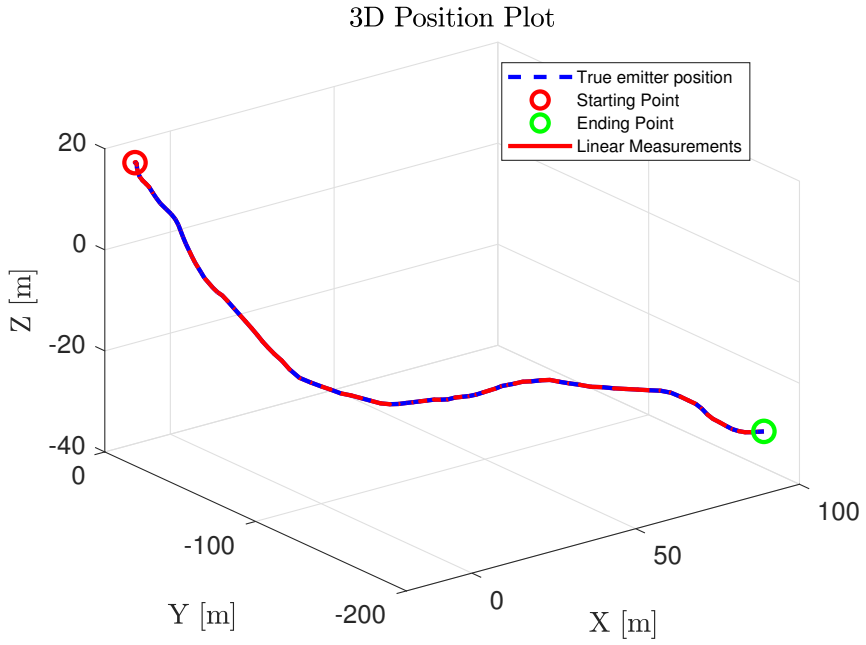


Figure 7: *Comparison of converted TDoA/AoA measurements with true emitter positions in 3D*

Additionally, vectors specifying the distance to the area defined above were used as bounds on the variables. Finding solution to (2.12) is an iterative process. For each iteration, the initial condition is based on the estimation from the previous estimation cycle.

The output of the nonlinear least-squares is the vector $\mathbf{Z}_{\text{lin}}$, which represents the converted measurement directly providing position values. As an example, positions computed for each time are visualized in comparison with true positions of emitter in Figure 7.

In addition to the estimation of the emitter's position, the covariance matrix of noise of coverted TDoA/AoA measurement (denote as $\mathbf{R}$) is required. This matrix is crucial for measuring the level of uncertainty in the position and its one of the input necessary for the Kalman filter algorithm. Further, the covariance matrix $\mathbf{R}$ is calculated approximately using the inverse of the Jacobian matrix:

$$\mathbf{R} = \left(\mathbf{J}^{\dagger}(\mathbf{Z}_{\text{lin}})\right) \cdot \mathbf{Q}_v \cdot \left(\mathbf{J}^{\dagger}(\mathbf{Z}_{\text{lin}})\right)^{\mathsf{T}}, \tag{2.16}$$

where $\mathbf{Q}_v$ represents covariance matrix of the noise associated with the TDoA and AoA measurements and $\dagger$ denotes pseudoinverse. The Jacobian matrix $\mathbf{J}(\cdot)$ consists of n rows corresponding to each available TDoA measurement and additional $2n$ rows for each available AoA measurement as follows.

The rows of $\mathbf{J}(\cdot)$ corresponding to TDoA measurements are given as:

$$J_i^{\mathrm{TDoA}}(\mathbf{Z}_{\mathrm{lin}}) = \left.\frac{\partial h_i(\mathcal{X}_e)}{\partial \mathcal{X}_e}\right|_{\mathcal{X}_e=\mathbf{Z}_{\mathrm{lin}}} = \frac{1}{c}\left(\frac{\mathcal{X}_e - \mathcal{X}_i}{\|\mathcal{X}_e - \mathcal{X}_i\|} - \frac{\mathcal{X}_e - \mathcal{X}_j}{\|\mathcal{X}_e - \mathcal{X}_j\|}\right)_{\mathcal{X}_e=\mathbf{Z}_{\mathrm{lin}}}, \tag{2.17}$$

where $\mathbf{Z}_{\mathrm{lin}}$ represents the current estimate of the emitter position. The resulting row of the Jacobian matrix $J_i^{\mathrm{TDoA}}(\mathbf{Z}_{\mathrm{lin}})$ reflects the sensitivity of the TDoA measurements to change in the estimated emitter position.

Given the position of the emitter $\mathcal{X}_e$ and the receiver $\mathcal{X}_r$, the difference vector is $\Delta\mathcal{X} = \mathcal{X}_r - \mathcal{X}_e$. Similarly to TDoA, the rows of $\mathbf{J}(\cdot)$ corresponding to AoA measurements are given as:

$$J_i^{\mathrm{AoA}}(\mathbf{Z}_{\mathrm{lin}}) = \left[\begin{array}{ccc} \frac{\Delta Y}{\Delta X^2+\Delta Y^2} & -\frac{\Delta X}{\Delta X^2+\Delta Y^2} & 0 \\ \frac{\Delta X\cdot\Delta Z}{r\sqrt{\Delta X^2+\Delta Y^2}} & \frac{\Delta Y\cdot\Delta Z}{r\sqrt{\Delta X^2+\Delta Y^2}} & -\frac{\Delta X^2+\Delta Y^2}{r\sqrt{\Delta X^2+\Delta Y^2}} \end{array}\right]\Bigg|_{\mathcal{X}_e=\mathbf{Z}_{\mathrm{lin}}}, \tag{2.18}$$

where ΔX, ΔY, and ΔZ are the components of $\Delta\mathcal{X}$, and r is equal to:

$$r = \Delta X^2 + \Delta Y^2 + \Delta Z^2. \tag{2.19}$$

The rows of the Jacoban matrix in Equation (2.18) effectively capture how changes in the emitter's position influence the measured angles.

3 Dynamic Object Motion Models

For the purpose of describing the motion models, it is necessary to define the state vector. The state vector is a fundamental concept in system modeling. It contains comprehensive list of variables which define the system current state. For a single dimension, assume that the state vector can be defined as follows:

$$\mathbf{x}_k = \begin{bmatrix} x_k & \dot{x}_k & \ddot{x}_k \end{bmatrix}^\intercal, \tag{3.1}$$

where x_k represents objects position, $\dot{x}_k$ velocity and $\ddot{x}_k$ an acceleration at a discrete time step k. Note, that in the following text the symbol $^\intercal$ represents matrix transpose while T represents the sampling period. In Section 3.2 modeling in 3-D will be explained. Given the state vector, the state equation can be expressed as:

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{\Gamma} w_k. \tag{3.2}$$

This equation represents a linear relationship involving noise w_k and the previous value of the state vector $\mathbf{x}_{k-1}$. The process noise may represent mismodeling of dynamics. Its characteristic is described by the covariance matrix denoted as $\mathbf{Q}$. Value of $\mathbf{Q}$ is typically obtained throught simulations, experiments or identifications [10].

3.1 Dynamic models in navigation

For the purpose of this thesis, four dynamic models will be described: the nearly constant velocity (NCV) model, Singer acceleration model, discrete white noise acceleration model (DWNA) and discrete Wiener process acceleration model (DWPA).

3.1.1 Nearly constant velocity

The nearly constant velocity (NCV) model state consists of position and velocity. The moodel is acquired through the discretization of the continuous white noise acceleration model [11]. The resulting discrete time state equation (3.2) becomes:

$$\mathbf{x}_{k+1} = \mathbf{F}_{\text{NCV}}\mathbf{x}_k + \mathbf{v}_k, \tag{3.3}$$

where $\mathbf{x}_k = \begin{bmatrix} x_k, \dot{x}_k \end{bmatrix}^\intercal$, and the transition matrix is:

$$\mathbf{F}_{\text{NCV}} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}, \tag{3.4}$$

and v_k is discrete time process noise with zero-mean and covariance:

$$\mathbf{Q}_{\text{NCV}} = \begin{bmatrix} \frac{T^3}{3} & \frac{T^2}{2} \\ \frac{T^2}{2} & T \end{bmatrix} \tilde{q}, \tag{3.5}$$

where **power spectral density** of the continuous time white noise, denoted as $\tilde{q}$, models the uncertainty of the motion. It is possible to achieve NCV model by choosing a relatively low value for $\tilde{q}$. To ensure the validity of the model, it is necessary for the changes in the velocity during sampling to be *small* compared to the actual velocity [11], [12].

3.1.2 Singer acceleration model

For the Singer model, the state vector $\mathbf{x}_k$ involves position x_k, velocity $\dot{x}_k$ and acceleration $\ddot{x}_k$. It is assumed that the acceleration is modeled by a first-order Markov process:

$$\ddot{x}_{k+1} = \rho_m \ddot{x}_k + \sqrt{(1-\rho_m)}\sigma_m r_k, \tag{3.6}$$

where $\rho_m = e^{-\beta T}$ and $\beta = \frac{1}{\tau_m}$. The parameters τ_m and σ_m represent target maneuver time constants, r_k is zero-mean unit-standard deviation Gaussian distributed random variable [10].

The Singer model is defined as:

$$\begin{bmatrix} x_{k+1} \\ \dot{x}_{k+1} \\ \ddot{x}_{k+1} \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & T & \frac{1}{\beta^2} \cdot (-1 + \beta T + \rho_m) \\ 1 & 1 & \frac{1}{\beta} \cdot (1 - \rho_m) \\ 0 & 0 & \rho_m \end{bmatrix}}_{\mathbf{F}_{\text{Singer}}} \begin{bmatrix} x_k \\ \dot{x}_k \\ \ddot{x}_k \end{bmatrix} + \mathbf{w}_k, \tag{3.7}$$

where $\mathbf{F}_{\text{Singer}}$ is **Singer model transition matrix**.

The formula for the process covariance matrix is given as:

$$\mathbf{Q}_{\text{Singer}} = \frac{2\sigma_m^2}{\tau_m} \begin{bmatrix} q_{11} & q_{12} & q_{13} \\ q_{21} & q_{22} & q_{23} \\ q_{31} & q_{32} & q_{33} \end{bmatrix}, \tag{3.8}$$

where:

$$\begin{aligned}
q_{11} &= \frac{1}{2\beta^5} \cdot \left[1 - e^{-2\beta T} + 2\beta T + \frac{2\beta^3 T^3}{3} - 2\beta^2 T^2 - 4\beta T e^{-\beta T}\right], \\
q_{12} &= \frac{1}{2\beta^4} \cdot \left[e^{-2\beta T} + 1 - 2e^{-\beta T} + 2\beta T e^{-\beta T} - 2\beta T + \beta^2 T^2\right], \\
q_{13} &= \frac{1}{2\beta^3} \cdot \left[1 - e^{-2\beta T} - 2\beta T e^{-\beta T}\right], \\
q_{22} &= \frac{1}{2\beta^4} \cdot \left[4e^{-2\beta T} - 3 - e^{-2\beta T} + 2\beta T\right], \\
q_{23} &= \frac{1}{2\beta^2} \cdot \left[e^{-2\beta T} + 1 - 2e^{-\beta T}\right], \\
q_{33} &= \frac{1}{2\beta} \cdot \left[1 - e^{-2\beta T}\right].
\end{aligned} \quad . \tag{3.9}$$

In the case that the sampling interval is short, that means $T \ll \tau_m$, the formulation is given as:

$$\lim_{\beta T \to 0} \mathbf{Q}_{\text{Singer}} = \frac{2\sigma_m^2}{\tau_m} \begin{bmatrix} \frac{T^5}{20} & \frac{T^4}{8} & \frac{T^3}{6} \\ \frac{T^4}{8} & \frac{T^3}{3} & \frac{T^2}{2} \\ \frac{T^3}{6} & \frac{T^2}{2} & T \end{bmatrix}. \tag{3.10}$$

In the case that maneuver time constant is much smaller than the sampling period ($T \gg \tau_m$), an accurate estimate of acceleration cannot be obtained, and thus the model can use only position and velocity as is stated in [10]:

$$\mathbf{F}_{\text{Singer}} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \qquad \mathbf{Q}_{\text{Singer}} = \frac{2\sigma_m^2}{\tau_m} \begin{bmatrix} \frac{T^3}{3} & \frac{T^2}{2} \\ \frac{T^2}{2} & T \end{bmatrix}. \tag{3.11}$$

3.1.3 Discrete white noise acceleration model

The state vector $\mathbf{x}$ of Discrete White Noise Acceleration (DWNA) model consists of position and velocity. The model is a common kinematics model which is defined in the discrete time. In the model, the process noise denoted as w_k, is a scalar-valued zero-mean white sequence with variance σ_v^2 at each time step. The DWNA model is:

$$\mathbf{x}_{k+1} = \mathbf{F}_{\text{DWNA}} \mathbf{x}_k + \mathbf{\Gamma}_{\text{DWNA}} w_k, \tag{3.12}$$

where $\mathbf{\Gamma}_{\text{DWNA}}$ is a noise gain [11]. The state equaton for the DWNA model is shown in equation (3.12), where transition matrix and vector gain multiplying the scalar process

noise are given as:

$$\mathbf{F}_{\text{DWNA}} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \qquad \mathbf{\Gamma}_{\text{DWNA}} = \begin{bmatrix} \frac{1}{2}T^2 \\ T \end{bmatrix}. \tag{3.13}$$

The covariance matrix of the process noise is:

$$\mathbf{Q}_{\text{DWNA}} = E[\mathbf{\Gamma}_{\text{DWNA}} w_k w_k \mathbf{\Gamma}_{\text{DWNA}}^{\intercal}] = \mathbf{\Gamma}_{\text{DWNA}} \sigma_v^2 \mathbf{\Gamma}_{\text{DWNA}}^{\intercal} = \begin{bmatrix} \frac{1}{4}T^4 & \frac{1}{2}T^3 \\ \frac{1}{2}T^3 & T^2 \end{bmatrix} \sigma_v^2. \tag{3.14}$$

Note that the rank of $\mathbf{Q}_{\text{DWNA}}$ is 1, making it singular which can lead to a problem in some cases [11].

3.1.4 Discrete Wiener process acceleration model

The discrete Wiener process acceleration (DWPA) model describes dynamics of three state variables: position, velocity and acceleration. The model can be described using the state equation:

$$\mathbf{x}_{k+1} = \mathbf{F}_{\text{DWPA}} \mathbf{x}_k + \mathbf{\Gamma}_{\text{DWPA}} w_k, \tag{3.15}$$

where:

$$\mathbf{F}_{\text{DWPA}} = \begin{bmatrix} 1 & T & \frac{1}{2}T^2 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix} \qquad \mathbf{\Gamma}_{\text{DWPA}} = \begin{bmatrix} \frac{1}{2}T^2 \\ T \\ 1 \end{bmatrix} \tag{3.16}$$

and v_k represents white noise with a zero-mean and variance σ_v^2 $v_k \sim \mathcal{N}(0, \sigma_v^2)$ [11].

The covariance matrix of the process noise is given as:

$$\mathbf{Q}_{\text{DWPA}} = \mathbf{\Gamma}_{\text{DWPA}} \sigma_v^2 \mathbf{\Gamma}_{\text{DWPA}}^{\intercal} = \begin{bmatrix} \frac{1}{4}T^4 & \frac{1}{2}T^3 & \frac{1}{2}T^2 \\ \frac{1}{2}T^3 & \frac{1}{2}T^2 & T \\ \frac{1}{2}T^2 & T & 1 \end{bmatrix} \sigma_v^2 \tag{3.17}$$

3.2 Modeling for several spatial dimentions

A moving object within our three-dimensional world can easily be described by a vector which contains its three-dimensional Cartesian position and velocity vector. The state vector is therefore as follows:

$$\mathbf{x} = \begin{bmatrix} x & \dot{x} & \ddot{x} & y & \dot{y} & \ddot{y} & z & \dot{z} & \ddot{z} \end{bmatrix}^\mathsf{T}, \tag{3.18}$$

where $\begin{bmatrix} x & y & z \end{bmatrix}^\mathsf{T}$ is a vector of the object position, $\begin{bmatrix} \dot{x} & \dot{y} & \dot{z} \end{bmatrix}^\mathsf{T}$ is its velocity vector and $\begin{bmatrix} \ddot{x} & \ddot{y} & \ddot{z} \end{bmatrix}^\mathsf{T}$ is the vector of accelerations [13].

The dynamics for $\mathbf{x}$ can be transferred into matrix form:

$$\mathbf{x}_{k+1} = \mathbf{F}_k \mathbf{x}_k + \mathbf{\Gamma} \mathbf{w}_k. \tag{3.19}$$

The resulting form of equation for 3D scenario is as follows:

$$\underbrace{\begin{bmatrix} x_{k+1} \\ \dot{x}_{k+1} \\ \ddot{x}_{k+1} \\ y_{k+1} \\ \dot{y}_{k+1} \\ \ddot{y}_{k+1} \\ z_{k+1} \\ \dot{z}_{k+1} \\ \ddot{z}_{k+1} \end{bmatrix}}_{\mathbf{x}_{k+1}} = \underbrace{\begin{bmatrix} \mathbf{F}_\alpha & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_\alpha & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{F}_\alpha \end{bmatrix}}_{\mathbf{F}_k} \underbrace{\begin{bmatrix} x_k \\ \dot{x}_k \\ \ddot{x}_k \\ y_k \\ \dot{y}_k \\ \ddot{y}_k \\ z_k \\ \dot{z}_k \\ \ddot{z}_k \end{bmatrix}}_{\mathbf{x}_k} + \begin{bmatrix} \mathbf{\Gamma}_\alpha & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Gamma}_\alpha & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{\Gamma}_\alpha \end{bmatrix} \underbrace{\begin{bmatrix} \mathbf{w}_{k,x} \\ \mathbf{w}_{k,y} \\ \mathbf{w}_{k,z} \end{bmatrix}}_{\mathbf{w}_k}, \tag{3.20}$$

where $\alpha \in \{\text{NCV, Singer, DWNA, DWPA}\}$ and $\mathbf{w}_k \sim \mathcal{N}(0, \mathbf{Q}_k)$. Value of $\mathbf{Q}_k$ can be formulated by taking covariance of $\mathbf{w}_k$:

$$\begin{aligned} \mathbf{Q}_k &= Cov(\mathbf{w}_k) = E[\mathbf{w}_k \mathbf{w}_k^\mathsf{T}] \\ \mathbf{Q}_k &= \begin{bmatrix} \mathbf{Q}_\alpha & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_\alpha & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{Q}_\alpha \end{bmatrix}, \end{aligned} \tag{3.21}$$

where $\mathbf{Q}_\alpha$ represents the specific covariance matrix of the process noise for the chosen dynamic model.

3.3 Observation model

While dynamic models predict the state of an object, the observation model describes how observations of the system state are obtained, including the influence of measurement noise. The observation equation is defined as follows:

$$\mathbf{z}_k = \mathbf{H}\mathbf{x}_k + \mathbf{v}_k, \tag{3.22}$$

where v_k represents measurement noise. Noises $\mathbf{w}_k$ and $\mathbf{v}_k$ are assumed to be uncorrelated zero-mean Gaussian white noises:

$$\begin{aligned} E(\mathbf{w}_k) = 0, \qquad & COV(\mathbf{w}_k) = E(\mathbf{w}_k\mathbf{w}_k^\intercal) = \mathbf{Q}_k, \\ E(\mathbf{v}_k) = 0, \qquad & COV(\mathbf{v}_k) = E(\mathbf{v}_k\mathbf{v}_k^\intercal) = \mathbf{R}_k, \\ E(\mathbf{v}_k\mathbf{w}_k^\intercal) = 0, & \end{aligned} \tag{3.23}$$

where $\mathbf{Q}_k$ and $\mathbf{R}_k$ are covariance matrices of the noises of the state equation and measurement equation [7], [14], [15].

In real-world scenarios, observation of the system often does not follow a linear relationship with the state vector. It is common problem in the complex systems. The nonlinear observation equation can be expressed in following form:

$$\mathbf{z}_k = h(\mathbf{x}_k) + \mathbf{v}_k, \tag{3.24}$$

where $h(\cdot)$ represents a nonlinear function. Nonlinear equations were discussed earlier in the context of TDOA measurements, where observation was given as:

$$\Delta t = h(\mathcal{X}_e) + \mathbf{v}. \tag{3.25}$$

4 State Estimation

State estimation plays a pivotal role in the realm of control systems and navigation to provide current state of dynamic system based on an observation. It is a fundamental method that enables determination of the object trajectory over time.

This section delves into the algorithms of the Kalman filter and the Extended Kalman filter. The KF is the best linear estimator with minimal means square error (MSE) for estimating the state of linear dynamic systems. However, many real-world systems exhibit non-linear dynamics or measurements, prohibiting the application of the standard Kalman filter. For these scenarios, the Extended Kalman filter modifies the approach to handle non-linear systems by linearizing them at the current estimate. Both of these algorithms will be described in the following Sections 4.1 and 4.2.

Real-world systems often cannot be accurately or completely described by a single model due to the presence of uncertainties or complex system behaviour. This observation leads to the multiple model approach, a technique that does not rely on a single state-space model. Instead, multiple models are implemented to enhance the overall performance of state estimation. This approach leverages the use of multiple models to create more realistic predictions. Section 4.3 will describe the Interacting Multiple Model (IMM) algorithm, which implements these principles.

4.1 Kalman filter

The Kalman filter (KF) is an algorithm used for estimating the state of a dynamic system. The estimation is based on the combination of measurements which are collected at the discrete time intervals and predictions given by the system's dynamic and measurement model. The state dynamics is described as:

$$\mathbf{x}_{k+1} = \mathbf{F}_k \mathbf{x}_k + \mathbf{w}_k. \tag{4.1}$$

The values of the measurements are usually affected by noise [15]. The measurement equation in the KF is assumed to be linear, which is fundamental requirement for the formulation of the filter. The equation is determined as:

$$\mathbf{z}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{v}_k, \tag{4.2}$$

where $\mathbf{z}_k$ represents the measurement vector at time step k, $\mathbf{H}_k$ is measurement matrix and $\mathbf{v}_k$ denotes the measurement noise.

The KF has been applied in the field of target tracking and navigation [14]. The KF uses knowledge of both deterministic and statistical properties of the system to provide optimal estimates given the information available. To achieve this goal, each iteration

must carry more information than just the state estimate. That is, the KF keeps track of uncertainties in its estimates, which indicate how large errors in the estimates of different state elements are and how they are correlated [7]. The estimated mean of the state vector is linked with covariance matrix describing errors which represent the uncertainties in the state estimation provided by the filter and the degree of correlation between errors in these estimates. KF algorithm is recursive and the initial values of the state vector and covariance matrix must be set [7].

4.1.1 KF Algorithm

The Kalman filter algorithm can be derived throught many approaches, which are provided in [15]. The resulting algorithm is given in three main steps:

1. Initialization $k = 0$

In this step, initial values for estimated state and covariance matrix have to be chosen:

$$p\Big[\mathbf{x}_{0|-1}|\mathbf{z}^{-1}\Big] = p\Big[\mathbf{x}_0\Big] = \mathcal{N}\Big[x_0; \hat{\mathbf{x}}_{0|-1}, \mathbf{P}_{0|-1}\Big], \tag{4.3}$$

where $\hat{\mathbf{x}}_{0|-1}$ and $\mathbf{P}_{0|-1}$ denote the state estimate and covariance matrix respectively. Note that the assumption of Gaussianity is made due to IMM algorithm.

2. Measurement update $k = 0, 1, ...$

In this step, the measurements are incorporated correcting the estimate as:

$$\begin{aligned} \hat{\mathbf{x}}_{k|k} &= \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k(\mathbf{z}_k - \mathbf{H}_k\hat{\mathbf{x}}_{k|k-1}), \\ \mathbf{K}_k &= \mathbf{P}_{k|k-1}\mathbf{H}_k^\intercal(\mathbf{H}_k\mathbf{P}_{k|k-1}\mathbf{H}_k^\intercal + \mathbf{R_k})^{-1}, \\ \mathbf{P}_{k|k} &= \mathbf{P}_{k|k-1} - \mathbf{K}_k\mathbf{H}_k\mathbf{P}_{k|k-1}, \end{aligned} \tag{4.4}$$

where $\mathbf{K}_k$ represents the **Kalman gain matrix** (which determines how much weight is given to the measurements), $\hat{\mathbf{x}}_{k|k}$ denotes the filtered state estimate and $\mathbf{P}_{k|k}$ is the error covariance matrix. Additionally, $\mathbf{P}_{k|k-1}$ and $\hat{\mathbf{x}}_{k|k-1}$ represent the prior covariance matrix and state estimate.

3. Prediction $k = 1, 2, ...$

In this step, the prediction of the estimate is calculated:

$$\begin{aligned} \hat{\mathbf{x}}_{k|k-1} &= \mathbf{F}\hat{\mathbf{x}}_{k-1|k-1}, \\ \mathbf{P}_{k|k-1} &= \mathbf{F}\mathbf{P}_{k-1|k-1}\mathbf{F}^\intercal + \mathbf{Q}. \end{aligned} \tag{4.5}$$

Note that how the algorithm is influenced by the level of uncertainty given by the model and the initialization, ie. $\mathbf{Q}$, $\mathbf{R}$ and $\mathbf{P}_{0|-1}$. If the uncertainty about the motion model is "high", the Kalman gain will also be "high", giving more emphasis on the measurements. On the other hand, if the measurement noise covariance, ie. the value of the $\mathbf{R}$, is "high", the Kalman gain will be „low". In this case, the filter trusts the motion model more than the measurements [7].

4.2 Extended Kalman filter

In the previous subsection, filtering for linear systems was considered. The issue is that in real-world scenarios, the assumption of linearity often does not hold, so the Kalman filter cannot be used. The assumption of linearity of measurements is abandoned, so the measurement equation for extended Kalman filter is defined as:

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k, \tag{4.6}$$

where $\mathbf{h}$ is non-linear function [15].

The EKF employs Taylor's series expansion of the function $\mathbf{h}(\mathbf{x}_k)$ at the best available estimate $\hat{\mathbf{x}}_{k|k-1}$. In this process, only the first two terms of the Taylor series are taken into account, reflecting the need for a linear approximation of the functions. This leads to:

$$\mathbf{h}(\mathbf{x}_k) \approx \mathbf{h}(\hat{\mathbf{x}}_{k|k-1}) + \mathbf{H}(\hat{\mathbf{x}}_{k|k-1})(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1}), \tag{4.7}$$

where:

$$\mathbf{H}(\hat{\mathbf{x}}_{k|k-1}) = \left.\frac{\partial \mathbf{h}(\mathbf{x}_k)}{\partial \mathbf{x}_k}\right|_{\mathbf{x}_k=\hat{\mathbf{x}}_{k|k-1}}, \tag{4.8}$$

$$\mathbf{H}(\hat{\mathbf{x}}_{k|k-1}) = \begin{bmatrix} \frac{\partial h_1(\hat{\mathbf{x}}_{k|k-1})}{\partial x_1} & \cdots & \frac{\partial h_1(\hat{\mathbf{x}}_{k|k-1})}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_m(\hat{\mathbf{x}}_{k|k-1})}{\partial x_1} & \cdots & \frac{\partial h_m(\hat{\mathbf{x}}_{k|k-1})}{\partial x_n} \end{bmatrix}, \tag{4.9}$$

is the Jacobian matrix of the function $\mathbf{h}$ at the point of actual best estimate $\hat{\mathbf{x}}_{k|k-1}$ [15]. As discussed previously, the Jacobian matrices for TDoA and AoA measurements are detailed in Section 2.1.5. Note that matrix $\mathbf{H}(\hat{\mathbf{x}}_{k|k-1})$ is equivalent to $\mathbf{J}(\hat{\mathbf{x}}_{k|k-1})$ as defined in Equation (2.16). The resulting EKF update is given by:

$$\begin{aligned} \hat{\mathbf{x}}_{k|k} &= \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k(\mathbf{z}_k - \mathbf{h}_k(\hat{\mathbf{x}}_{k|k-1})), \\ \mathbf{K}_k &= \mathbf{P}_{k|k-1}\mathbf{H}_k^\intercal(\hat{x}_{k|k-1})(\mathbf{H}_k(\hat{x}_{k|k-1})\mathbf{P}_{k|k-1}\mathbf{h}_k^\intercal(\hat{x}_{k|k-1}) + \mathbf{R_k})^{-1}, \\ \mathbf{P}_{k|k} &= \mathbf{P}_{k|k-1} - \mathbf{K}_k\mathbf{H}_k(\hat{x}_{k|k-1})\mathbf{P}_{k|k-1}, \end{aligned} \tag{4.10}$$

Note that the motion model is considered linear in this thesis and thus the prediction is given by the KF.

4.3 Multiple model approach

In this section, the basis of the multiple model approach is explained, in particular, the interacting multiple model method, which is used in a later sections of this thesis to estimate the position of objects, is described.

Multiple model approach (MMA) is used in control theory, where multiple models are combined to make predictions or decisions [16]. The approach does not rely on single state-space model. Instead, multiple models are used to improve overall performance of system. Systems where the multiple model approach is leveraged, are called **hybrid systems**. These systems contain continuous and discrete uncertainties.

4.3.1 Formulation

Consider r possible models of the system $\{M(j)\}_{j=1}^{r}$ and $M \in \{M(j)\}_{j=1}^{r}$ being currently active model further referred to as mode. The prior probability that system is in mode $M = M(j)$ is denoted with:

$$P\Big\{M(j)|Z^0\Big\} = \mu_0(j) \quad j = 1, 2, 3, \ldots, r, \tag{4.11}$$

where Z^0 represents the prior information. Note that $\sum_{j=1}^{r} \mu_0(j) = 1$. For convenience, it will be assumed that all models are linear-Gaussian. Subsequently, the multiple model approach will be demonstrated for fixed modes in time and for switching modes.

4.3.2 The Multiple Model Approach for fixed modes

In the MMA, **fixed modes** refer to a scenario where fixed set of possible models are given and one of them is correct for all time instances. For fixed modes the Bayesian formula for posterior probability will be applied:

$$\mu_k(j) = \frac{\Lambda(j)\mu_{k-1}(j)}{\sum_{i=1}^{r} \Lambda(i)\mu_{k-1}(i)}, \tag{4.12}$$

where $\Lambda(j)$ is a **likelihood function of mode** $M = M(j)$:

$$\Lambda(j) = p\left(\mathbf{z}_k|Z^{k-1}, M(j)\right) = \mathcal{N}\{\nu_k(j); 0, S_k(j)\}, \tag{4.13}$$

where $\nu(j)$ can be defined as inovation of mode $M = M(j)$ and $S(j)$ is its covariance. For each mode, a filter is assigned that provides mode-conditioned state estimates and mode-conditioned covariances.

The principle of MMA with a fixed model is that there is a bank of filters and according to the outcome of likelihood function the correct mode is selected from it. The principle is illustrated in Figure 8. Based on the selected mode state estimate $\hat{\mathbf{x}}_{k|k-1}$, and its

covariance $\mathbf{P}_{k|k-1}$ are computed . The latest mode probabilities are used to calculate

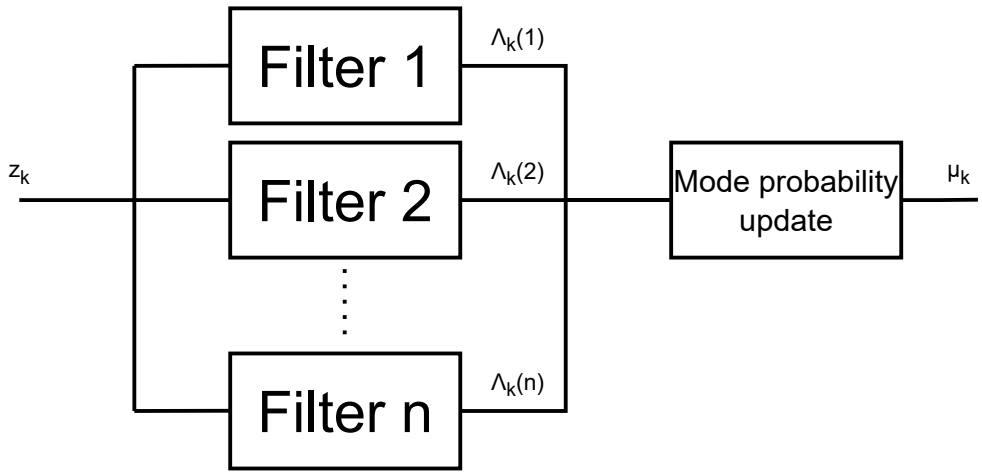


Figure 8: *Mode probability update*

estimates and covariances:

$$\begin{aligned}\hat{\mathbf{x}}_{k|k} &= \sum_{j=1}^{r} \mu_k(j)\hat{\mathbf{x}}^j_{k|k} \\ \mathbf{P}_{k|k} &= \sum_{j=1}^{r} \mu_k(j)\left(\mathbf{P}^j_{k|k} + [\hat{\mathbf{x}}^j_{k|k} - \hat{\mathbf{x}}_{k|k}][\hat{\mathbf{x}}^j_{k|k} - \hat{\mathbf{x}}_{k|k}]^\intercal\right).\end{aligned} \tag{4.14}$$

Once initialized, the filters proceed to operate recursively based on their own estimates. The likelihood functions associated to the estimates are used to refresh the probabilities of each modes, which are used to combine the estimates and covariances associated to specific mode. Note that under these assumptions, the posterior pdf of the state of the system is a Gaussian mixture:

$$p[\mathbf{x}_k|Z^k] = \sum_{j=1}^{r} \mu_k(j)\mathcal{N}[\mathbf{x}_k; \hat{\mathbf{x}}^j_{k|k}, \mathbf{P}^j_{k|k}]. \tag{4.15}$$

However, there are limitations to the given formulas. First, there has to be one correct model among all cosidered models, which has been in effect for all time instances.

Convergence of the probability of correct mode can be achieved if there is a correct mode (or near to correct mode) and no mode-jump occurs during the estimation process. In that case, the probability of the correct mode will coverge to one [16].

4.3.3 The Multiple Model Approach for switching modes

This formulation assumes that the modes can change over time. The hybrid system is considered to be modeled by:

$$\begin{aligned} \mathbf{x}_k &= \mathbf{F}_{M_k}\mathbf{x}_{k-1} + \mathbf{w}_{k-1,M_k}, \\ \mathbf{z}_k &= \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k, \end{aligned} \tag{4.16}$$

where M_k is mode at the time k. Similarly to the previous chapter, the correct mode M_k is searched among all available models **at the time** k:

$$M_k \in \{M(j)\}_{j=1}^{r}. \tag{4.17}$$

Regarding the taxonomy, M_k can be referred to as the **modal state**, which is a discrete random variable. Otherwise, $\mathbf{x}_k$ is called the **base state** and its value is usually continuous. Switches between modes suggests the definition of **mode history** or **sequence of the modes**:

$$M^{k,l} \in \Big\{M(i_{1,l}), M(i_{2,l}), \ldots, M(i_{k,l})\Big\} \quad 1 \leq i_{k,l} \leq r, \tag{4.18}$$

where $l = 1, \ldots, r^k$ denotes a specific sequence of the mode history, r represents the number of all modes, k is the time step and $i_{k,l}$ is the index of the model at the time k from the l-th mode history.

The process that allows switching between the models is named **mode jump process**. Under time invariant mode transitions and their independence with respect to the base state, the mode jump process is a homogenous Markov chain with known mode transition probabilities:

$$P(i,j) = P\Big\{M_k = M(j) | M_{k-1} = M(i)\Big\}. \tag{4.19}$$

The event, when the mode $M = M(j)$ is active at time k can be denoted with:

$$M_k(j) \equiv \{M_k = M(j)\}. \tag{4.20}$$

The mode history can be expressed using its **parent sequence** $M^{k-1,s}$ which states for s-th sequence from 0 to $k-1$. The resulting formulation:

$$M^{k,l} = \Big\{M^{k-1,s}, M_k(j)\Big\}, \tag{4.21}$$

where the first element of the set represents history s to the time instant $k-1$ and $M_k(j)$ is the new element of the parent sequence. The probabilities of transition (4.19) can be

denoted as follows:

$$p(i,j) = P\Big\{M_k(j)|M^{k-1,s}\Big\}, \tag{4.22}$$

where $M^{k-1,s} = P\Big\{M^{k-2,r}, M_{k-1}(j)\Big\}$.The probability of mode history can be further determined:

$$\mu^{k,l} = P\Big\{M^{k,l}|Z^k\Big\} = \frac{1}{c}p[\mathbf{z}_k|M^{k,l}, Z^{k-1}]p(i,j)\mu^{k-1,s}, \tag{4.23}$$

where c is a normalization constant. The complication lies in the conditional pdf which eventually results to be a Gaussian mixture with exponentially increasing number of terms. This exponential growth makes it impractical to handle. Feasible solutions lie in suboptional techniques which are using only N terms e.g., with the highest probabilities, the rest are ignored. The probabilities have to be renormalized after the approximation so they sum to unity. Another famous suboptimal solution called the **Interacting Multiple Model** will be described in the following subsection.

4.4 The Interacting Multiple Model

In the Interacting Multiple Model (IMM), there are r filters running in parallel and the total probability can be computed by:

$$p[\mathbf{x}_k|Z^k] = \sum_{j=1}^{r} p[\mathbf{x}_k|M_k(j), Z^k]P\Big\{M_k(j)|Z^k\Big\}, \tag{4.24}$$

which can also be expressed as:

$$p[\mathbf{x}_k|Z^k] = \sum_{j=1}^{r} p[\mathbf{x}_k|M_k(j), \mathbf{z}_k, Z^{k-1}]\mu_j(k). \tag{4.25}$$

Bayesian theorem can be used to compute the posterior pdf:

$$p[\mathbf{x}_k|M_k(j), \mathbf{z}_k, Z^{k-1}] = \frac{p[\mathbf{z}_k|M_k(j), \mathbf{x}_k]}{p[\mathbf{z}_k|M_k(j), Z^{k-1}]}p[\mathbf{x}_k|M_k(j), Z^{k-1}]. \tag{4.26}$$

The prior and posterior probability calculation represent one cycle of the estimation of $M_k(j)$. With a usage of total probability theorem, the prior pdf can be expressed as:

$$\begin{aligned} p[\mathbf{x}_k|M_k(j), Z^{k-1}] &= \sum_{i=1}^{r} p[\mathbf{x}_k|M_k(j), M_{k-1}(i), Z^{k-1}]P\Big\{M_{k-1}(i)|M_k(j), Z^{k-1}\Big\}, \\ p[\mathbf{x}_k|M_k(j), Z^{k-1}] &= \sum_{i=1}^{r} p[\mathbf{x}_k|M_k(j), M_{k-1}(i), Z^{k-1}]\mu_{k-1|k-1}(i|j), \end{aligned} \tag{4.27}$$

where the weights, denotes as $\mu_{k-1|k-1}(i|j)$, are given by posterior probabilities. With the use of approximation, where the past information up to $k-1$ denoted with Z^{k-1} is summarized by r model-conditioned estimates and covariances, the pdf can be formulated as mixture of Gaussian pdfs:

$$p[\mathbf{x}_k|M_k(j), Z^{k-1}] = \sum_{i=1}^{r} \mathcal{N}\left\{\mathbf{x}_k; \hat{\mathbf{x}}^{i|j}_{k|k-1}, \mathbf{P}^{i|j}_{k|k-1}\right\} \mu_{k-1|k-1}(i|j), \tag{4.28}$$

where:

$$\begin{aligned} \hat{\mathbf{x}}^{i|j}_{k|k-1} &= \mathbf{F}_j \hat{\mathbf{x}}^{i}_{k-1|k-1}, \\ \mathbf{P}^{i|j}_{k|k-1} &= \mathbf{F}_j \mathbf{P}^{i}_{k-1|k-1} \mathbf{F}_j^{\intercal} + \mathbf{Q}_j. \end{aligned} \tag{4.29}$$

Then **moment matching** by a single Gaussian pdf can be used:

$$p[\mathbf{x}_k|M_k(j), Z^{k-1}] \approx \mathcal{N}\left\{\mathbf{x}_k; \hat{\mathbf{x}}^{0j}_{k|k-1}, \mathbf{P}^{0j}_{k|k-1}\right\}, \tag{4.30}$$

where:

$$\begin{aligned} \hat{\mathbf{x}}^{0j}_{k|k-1} &= \sum_{i=1}^{r} \mu_{k-1|k-1}(i|j) \underbrace{\hat{\mathbf{x}}^{i|j}_{k|k-1}}_{\mathbf{F}_j \hat{\mathbf{x}}^{i}_{k-1|k-1}} \\ &= \mathbf{F}_j \hat{\mathbf{x}}^{0j}_{k-1|k-1} \\ \mathbf{P}^{0j}_{k|k-1} &= \mathbf{F}_j \mathbf{P}^{0j}_{k-1|k-1} \mathbf{F}_j^{\intercal} + \mathbf{Q}_j, \end{aligned} \tag{4.31}$$

with:

$$\begin{aligned} \hat{\mathbf{x}}^{0j}_{k-1|k-1} &= \sum_{j=1}^{r} \mu_{k-1|k-1}(i|j) \hat{\mathbf{x}}^{i|j}_{k-1|k-1} \\ \mathbf{P}^{0j}_{k-1|k-1} &= \sum_{j=1}^{r} \mu_{k-1|k-1}(i|j) \left(\mathbf{P}^{i|j}_{k-1|k-1} + [\hat{\mathbf{x}}^{i|j}_{k-1|k-1} - \hat{\mathbf{x}}_{k-1|k-1}][\hat{\mathbf{x}}^{i|j}_{k-1|k-1} - \hat{\mathbf{x}}_{k-1|k-1}]^{\intercal}\right), \end{aligned} \tag{4.32}$$

which may be called as mixed state estimates.

For further understanding of the IMM approach, one recursion cycle of IMM will be shown. Each cycle of IMM consists of three main steps. The first step is called **interaction/mixing** (Figure 9). During this step estimates $\hat{x}^{i}_{k-1|k-1}$ are mixed with the weights $\mu_{k-1|k-1}(i|j)$ referred to as mixing probabilities. The second step is **filtering** (Figure 10) which updates estimates based on the measurement $\mathbf{z}_k$. The last step is **combination** which combines estimates and covariances of each filters which is illustrated in the Figure 11. The algorithm will be disscussed in more detail in chapter 4.4.1 [17].

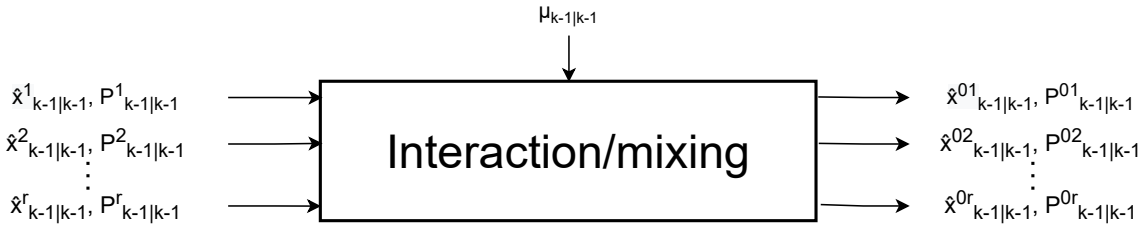


Figure 9: *IMM - interaction step [17]*

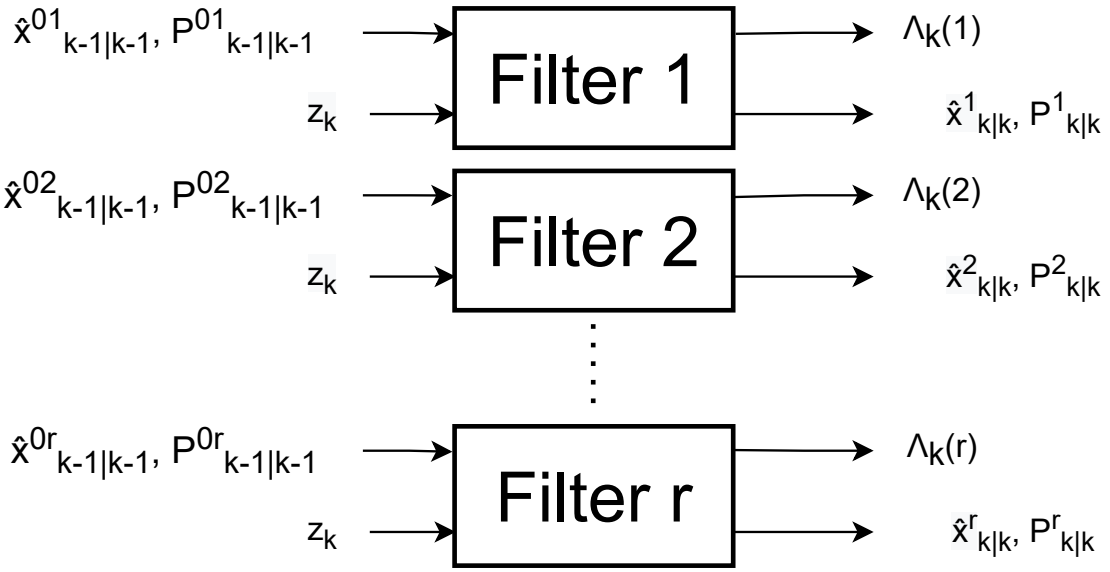


Figure 10: *IMM - Filtering step [17]*

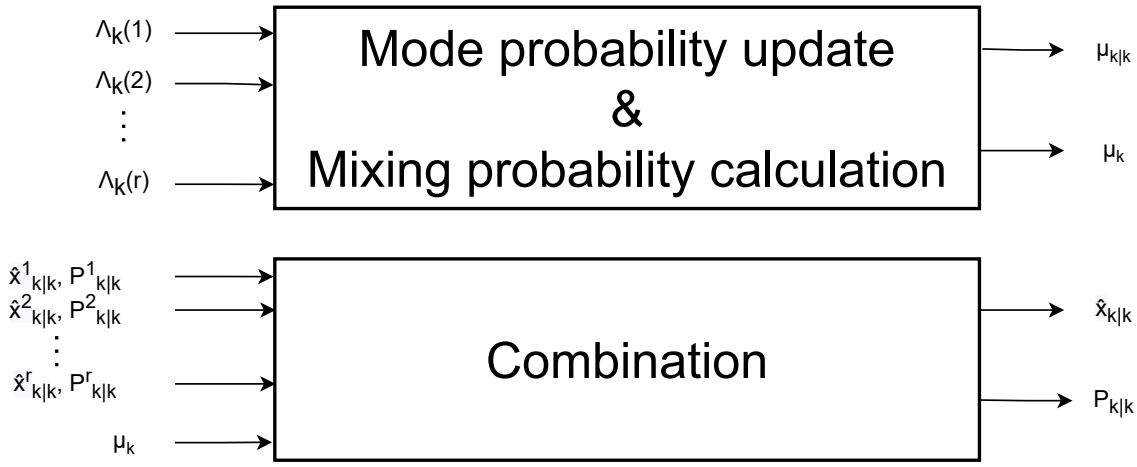


Figure 11: *IMM - Combination step [17]*

4.4.1 Algorithm description

The IMM algorithm has three major properties: it is recursive, modular and its computional requirements are fixed during each cycle [5], [17]. Each cycle is composed of the following steps:

1. Calculation of the mixing probabilities

In this step, it is necessary to calculate the probability that model $M(i)$ was active at time $k-1$ in the case where $M(j)$ is active at the time k under the condition Z^{k-1}. Mixing probability is given by formula:

$$\mu_{k-1|k-1}(i|j) = \frac{1}{\overline{c}(j)} P\Big\{M_k(j)|M_{k-1}(i), Z^{k-1}\Big\} P\Big\{M_{k-1}(i)|Z^{k-1}\Big\}. \tag{4.33}$$

The formula can be expressed as:

$$\mu_{k-1|k-1}(i|j) = \frac{1}{\overline{c}(j)} p(i,j)\mu_{k-1}(i), \tag{4.34}$$

where $\mu_{k-1}(i)$ is mode probability. This formulation allows to calculate mixing at the beginning of each cycle because $\mu_{k-1|k-1}(i|j)$ is conditioned on Z^{k-1}. Constant $\overline{c}(j)$ is normalization factor computed by:

$$\overline{c}(j) = \sum_{i=1}^{r} p(ij)\mu_{k-1}(i). \tag{4.35}$$

2. Mixing

After the calculation of the mixing probabilities, an initial condition is obtained for all filters by mixing the state estimates of all the parallel filters. The mixed initial condition for the filter matched to $M_k(j)$ can be express as:

$$\hat{\mathbf{x}}^{0j}_{k-1|k-1} = \sum_{i=1}^{r} \hat{\mathbf{x}}^{i}_{k-1|k-1}\, \mu_{k-1|k-1}(i|j). \tag{4.36}$$

This also applies to the covariance:

$$\begin{aligned} \mathbf{P}^{0j}_{k-1|k-1} &= \sum_{i=1}^{r} \mu_{k-1|k-1}(i|j) \left\{\mathbf{P}^{i}_{k-1|k-1} + [\hat{\mathbf{x}}^{i}_{k-1|k-1} - \hat{\mathbf{x}}^{0j}_{k-1|k-1}] \cdot [\cdot]^{\intercal}\right\} \\ j &= 1, \ldots, r. \end{aligned} \tag{4.37}$$

3. Prediction step

This step predicts the state and covariance of each model based on its dynamics:

$$\begin{aligned}\hat{\mathbf{x}}^{0j}_{k|k-1} &= \mathbf{F}_j \hat{\mathbf{x}}^{0k}_{k-1|k-1},\\ \mathbf{P}^{0j}_{k|k-1} &= \mathbf{F}_j \mathbf{P}^{0j}_{k-1|k-1} \mathbf{F}_j^{\intercal} + \mathbf{Q}_j,\end{aligned} \tag{4.38}$$

where $\mathbf{F}_j$ is the state transition matrix for model j, $\mathbf{Q}_j$ is process noise covariance matrix for model j. For convenience, the subscript j substitute $M_k(j)$. The prediction $\hat{\mathbf{x}}^{0j}_{k|k-1}$ and its corresponding covariance matrix $\mathbf{P}^{0j}_{k|k-1}$ are computed for each model j.

4. Filtering step

After the prediction, the filtering step refines the state estimate using the new measurements:

$$\begin{aligned}\hat{\mathbf{x}}^{j}_{k|k} &= \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}^{0j}(\mathbf{z}_k - \mathbf{h}(\hat{\mathbf{x}}^{0j}_{k|k-1})),\\ \mathbf{P}^{j}_{k|k} &= \mathbf{P}^{j}_{k|k-1} - \mathbf{K}^{0j}\mathbf{H}(\hat{x}^{0j}_{k|k-1})\mathbf{P}^{0j}_{k|k-1},\\ \mathbf{K}^{0j} &= \mathbf{P}^{0j}_{k|k-1}(\mathbf{S}^{j})^{-1}\\ \mathbf{S}^{j} &= \mathbf{H}(\hat{x}^{0j}_{k|k-1})\mathbf{P}^{j}_{k|k-1}\mathbf{H}^{\intercal}(\hat{x}^{0j}_{k|k-1}) + \mathbf{R},\end{aligned} \tag{4.39}$$

where $\mathbf{S}^j$ is an innovation covariance matrix for model j. Note that for EKF, $\mathbf{h}(\cdot)$ is a nonlinear function and the matrix $\mathbf{H}$ is Jacobian matrix.

5. Mode-matched filtering

As shown on the Figure 10, renormalized estimates and covariances are used to compute likelihood functions corresponding to the r filters:

$$\begin{aligned}\Lambda_k(j) &= p\left[\mathbf{z}_k | M_k(j), Z^{k-1}\right],\\ j &= 1, \ldots, r.\end{aligned} \tag{4.40}$$

It can be also expressed as:

$$\begin{aligned}\Lambda_k(j) &= \mathcal{N}\left[\mathbf{z}_k; \mathbf{h}_k(\hat{\mathbf{x}}_{k|k-1}), \mathbf{S}^j\right]\\ j &= 1, \ldots, r.\end{aligned} \tag{4.41}$$

6. Mode probability update

Mode probability update is calculated based on the likelihoods obtained from the previous step:

$$\begin{aligned} \mu_k(j) &= P\Big\{M_k(j)|Z^k\Big\} \\ \mu_k(j) &= \frac{1}{c}\Lambda_k(j)\overline{c}(j) \quad j = 1, \ldots, r. \end{aligned} \tag{4.42}$$

Factor c is the normalization constant for the mode probability update:

$$c = \sum_{j=1}^{r} \Lambda_k(j)\overline{c}(j). \tag{4.43}$$

7. Estimate and covariance combination

In this step, estimates and corresponding covanriances are combined to yield a single estimate according to the mixture formula:

$$\begin{aligned} \hat{\mathbf{x}}_{k|k} &= \sum_{j=1}^{r} \hat{\mathbf{x}}_{k|k}^{j}\mu_k(j), \\ \mathbf{P}_{k|k} &= \sum_{j=1}^{r} \mu_j(k)\left\{\mathbf{P}_{k|k}^{j} + [\hat{\mathbf{x}}_{k|k}^{j} - \hat{\mathbf{x}}_{k|k}]\cdot[\cdot]^{\intercal}\right\}. \end{aligned} \tag{4.44}$$

This last step is not necessary for the recrusion because it is not part of the algorithm itself.

5 Architectural Overview of the Tracking Assessement Framework

The aim of this thesis is to analyse the quality of the estimates provided. This section provides an exploration of the simulation and estimation methodologies which were applied within the thesis. Four dynamic motion models, which were presented in Section 3, are used to generate ground-truth values and to simulate the measurements, for which the ranging principles from the Section 2 section are applied. Furthermore, the IMM algorithm from the 4.4 Section is implemented to estimate the position of the object, which will be evaluated in terms of quality. The general structure of the code developed in this thesis is grapically summarized by the Figure 12.

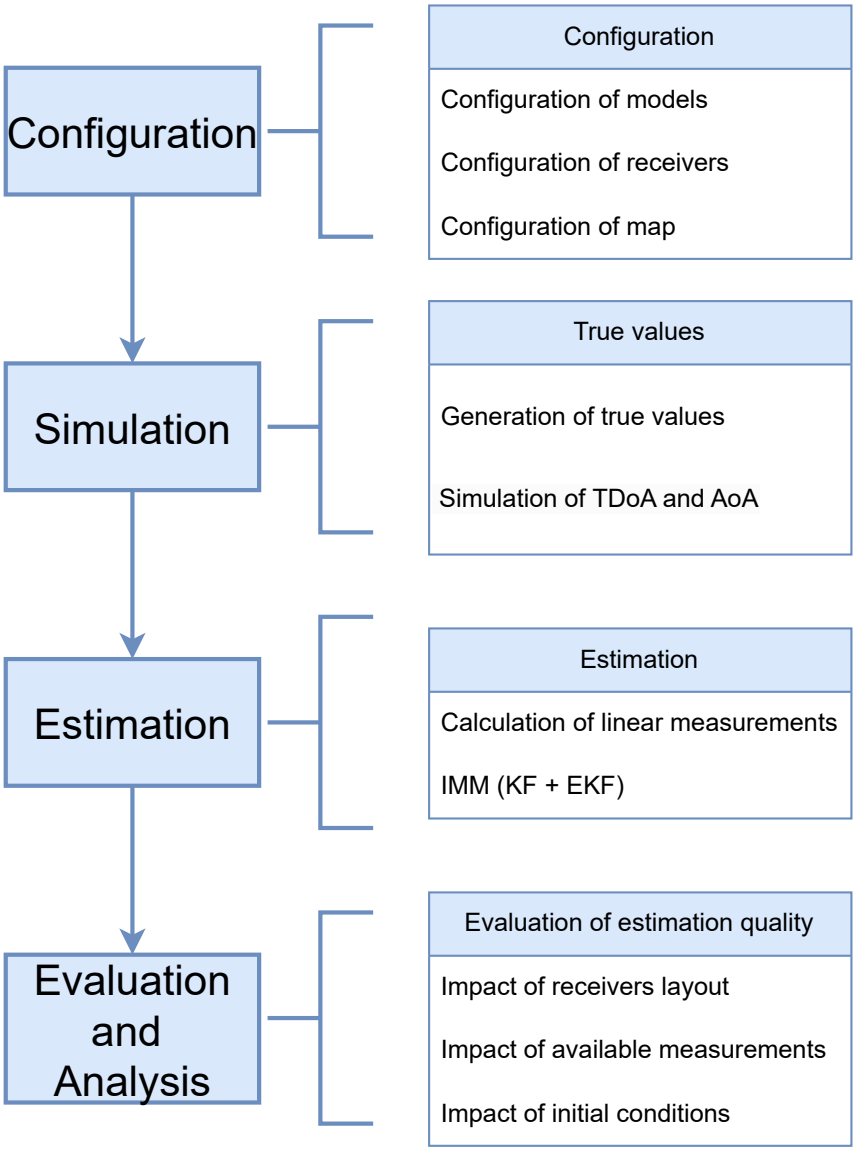


Figure 12: *Architecture of code*

The structure is divided into four main components, each playing a crucial role in accurately simulating and analyzing the behavior of multilateral tracking systems. All components are modular, allowing for the possibility of replacements. For instance, the part involving the simulation of true trajectories and generating measurements could be substituted with actual values and measurements obtained from real-world scenarios once the data assciation is resolved.

The configuration component of the framework offers a flexible approach to setting up dynamic models for both estimation and simulation, allowing for detailed customization of receiver characteristics and select the map where the quality of the estimation results will be displayed.

In this thesis, the trajectories to be estimated are generated to analyse the dynamic systems. This process employs a Hidden Markov Model (HMM) approach that utilizes a combination of predefined models decsribed in the Section 3 with configurations provided by the user.

The one of the main parts of this thesis is the implementation of the IMM algorithm for estimating the position of an object. This implementation involves calculation of mixing probabilities, mixing, mode-match filtering, mode probabilty update and estimate combination. The IMM algorithm is chosen for its flexibility to adapt various models and its proficient handling of uncertainties.

The final component of the framework enables the evaluation of estimation quality, leveraging a different sets of configuration that include the layout of receivers, initial conditions, and the availability of measurements. To assess the performance of the estimations, two distinct methods are implemented: the Root Mean Square Error (RMSE) and the Average Normalized Estimation Error Squared (ANEES). These approaches provide an analysis of estimation accuracy, enabling a deeper understanding of the system's performance under various scenarios. The RMSE and ANEES methodologies are further described in later sections.

Further sections will delve into the specifics of configuration setups, trajectory generation methods, the IMM implementation and analysis methodology of the obtained results under different scenarios.

5.1 Configuration module

The purpose of this section is to clarify the configuration of the models, receivers, and evaluation map used for quality estimation. The configuration framework is divided into three main parts: the configuration of the models, the configuration of the receivers, and the configuration of the map, which is used to evaluate the quality of the estimated states by the IMM algorithm.

The configuration of the models includes by default four models described previously:

NCV, DWPA, Singer and DWNA. Each model provides information about the object, with the DWPA and Singer models offering additional details about the acceleration of the object. Each model has a fixed structure as described in the Section 3. Parameters such as the sampling period T or power spectral density q can be adjusted to enhance the model's accuracy and make it more reflective of real-world scenarios. In addition, this section involves the definition of inital probabilities and transition matrices, which are necessary for the HMM and the IMM algorithm. Note that any number of models can be used within the IMM algorithm framework if the models have the same space dimension in the current implementation. It is only necessary to expand transition matrix and initial probabilities. For the purpose of this thesis, the models are paired as follows: NCV with DWNA and Singer with DWPA. Note that, the user can define new models if desired.

In the case of the receivers, several parameters are adjustable. Users can select any number of receivers according to their simulation needs. Each receiver has several configurable parameters:

- **Position**: Defined by coordinates.
- **Availability of AoA measurements**: The parameter allows the simulation of AoA measurements for the current receiver.
- **AoA noise characteristics**: The AoA noise is assumed to be Gaussian, allowing the user to modify its mean value and variance.
- **Availability of ToA**: The parameter allows the simulation of ToA measurement for the current receiver.
- **ToA noise characteristics**: Similar to AoA, the ToA noise is considered Gaussian, with adjustable mean value and variance.

These settings provide flexibility in how receivers are configured and contribute to the simulation's realism and adaptability to different scenarios. TDoA measurements critically depend on the geometric configuration of the receivers. This thesis explores several geometric layouts of receivers to evaluate their impact. One of the layouts is the so-called star layout, which is shown in Figure 13. In this layout, the reference receiver is the central point relative to which TDoA measurements from other receivers are calculated.

The last part of configuration framework involves setting up the evaluation map, which is defined as discretization of the tracking area into rectangular cells, where the estimation quality can be visualized efficiently. The map is defined within a 3-D plane, declared by its edges. The user have the flexibility to define the level of quantization of the map, which divides the space into distinct discretization cells. This allows for a detailed analysis of the estimation quality within each cells. This structured approach provides way to observe

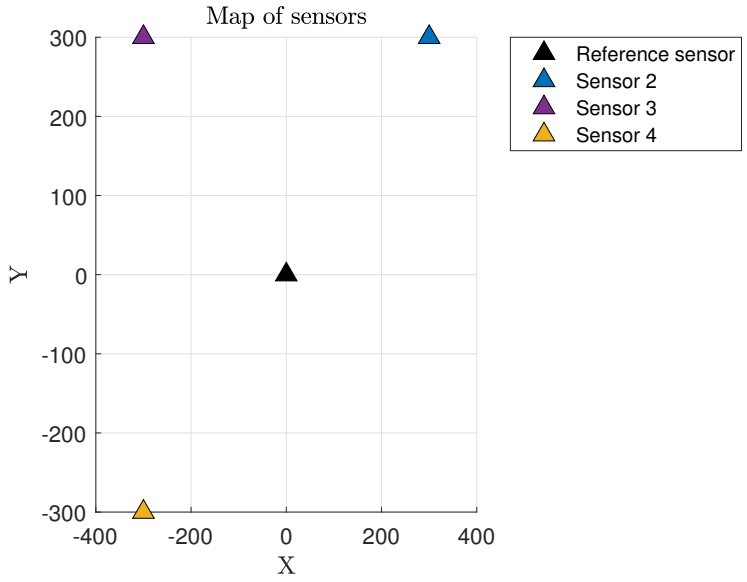


Figure 13: *Star layout of the receivers*

and evaluate the quality of estimation under various conditions. The example of splitting the space into distinct cells is shown in Figure 14.

The configuration also involves the definition of the starting positions from which individual simulations of trajectories will start. The simulated trajectories should ideally cover the entire map as densely as possible. The choice of initial conditions includes multiple variants to help to calculate estimation quality for whole map. This discretization

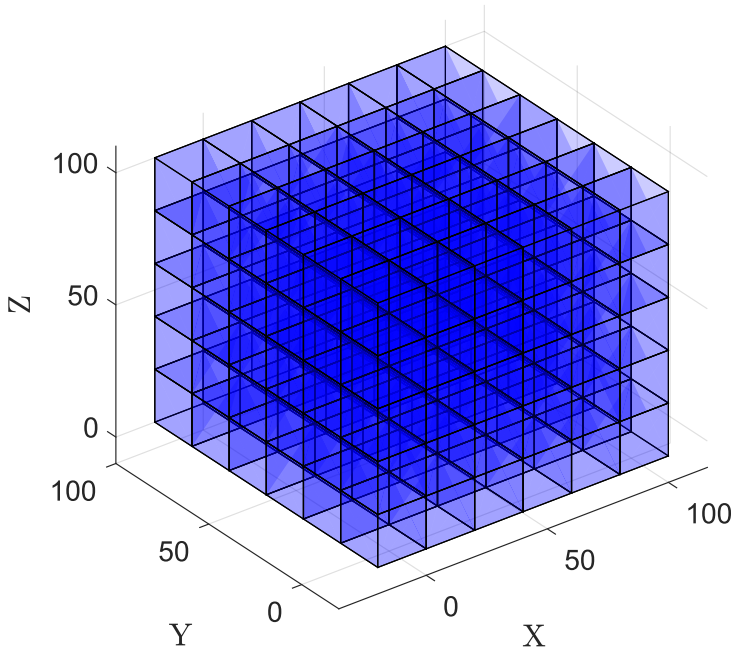


Figure 14: *Example evaluation map*

serves as the basis for the visualization, which is further discussed in Section 5.4.

5.2 Simulations of true values and measurements

Due to the lack of real-world measurement data suitable to perform large-scale estimation quality assessement, it is necessary to obtain simulated data. The true data of positions, velocities and, where applicable, accelerations are simulated according to the hidden Markov Model (HMM). Within the simulation framework, it is possible to choose between two pairs of models by default with the same dimensionality: NCV with DWNA and Singer with DWPA. The choice between the models depends on inital probabilities and a transition matrix, which are set by the user. The initial probabilities define the dynamic model being the active model at the beginning of the simulation. Transition matrix determines likelihood of switching between models within HMM setup. Additionally, user can specify the length of each trajectory simulation.

At each step, determined by the sampling period T, the HMM determines whether the transition between models occurs based on the transition matrix. If the transition is triggered, the simulation switches to the appropriate model, which defines the dynamics used to calculate the next state. This approach aims to mimic potential changes in motion dynamic that may occur in real-world scenario. The simulated trajectory for the Singer and DWPA models is illustrated in Figure 15. Figure 16 demonstrates which model is active at each time step. The parameters for this simulation are specified as follows: For the Singer model, the maneuver time constant is $\tau_m = 10$ and the process noise standard deviation is $\sigma_m = 0.1$. For the DWPA model, the noise standard deviation is set to $\sigma_v = 1$. For both models, sampling period is set to $T = 0.1s$.

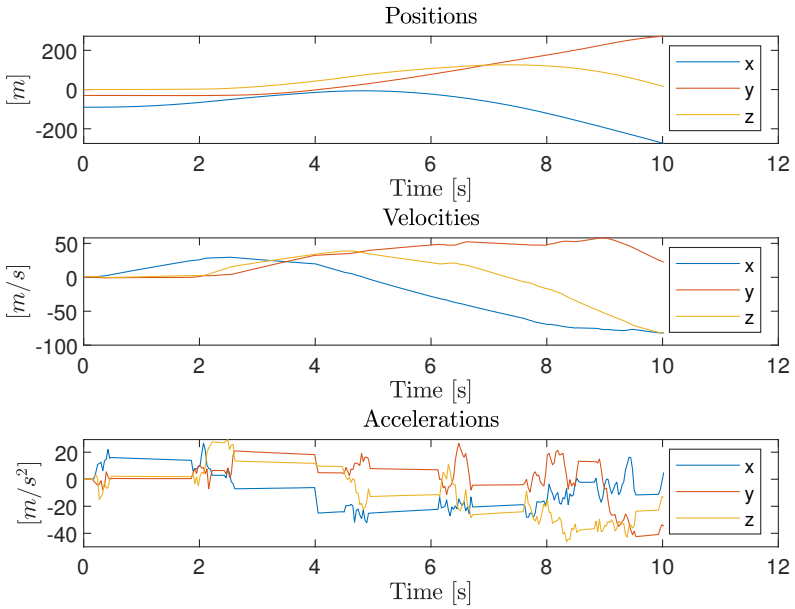


Figure 15: *Generated trajectory using HMM with Singer and DWPA models*

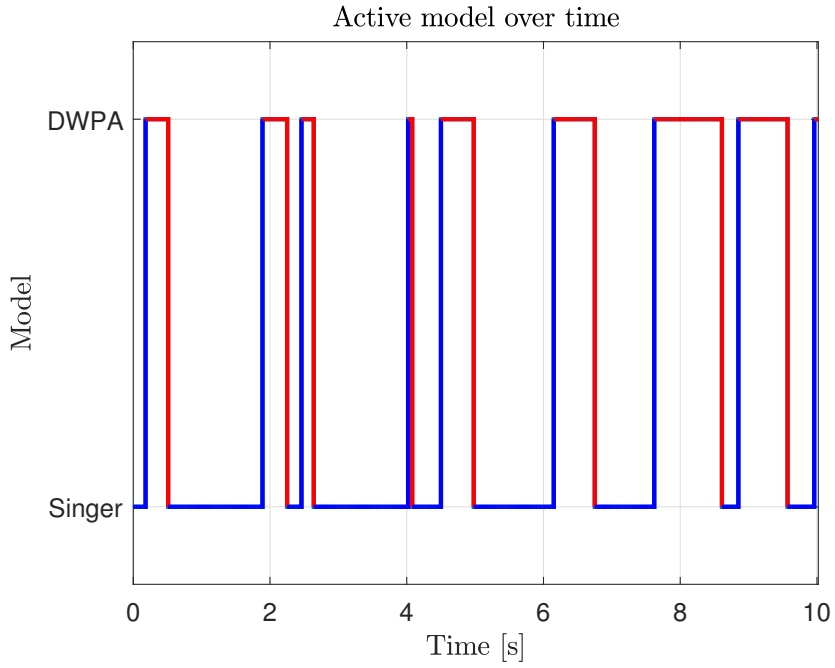


Figure 16: *Active model over time*

Once the true data for positions, velocities, and accelerations have been generated through simulation, the next step involves creating the measurements of TDoA and AoA. TDoA measurements are calculated based on the true distances obtained from the simulation, considering the predefined positions of various receivers according to equation (4.9). Similarly, the AoA measurements are simulating by determining the angle of incoming signals relative to the position of each receiver according to equation (2.17).

5.3 Estimation framework

With the generated measurements, the estimation of trajectory can be executed using the IMM algorithm explained in Section 4.4.1. The algorithm effectively combines multiple dynamic models to adapt various motion behaviors. In this section, the principal methods used in each scenario will be further analyzed and explained in detail.

TDoA and AoA measurements are later utilized by the IMM algorithm to estimate the state of the object. Processing of such measurements, however, can be approached in several ways. For the Kalman filtering framework (i.e., linear filtering), it is necessary to transform these measurements at each time step into a 3-D position, as described in detail in Section 2.1.5. In contrast, the extended Kalman filtering framework (i.e. nonlinear filtering) which handles the non-linear measurements directly.

The effectiveness of these approaches across various scenarios will be compared in a later section, providing insights into their relative efficiencies under different conditions.

5.3.1 Estimation using the IMM algorithm

The IMM algorithm is implemented as it was described in the Section 4.4.1. This implementation accommodates various models, but it requires initial probabilities, a transition matrix, and initial conditions of each used model for both state estimates and their covariance matrices. These may either be the same as used for the simulation, or not, depending on the user needs.

For fair comparisons, the initial conditions for the state estimation are set based on the first few measurements of TDoA or AoA, depending on the model used. For models with nine state dimensions, such as Singer and DWPA, the first three measurements are utilized. For models with six state dimensions, such as NCV and DWNA, only the first two measurements are needed. These initial measurements are transformed into position vectors as outlined in Section 2.1.5 and used to provide the initial position estimates. To complete the entire initial state vector, initial estimates of velocity and acceleration are also required. For each spatial dimension, these are derived using the following transformation matrices: For the NCV and DWNA models, the transformation matrix is defined as:

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ \frac{-1}{T} & \frac{1}{T} \end{bmatrix}, \tag{5.1}$$

which models a simple velocity-based transition and estimates the velocity as a function of position changes during the sampling period. For the Singer and DWPA models, which account for acceleration, the transformation matrix is:

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 1 \\ \frac{-1}{2T} & 0 & \frac{1}{2T} \\ \frac{1}{T^2} & \frac{-2}{T^2} & \frac{1}{T^2} \end{bmatrix}, \tag{5.2}$$

This matrix models acceleration by relating it to the change in velocity over time. Note that the matrix $\mathbf{A}$ is defined only for one dimension. Assume the generic state that consists of position, velocity and possibly acceleration in the case of Singer and DWPA:

$$\xi_k = \begin{bmatrix} \xi_k & \dot{\xi}_k & \ddot{\xi}_k \end{bmatrix}^\mathsf{T}, \tag{5.3}$$

The inital state mean for generic variable, i.e. the initial state mean for one dimension, is calculated using:

$$\xi_{\text{init}} = \mathbf{A}\mathbf{z}, \tag{5.4}$$

where $\mathbf{z}$ denotes measurement transformed from TDoA and AoA values into the position

vector as it is described in Section 2.1.5.

The initial covariance matrix is also calculated analytically using the first three measurements, but this time the covariance matrix of measurement $\mathbf{R}$ is used together with the matrix $\mathbf{A}$ defined above:

$$\mathbf{P}_{\text{init}} = \mathbf{ARA}^{\intercal}, \tag{5.5}$$

where $\mathbf{R}$ is the covariance matrix of measurements for one dimension and for the first two or three time steps depending on whether NCV and DWNA or Singer and DWPA is used.

For each model, both the initial mean vector and covariance matrix are computed by propagating the measurement uncertainties through the transformation matrices $\mathbf{A}$. Further details are omitted for simplicity.

The IMM algortihm allows to handle a varienty of scenarios with different number and types of measurements. As previously mentioned, the IMM can process both linear measurements by emplying the KF equations and nonlinear measurements using EKF.

Continuing the example from Section 2.1.5 the coresponding estimation using the EKF with three available TDoAs and four AoAs using Singer and DWPA models is demonstrated in the Figure 17.

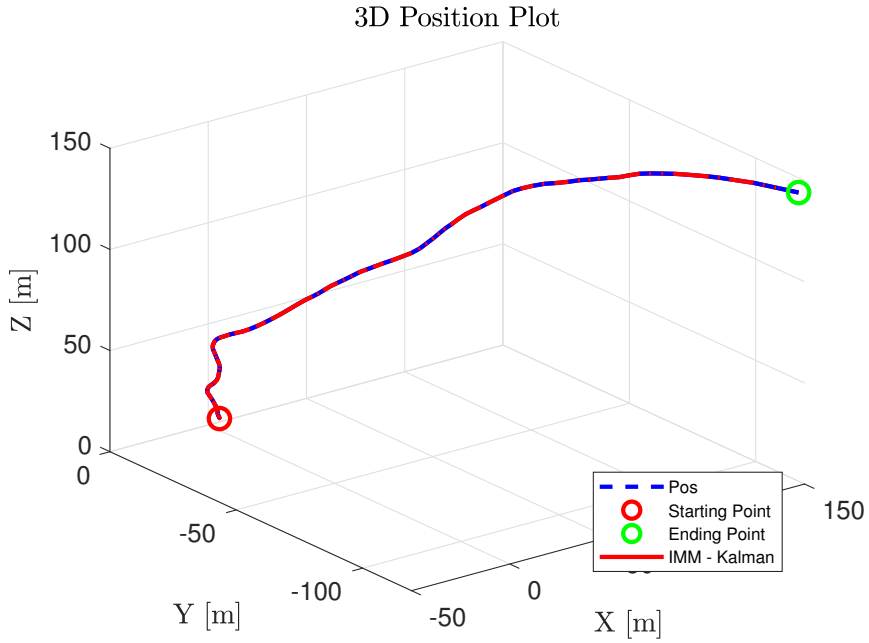


Figure 17: *IMM algorithm estimation*

5.4 Visualization module

To ease the assessement of the results, a visualization module analyzes each generated trajectory and the corresponding position estimates. As previously mentioned, quantization is utilized as a configuration parameter. This setting ensures that the space is divided into distinct sections where the evaluation of estimation quality takes place.

Two different performance metrics are utilized: Root Mean Square Error (RMSE) and Average Normalized Estimation Error Squared (ANEES). The estimation quality with RMSE is assessed within each quantized section of the space. For the purpose of this thesis, the RMSE is defined as the square root of the mean value of squared residuals. For a given cell:

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{N_{\mathrm{cell}}} \left\| \hat{\mathcal{X}}^i - \mathcal{X}^i \right\|^2}{N_{\mathrm{cell}}}}, \tag{5.6}$$

where $\mathcal{X}^i$ represents true position of emitter, $\hat{\mathcal{X}}^i$ is its corresponding estimated position of the emitter at the same time step and $\|\cdot\|$ denotes the Euclidean norm. The symbol N_{cell} is the number of ($\hat{\mathcal{X}}^i$, $\mathcal{X}^i$) pairs where the true position $\mathcal{X}^i$ belongs to the specific cell. The RMSE quality is therefore related to a specific cell, which is created by discretizing the space.

The ANEES expresses how the confidence the filter has in its estimates matches reality. An ANEES value of one indicates consistency, suggesting that the filter's error covariance matrix matches the true error covariance. Values greater than one indicate that the filter is overconfident with its estimate, which means that the estimation error covariance is "smaller" than it should be. Lastly, values less than one mean underconfidence of the filter's predictions. The formula of ANEES is given as:

$$\mathrm{ANEES} = \frac{1}{N \cdot n_x} \sum_{k=1}^{N} (\hat{\mathcal{X}}^i_{e,k|k} - \mathcal{X}^i_{e,k})^\intercal (\mathbf{P}^i_{k|k})^{-1} (\hat{\mathcal{X}}^i_{e,k|k} - \mathcal{X}^i_{e,k}), \tag{5.7}$$

where n_x is dimension of the position vector, which is always 3 since movement occurs in 3D space, and $\mathbf{P}^i_{k|k}$ is the estimated error covariance matrix and and N is in this case the number of time instants in one specific Monte Carlo sample. Unlike RMSE, which is calculated per cell, ANEES is applied to a single sample to provide comprehensive information about that specific sample.

One example of visualization of the estimation quality for a given map is illustrated in Figure 18. Note that the layout of the receivers for this specific example is illustrated in Figure 19. The discretization of the space enable to assess the estimation quality within each cells. It is thus possible to determine for which areas of the map provided estimate is accurate. The estimation quality is color-coded as it is shown in Figure 18. For this

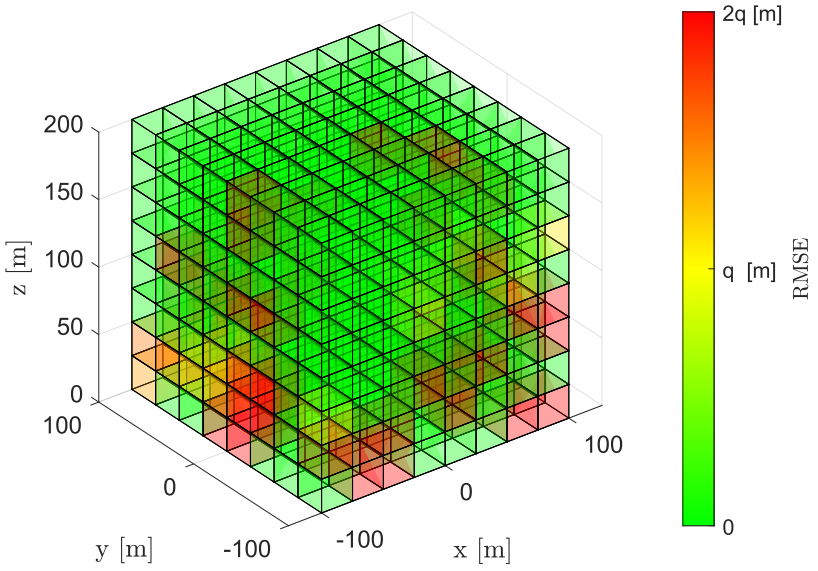


Figure 18: *Visualization of the estimation quality within the evaluation map*

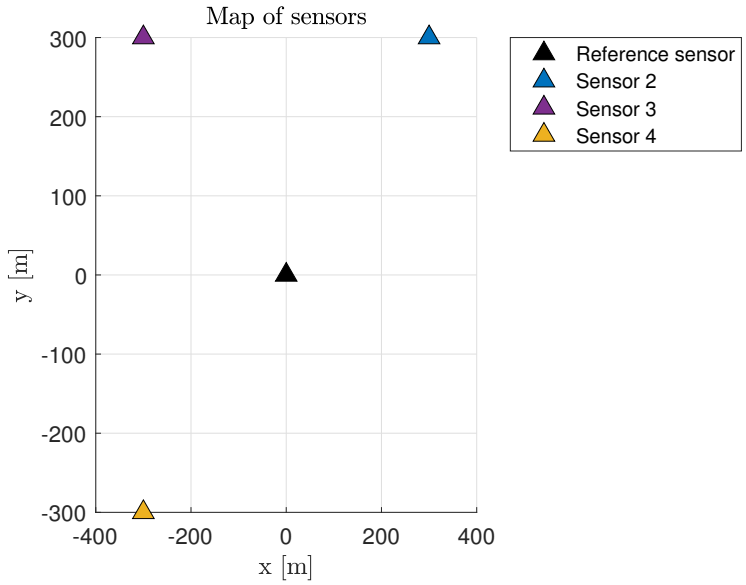


Figure 19: *Layout of the receivers for the specific example*

example RMSE is used. The color gradient transitions from red color, which indicates poor quality of estimates, throught the yellow for moderately accurrate estimates to green color that express high precision of the estimate. Moreover, each segment of the space is colored based on the specified treshold. This threshold can be chosen arbitrarily by the user and it needs to be set manually. The automatic scaling of the errors based on the worst and best error cases is not feasible because of the possibility of excessively large errors, that could obscure all other errors. Therefore, the maximum error will be saturated by the user. For the purposes of this example, the treshold is set to the value of quantization

due the quantization is smooth in this example. An estimate is considered accurate if the RMSE places the estimate in a circle centered at the true position with a diameter equal to the quantization value. If the estimate lies outside this circle but inside a circle with a diameter twice the quantization, it is considered moderately accurate. Estimates lying outside the circle with twice the diameter of the quantization are considered as inaccurate. The example of such a coloring for one generated trajectory and its estimate is demonstrated in the Figure 20, where it can be seen that the estimation quality can be considered good over entire scenario. Note that the threshold is another parameter that can be set by the user, and this parameter determines the boundaries for moderate and poor quality. The example of the poor estimated position of the emitter and its

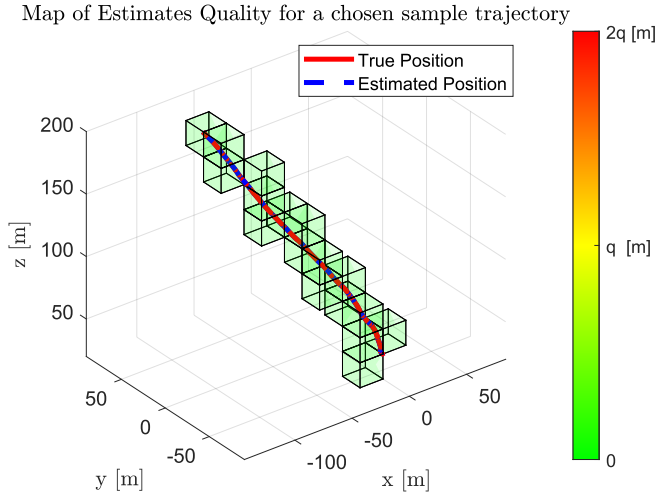


Figure 20: *Visualization of the estimate quality for one trajectory*

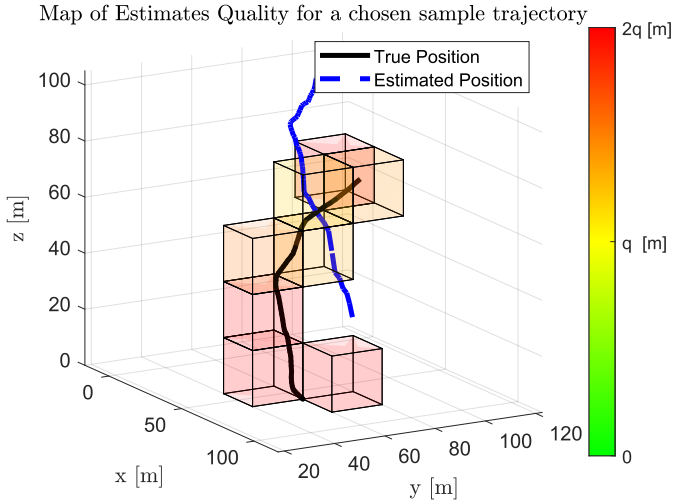


Figure 21: *Visualization of the poor estimation quality for one trajectory*

visualization of quality is shown in the figure 21.

The visualization in 3-D is not informative for the assessing the quality of estimates across the entire defined map. For this purpose, it is possible to select a specific vertical-level and visualize the quality in 2-D space at the particular level. The estimation quality at a specific level $z = 0$ of the map illustrated in Figure 18 is shown in Figure 22.

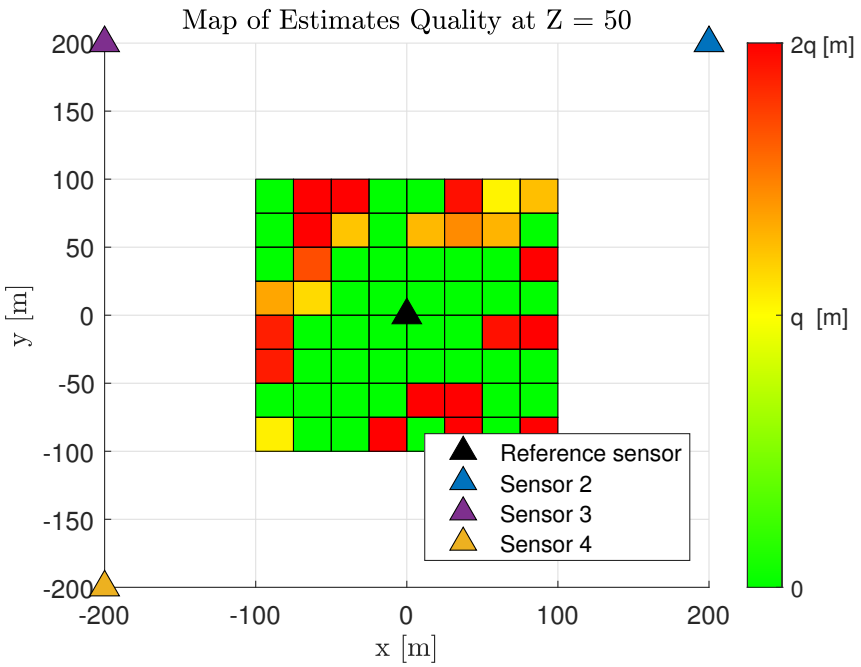


Figure 22: *Visualization of estimation quality in 2D for specific layer*

Alternatively, the user can also visualize a specific layer along the z or y axis. This is shown in the figure 23.

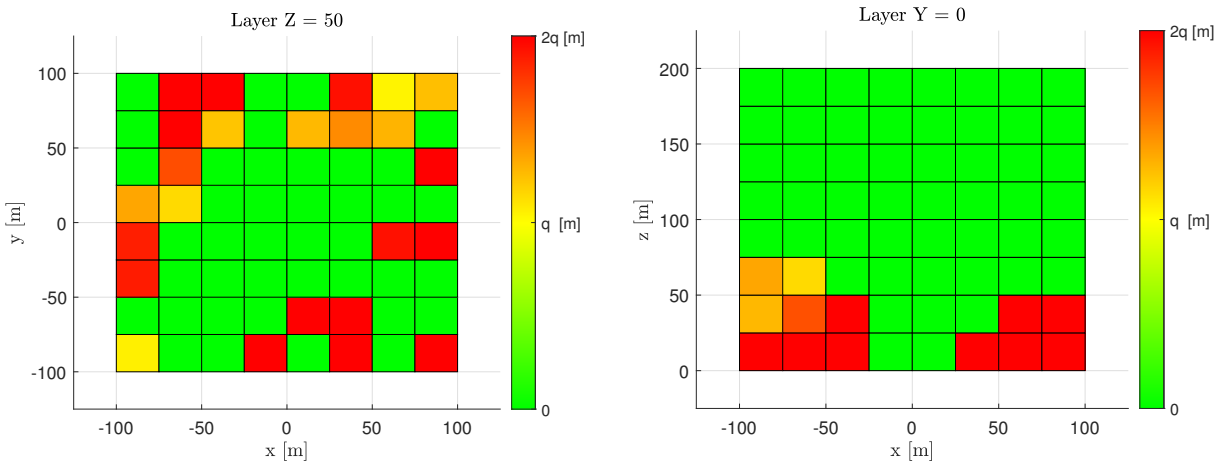


Figure 23: *Visualization of estimate quality in 2D for specific layers of z and y*

The user can also select a specific Monte Carlo simulation sample i.e., a trajectory to be tracked and its corresponding reacking results and visualize its trajectory for each axis, as well as the R(M)SE and (A)NEES quality metrics. For an example trajectory, this is

visualized in Figures 24 and 25. Note that for a single trajectory, the terms "mean" and "average" might not be exact, so they are given in parentheses.

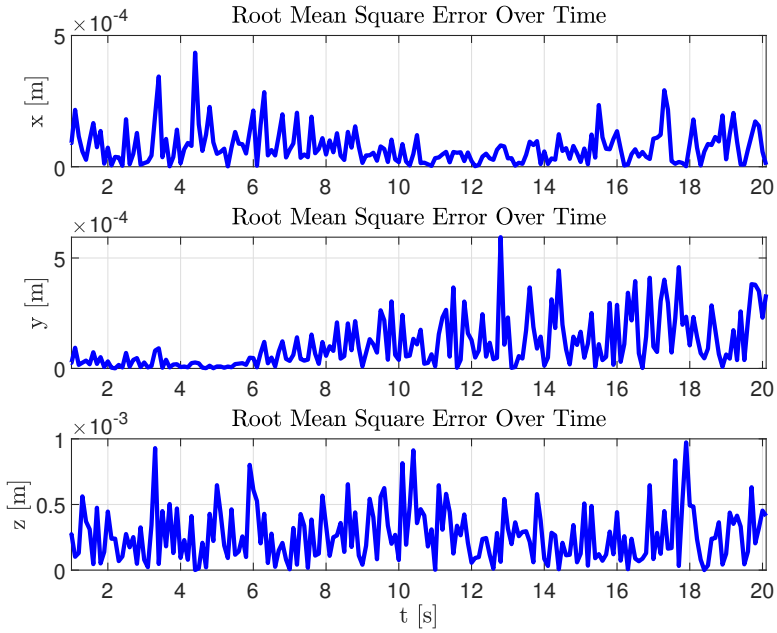


Figure 24: *RMSE metric for each axis for the example trajectory*

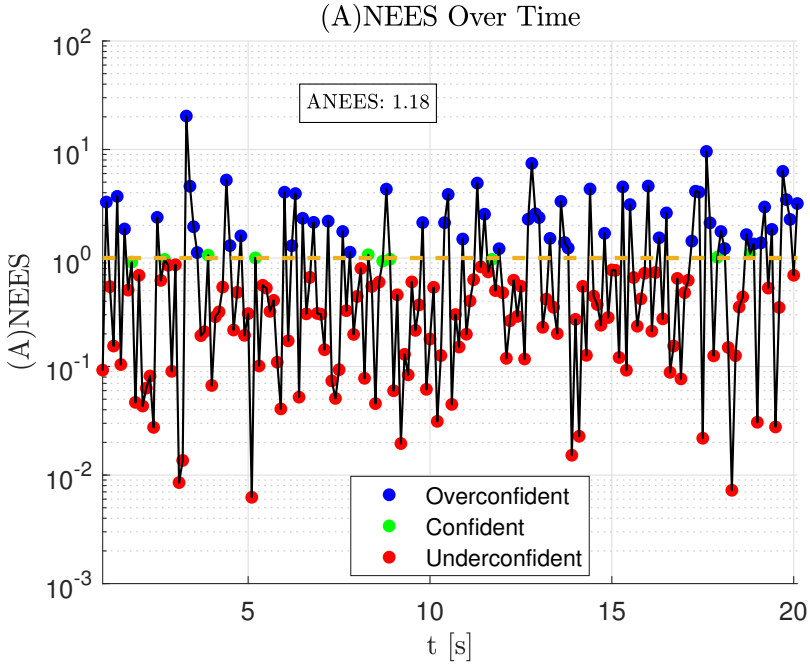


Figure 25: *ANEES metric for the example trajectory*

The visualization techniques presented in this chapter were developed with the aim of providing the possibility of deeper analysis of the impact of various factors, such as the layout of the receivers or availability of the measurements.

6 Simulation Experiments

In this section, simulation experiments are conducted and analyzed using the methodologies and techniques previously explained. As was described in Section 5.1, the user has nearly unlimited possibilities to setup simulations. For the purpose of this thesis, however, only a few scenarios are shown. These scenarios are chosen to demonstrate the impact of several configurations on the quality of the estimation and to show the diversity of possible configurations.

For the illustrations presented in this thesis, a maximum of four receivers is utilized. The analysis will include two experiments. For the experiment, the evaluation map is defined along the x-axis from -5000 to 5000 meters, the y-axis follows the same range and the z-axis range is declared from 0 to 10000 meters. The quantization for all axis is defined as $q = 500\ m$. The parameters of the individual sensors are as follows:

- Reference sensor:
  - Position: $(0, 0, 0)\ m$.
  - AoA measurement noise characteristics: $\mathcal{N}(0, \sigma^2_{\text{AoA}})$, where $\sigma_{\text{AoA}} = \frac{3\pi}{180}\ rad$.
  - ToA measurement noise characteristics: $\mathcal{N}(0, \sigma^2_{\text{ToA}})$, where $\sigma_{\text{ToA}} = 2 \times 10^{-10}\ s$.
- Sensor 2:
  - Position: $(-10483.39, 15093.59, 3.24)\ m$.
  - AoA measurement noise characteristics: $\mathcal{N}(0, \sigma^2_{\text{AoA}})$, where $\sigma_{\text{AoA}} = \frac{3\pi}{180}\ rad$.
  - ToA measurement noise characteristics: $\mathcal{N}(0, \sigma^2_{\text{ToA}})$, where $\sigma_{\text{ToA}} = 1 \times 10^{-10}\ s$.
- Sensor 3:
  - Position: $(14472.85, 7020.92, 53.89)\ m$.
  - AoA measurement noise characteristics: $\mathcal{N}(0, \sigma^2_{\text{AoA}})$, where $\sigma_{\text{AoA}} = \frac{3\pi}{180}\ rad$.
  - ToA measurement noise characteristics: $\mathcal{N}(0, \sigma^2_{\text{ToA}})$, where $\sigma_{\text{ToA}} = 2 \times 10^{-10}\ s$.
- Sensor 4:
  - Position: $(-3148.59, -18307.34, 213.20)\ m$.
  - AoA measurement noise characteristics: $\mathcal{N}(0, \sigma^2_{\text{AoA}})$, where $\sigma_{\text{AoA}} = \frac{3\pi}{180}\ rad$.
  - ToA measurement noise characteristics: $\mathcal{N}(0, \sigma^2_{\text{ToA}})$, where $\sigma_{\text{ToA}} = 3 \times 10^{-10}\ s$.

For this configuration, the sampling period for models was set to $T = 0.03\ s$. Power spectral density of the NCV noise is $q = 100\ m^2 s^{-3}$. For DWNA and DWPA models, variance has been chosen as $\sigma_v = 30\ ms^{-1}$ and maneuver time constant for the Singer

model have been selected as $\tau_m = 10\ s$. Initial probabilities are given as $\begin{bmatrix}0.3 & 0.7\end{bmatrix}$ and transition matrix is defined as:

$$\mathbf{P} = \begin{bmatrix}0.95 & 0.05\\ 0.05 & 0.95\end{bmatrix}. \tag{6.1}$$

Time of the each Monte Carlo simulation is set to 10 s. Additionally, to explore the impact of measurement frequency on the estimation quality, scenarios considering only every tenth measurement, are added. A moderate threshold is defined as 20 meters, and any estimates with an RMSE exceeding 40 meters will be classified as poor. This layout, illustrated in Figure 26, represents a typical configuration used in ranging implementations. The Figure also shows the map that will be covered.

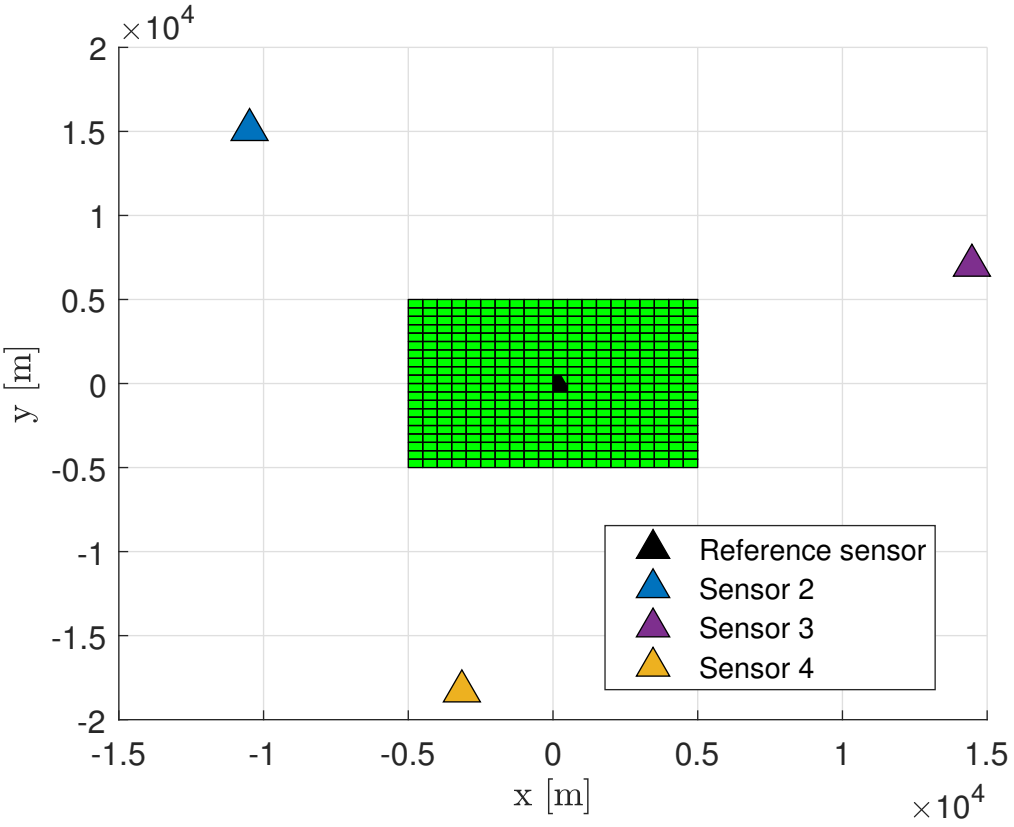


Figure 26: *Experiment: layout of receivers*

For each experiment six typical scenarios will be observed and compared to assess the corresponding tracking performance.

Scenarios:

- Scenario 1-3: Transformation of both TDoA and AoA measurements into the position vector $[x, y, z]$ at each time step, followed by the application of the Kalman filter:

  1. Three TDoA and no AoA Available.
  2. Three TDoA and four AoA Available.
  3. Two TDoA and three AoA Available.

- Processing non-linear TDoA and AoA measurements with the Extended Kalman Filter:

  4. Three TDoA and no AoA Available.
  5. Three TDoA and four AoA Available.
  6. Two TDoA and three AoA Available.

In the following section, the specific receiver layouts will be analyzed for all the above scenarios. The results will be presented through plots which were described in the Section 5.4. These comparisons highlight how the number of measurements and their processing method can affect the quality of estimation. Additionally, the effect of reduced measurement sampling frequency will also be investigated. Note that this experiment serves primarily to demonstrate the capabilities of visually representing the estimation quality. A much larger area would need to be covered in order to draw conclusions from the data.

6.1 Part I: Frequent measurements

In this section, experiment will be executed including comparison of each scenarios. The experiment provides comparison of the RMSE estimation quality for the specific layout and the impact of the alignment of the receivers. This experiment mainly focuses on demonstrating configuration options and plotting for analysis. Initially, the estimation quality is compared between a Kalman filter, which processes transformed TDoA and AoA measurements into a position vector at each time step, and an extended Kalman filter, which directly utilizes TDoA and AoA measurements. It is noted that while scenarios 1-3 are identical to scenarios 4-6, the only distinction lies in the handling of measurements. Individual scenarios are illustrated in Figures 29, 32 and 35. Although the quantization was chosen higher during generation, it was increased to $q = 1000\ m$ for rendering purposes.

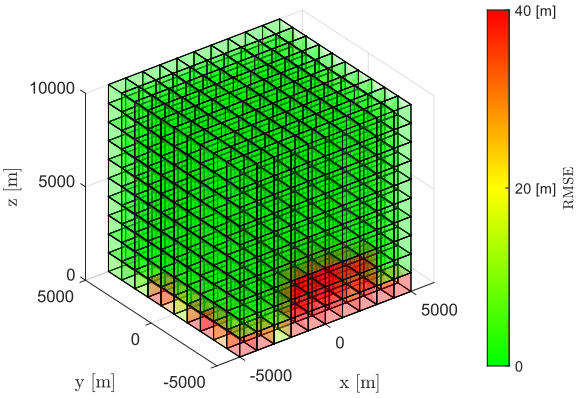


Figure 27: *Scenario 1: KF*

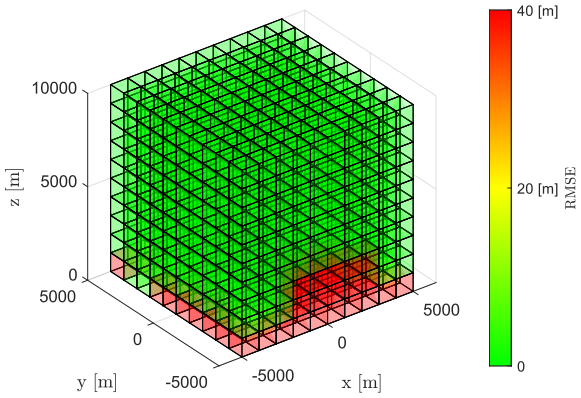


Figure 28: *Scenario 4: EKF*

Figure 29: *NCV and DWNA models - scenario 1 and 4 (three TDoA, no AoA)*

As the results show, the estimation quality is slightly better for scenarios where EKF with direct handling of TDoA and AoA measurements is used for estimation. The best estimate is provided for the case where all TDoA and AoA measurements are available. The graphs also show that the estimation quality is quite high for the chosen threshold of 40 m, which determines the sensitivity of coloring the cells.

For a more detailed analysis, the quantization resolution has been increased by reducing the cell size from 1000 to 500 meters for each axis. The resulting Figure 36, 37 and 38 illustrate the quality across all z-levels for all scenarios, with i.e., averaged RMSE over all z-values. A more detailed look at the results shows that the quality of the estimate is not as good as the 3-D plots suggest.

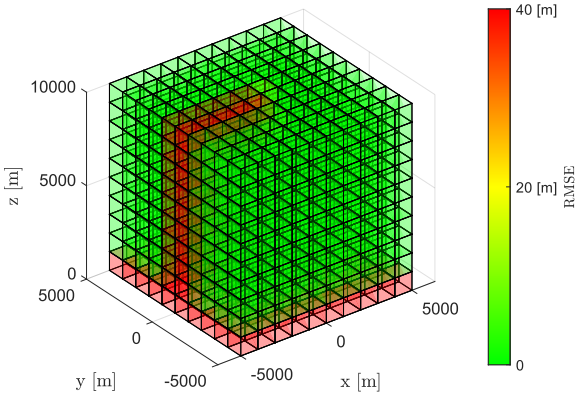


Figure 30: *Scenario 2: KF*

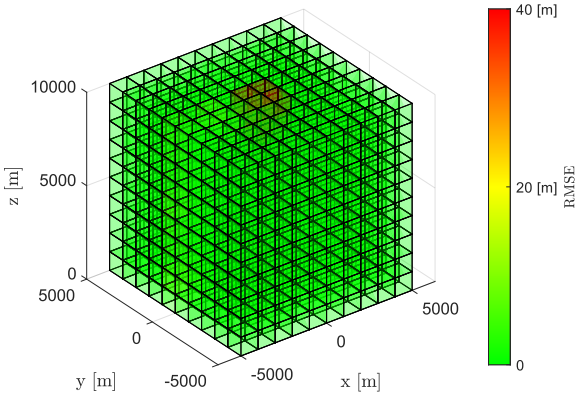


Figure 31: *Scenario 5: EKF*

Figure 32: *NCV and DWNA models - scenario 2 and 5 (three TDoA, four AoA)*

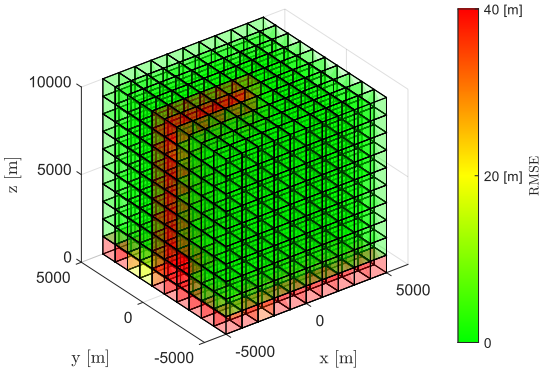


Figure 33: *Scenario 3: KF*

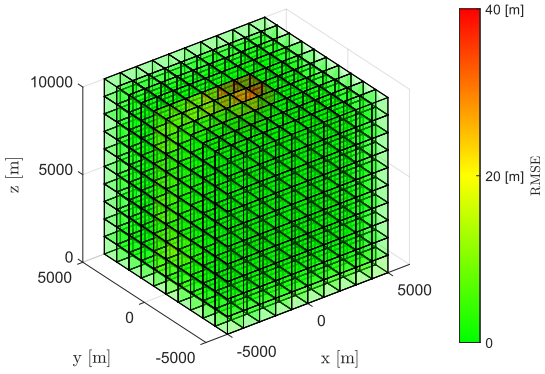
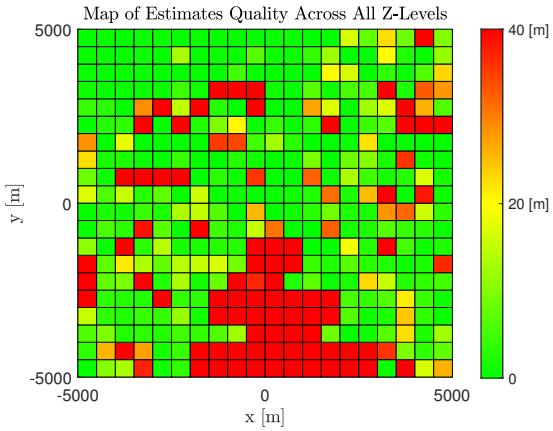


Figure 34: *Scenario 6: EKF*

Figure 35: *NCV and DWNA models - scenario 3 and 6 (two TDoA, three AoA)*



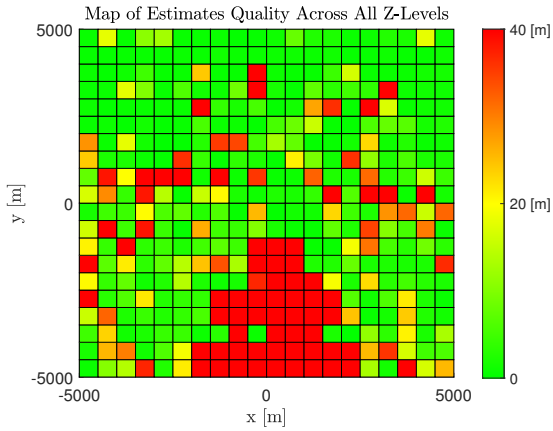


Figure 36: *Scenario 1 and 4: the quality across all z-levels*

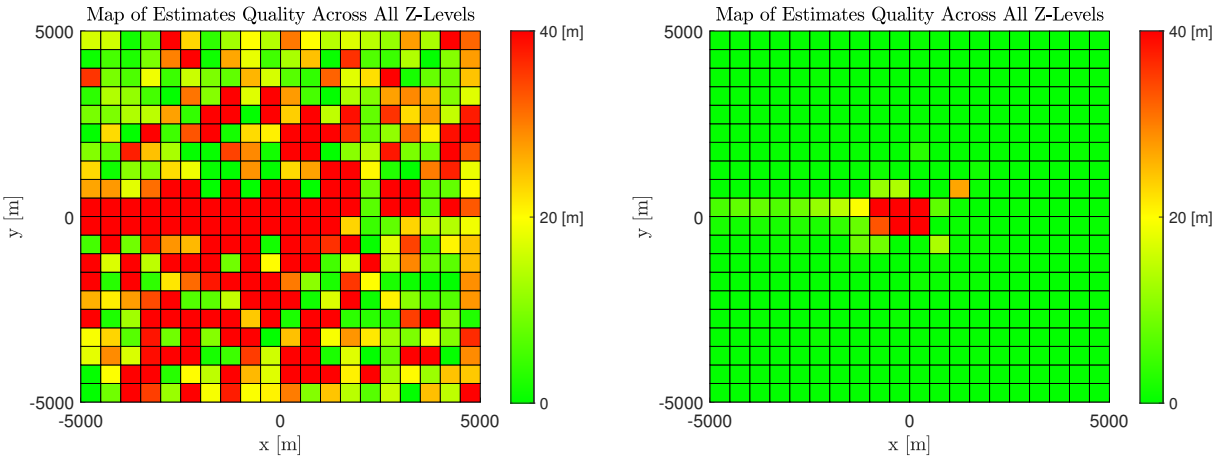


Figure 37: *Scenario 2 and 5: the quality across all z-levels*

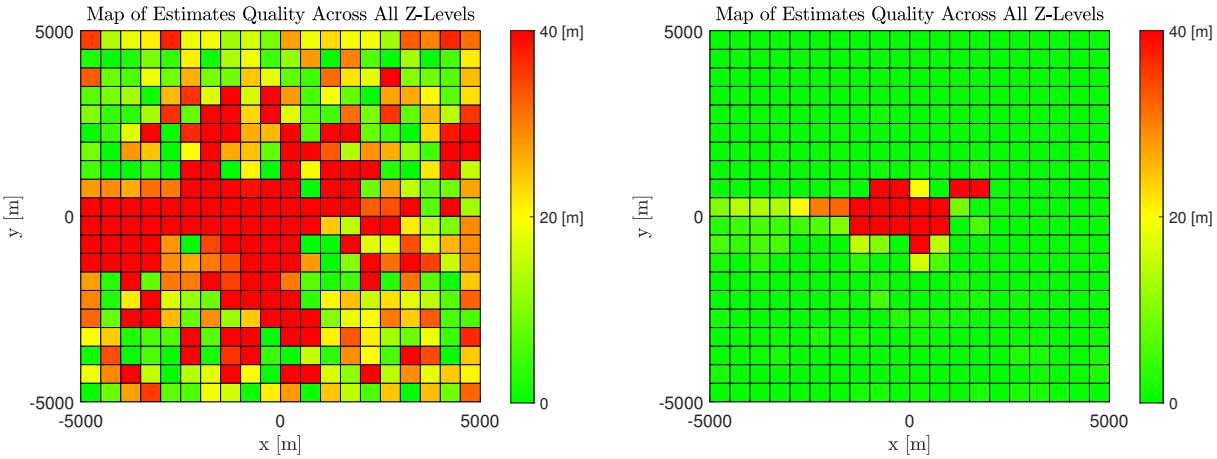


Figure 38: *Scenario 3 and 6: the quality across all z-levels*

The results for the Singer and DWPA model look similar. The scenario 3 and 6 are compared in Figure 41 and corresponding Figure 42 illustrates the quality across all z-levels these two scenarios.

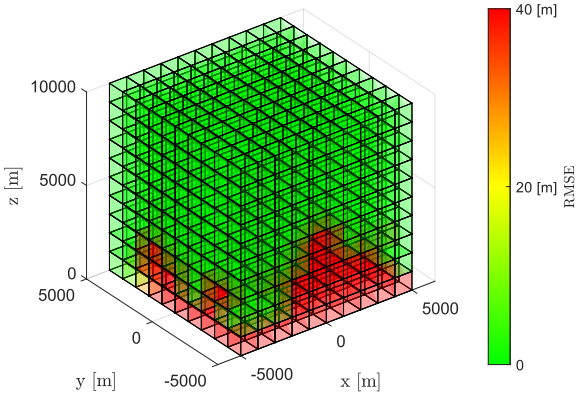


Figure 39: *Scenario 1: KF*

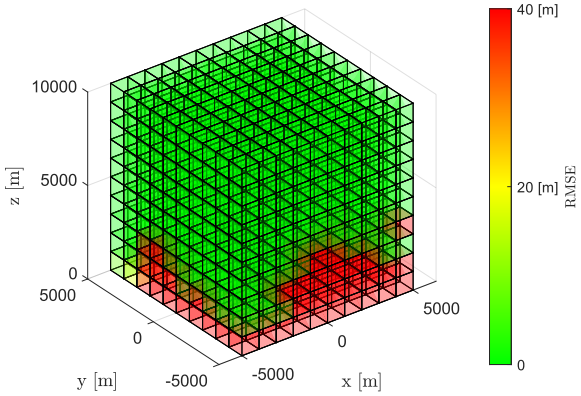


Figure 40: *Scenario 4: EKF*

Figure 41: *Singer and DWPA models - scenario 3 and 6: (three TDoA, no AoA)*

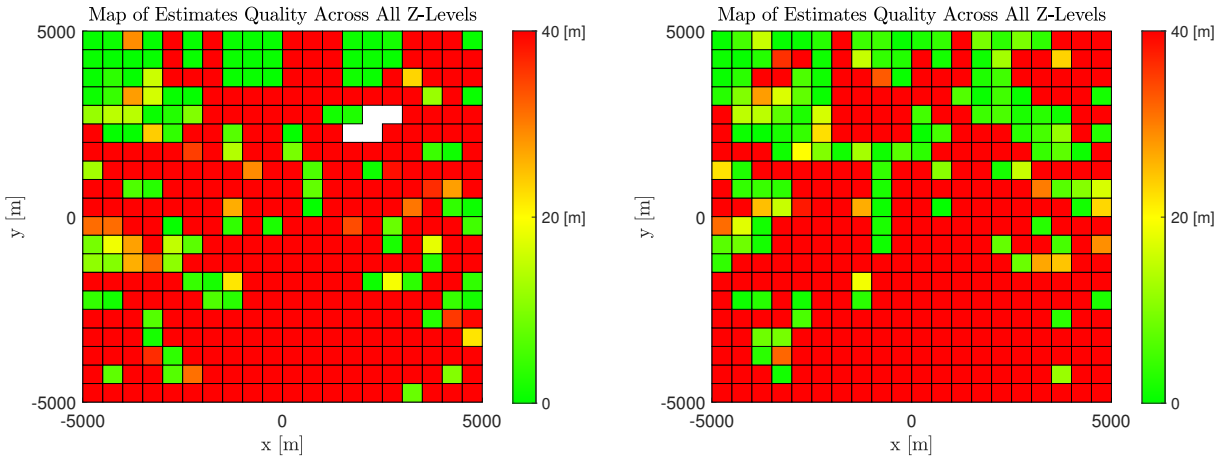


Figure 42: *Scenario 1 and 4: the quality across all z-levels*

Note that for the 2D analysis, the RMSE is averaged over all levels of z. The 2D plot serves as a complement to the 3D representation and clearly illustrates that the frequency of incorrect location estimates in the lower part of the map far exceeds the frequency of correct estimates in the upper part. As a result, the resulting 2D plot looks almost entirely red. This implies that an assessment of the quality of the estimates with respect to multiple evaluations is necessary for the evaluation.

6.2 Part II: Scarse measurements

In the second part of this experiment, only every tenth measurement will be utilized for updating the KF and EKF. During the other steps, the IMM will rely on the provided model dynamics and previously calculated values, with measurement updates occuring only at each tenth time step. For clarity, two scenarios will be demonstrated: Scenario 1 and Scenario 4. Results for these scenarios are shown in Figures 45 for NCV and DWNA models.

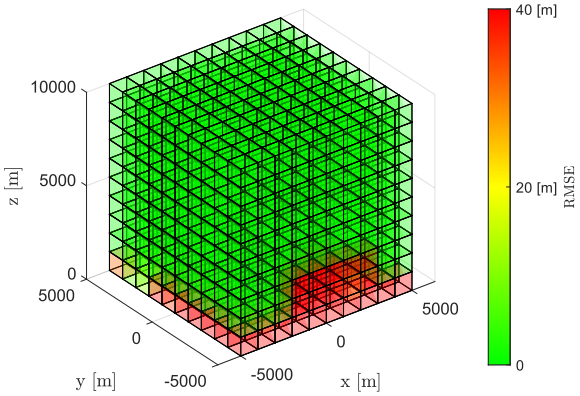


Figure 43: *Scenario 1: KF*

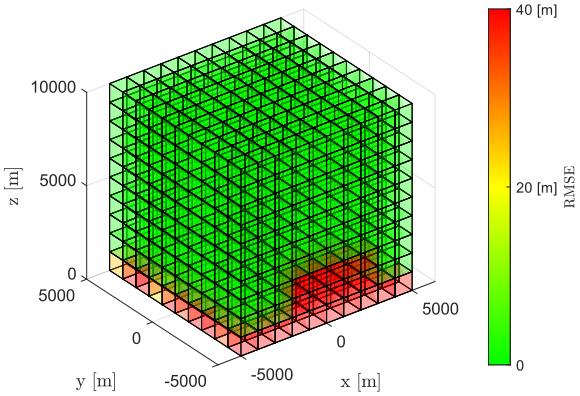


Figure 44: *Scenario 4: EKF*

Figure 45: *NCV and DWNA models - scenario 1 and 4 (every tenth measurement is used)*

The figures seem to show that although the estimation quality has dropped with the frequency of measurements, as could one expect. For comparison, the 2-D graphs averaged across all z-values are illustrated in Figures 46.

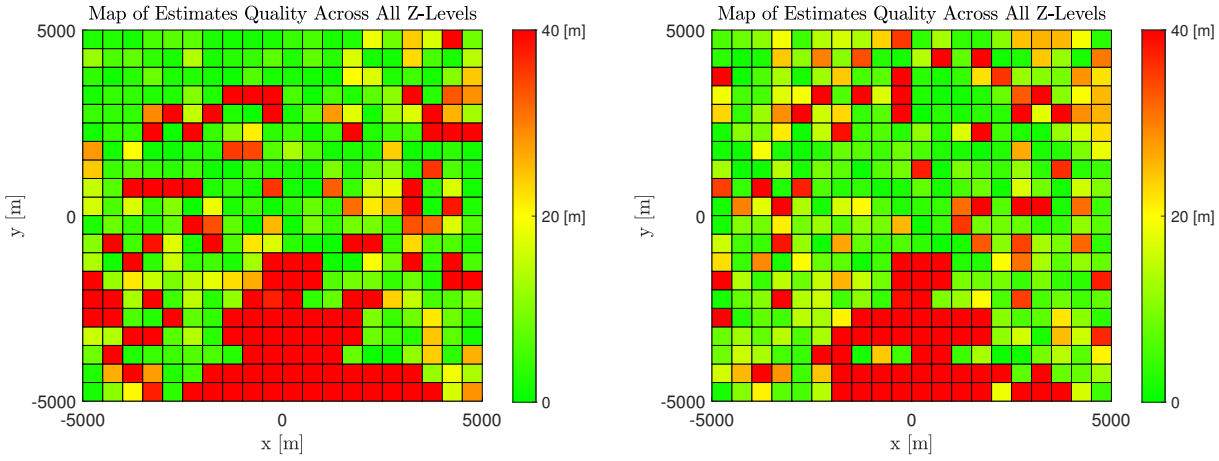


Figure 46: *Scenario 1 and 4: the quality across all z-levels (every tenth measurement is used)*

These results better reflect that the quality of the estimate has declined compared to previous Figures 36.

6.3 Experiment conclusion

These experiments demonstrate the capabilities of the implemented software, which allows for flexible configuration of the number of receivers and measurements. That is, the software enables users to create maps depicting estimation quality of any size with customizable quantization. Results can be analyzed in both 3-D and 2-D formats. Users have the option to select specific z-levels or y-levels for plotting or, alternatively, to analyze averaged values across all (e.g. z) levels. While individual Monte Carlo sample analyses are possible and detailed in Section 5.4, they are not presented in this section due to the extensive number of trajectories (over 9000) required to represent such a large map effectively. The experiment serves merely as an illustration of the capabilities of the implemented software. It can be seen that a more extensive experiment (using, e.g., a larger map) would be required for answering many practical questions, such as how to receiver's layout. noise parameters or parameters of the motion model influence the tracking quality. Such experiments are, however, outside the scope of this thesis. However, the experiment allows for the following observations:

- In the first experiment, specifically scenarios 1, 2 and 3, which employ the KF, the low profile estimates are not provided in high quality. In scenario 4, which utilizes only TDoA measurements, a similar issue is observed. However, incorporating AoA measurements into the EKF resolves this problem.
- When using the KF with transformed measurements in the first experiment, inaccurate estimates consistently occur in the same direction for scenarios 2 and 3. This issue likely arises from AoA ambiguity, particularly at the specific angle where the four-quadrant arctangent switches from -180 to 180 degrees. This effect is compensated for in the EKF, but not in the solutions involving nonlinear least squares that are utilized in the KF.
- Scarse measurements result in a decrease in the estimation quality, although the decline is not drastic.

7 Conclusion

This thesis explored the application of multilateration technique for tracking maneuvering objects. The objective of this thesis is to evaluate the accuracy of object position estimation within a specific map. For this evaluation, an object tracking algorithm employing radar multilateration was utilized, enabling a flexible configuration of individual receiver units regarding their quantity, positioning, and measurement availability.

The first part of the thesis was devoted to the theoretical foundations, in particular: ranging, dynamic object motion models and state estimation. For ranging, geometric principles of positioning were explored. Terms such as time of arrival, time difference of arrival and angle of arrival were explained and the mathematical formulas crucial for simualtion were defined. Subsequently, the thesis delved into dynamic object motion models, where models such as the nearly constant velocity and discrete white noise acceleration were detailed, along with more complex models like the Singer and discrete Wiener process acceleration that model acceleration in addition to position and velocity. Further, the state estimation process were explored and the Interacting Multiple Model algorithm were utilized.

Subsequently, the tracking assessement framework was implemented in MATLAB. It consists of five main parts: configuration, ground-truth value generation, measurement simulation, estimation process, and evaluation of estimation quality. In the configuration part, the user can select various models and parameters, choose the number of receivers, determine their positions, specify the availability of individual measurements for each receiver, define the noise characteristics of the measurements, and select the evaluation map in which the estimation quality will be evaluated. The next part of the program is dedicated to generating ground-truth values. These true values are later used to simulate time difference of arrival and angle of arrival measurements. In the estimation section, the Interacting Multiple Model algorithm is implemented for state estimation using Kalman filter or extended Kalman filter based on the handling of linear or non-linear measurements respectively. The outcomes of the estimation process are further analyzed in the last part of the program, which is devoted to evaluating the results. The estimated values are compared with the true trajectories, and the results are illustrated through a map with color-coding according to chosen evaluation metrics, which was the root mean squared error. The average normalized estimation error squared was used as well for evaluating individual Monte Carlo samples. The framework was designed with separate modules where each module consists of inputs and outputs that allows easy substitution. Thus, each module can operate on its own by providing the appropriate inputs. As a result, real data could be used instead of simulated ones as well.

The final part of this thesis was devoted to simulations, where several possible scenarios were showcased. These scenarios included different measurement approaches, employing

both Kalman filter with transformed time difference of arrival and angle of arrival measurements and extended Kalman filter that works directly with the values of these measured signals. It should be emphasized that the experiment serves mainly as a demonstration of the possibilities of graphical illustration. The result of this thesis is the implemented software that allows the user to configure their own scenario and analyze the results.

Future work could focus on expanding the evaluation map for a wider range of scenarios and different receiver layout. Integrating a combination of motion models, whose state dimensions are not the same, e.g. using all four models simultaneously for estimation process could be of interest as well.

References

[1] V. Vasyliev, D. Vasyliev, and K. Naumenko, "Using data of multilateration surveillance system for aircraft tracking," in 2016 4th International Conference on Methods and Systems of Navigation and Motion Control (2016, pp. 279–283. DOI: 10.1109/MSNMC.2016.7783161.

[2] M. Khalaf-Allah, "Time of arrival (toa)-based direct location method," in 2015 16th International Radar Symposium (IRS), 2015, pp. 812–815. DOI: 10.1109/IRS.2015.7226229.

[3] Y. Yang, J. Baldwin, and A. Smith, "Multilateration tracking and synchronization over wide areas," in Proceedings of the 2002 IEEE Radar Conference (IEEE Cat. No.02CH37322), 2002, pp. 419–424. DOI: 10.1109/NRC.2002.999755.

[4] K. Ho and W. Xu, "An accurate algebraic solution for moving source location using tdoa and fdoa measurements," IEEE Transactions on Signal Processing, vol. 52, no. 9, pp. 2453–2463, 2004. DOI: 10.1109/TSP.2004.831921.

[5] H. Blom and Y. Bar-Shalom, "The interacting multiple model algorithm for systems with markovian switching coefficients," IEEE Transactions on Automatic Control, vol. 33, no. 8, pp. 780–783, 1988. DOI: 10.1109/9.1299.

[6] T. L. Ogle and W. D. Blair, "Design of imm estimators with nearly constant velocity modes for tracking maneuvering targets," in 2017 IEEE Aerospace Conference, 2017, pp. 1–8. DOI: 10.1109/AERO.2017.7943656.

[7] P. Groves, Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems, Second Edition. 2013.

[8] T. Sathyan, A. Sinha, and T. Kirubarajan, "Passive geolocation and tracking of an unknown number of emitters," IEEE Transactions on Aerospace and Electronic Systems, vol. 42, no. 2, pp. 740–750, 2006. DOI: 10.1109/TAES.2006.1642587.

[9] A. Jasch, T. Feuerle, G. Scoor, and P. Hecker, "Geometrical siting considerations for wide area multilateration systems," in IEEE/ION Position, Location and Navigation Symposium, 2010, pp. 1304–1308. DOI: 10.1109/PLANS.2010.5507349.

[10] P. R. Blackman S., Design and analysis of modern tracking systems. 1999.

[11] T. Bar-Shalom Willet, Tracking and Data Fusion: A Handbook of Algorithms. 2012. DOI: 10.1109/MCS.2012.2204808.

[12] W. D. Blair, "Design of nearly constant velocity track filters for brief maneuvers," in 14th International Conference on Information Fusion, 2011, pp. 1–8.

[13] X. Rong Li and V. Jilkov, "Survey of maneuvering target tracking. part i. dynamic models," IEEE Transactions on Aerospace and Electronic Systems, vol. 39, no. 4, pp. 1333–1364, 2003. DOI: 10.1109/TAES.2003.1261132.

[14] Q. Li, R. Li, K. Ji, and W. Dai, "Kalman filter and its application," in 2015 8th International Conference on Intelligent Networks and Intelligent Systems (ICINIS), 2015, pp. 74–77. DOI: 10.1109/ICINIS.2015.35.

[15] J. Duník, "Identifikace systémů a filtrace," 2020. DOI: https://doi.org/10.24132/ZCU.2018.07750.

[16] T. K. Yaakov Bar-Shalom X.-Rong Li, "Adaptive estimation and maneuvering targets," in Estimation with Applications to Tracking and Navigation, ch. 11, pp. 421–490, ISBN: 9780471221272. DOI: https://doi.org/10.1002/0471221279.ch11. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/0471221279.ch11. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/0471221279.ch11.

[17] E. Mazor, A. Averbuch, Y. Bar-Shalom, and J. Dayan, "Interacting multiple model methods in target tracking: A survey," IEEE Transactions on Aerospace and Electronic Systems, vol. 34, no. 1, pp. 103–123, 1998. DOI: 10.1109/7.640267.