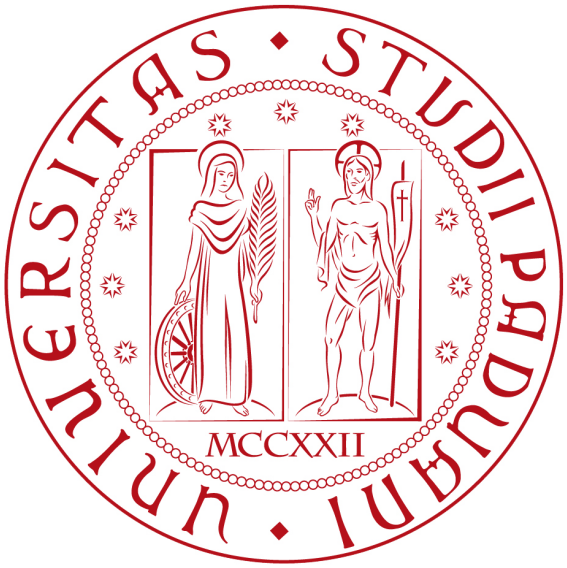
University of Padova
Academic year 2016-2017 (795rd)
Department of Information Engineering (DEI)
Master's thesis in Telecommunications Engineering



Learning techniques for positioning algorithms

Author: Giovanni Pilon
Advisor: Andrea Zanella

Dedicato alla mia famiglia, ed in particolare a mio papà, Claudio.

Abstract

This work develops a study of machine learning applied to positioning techniques based on times of arrival . We introduce a probabilistic framework to model the times of arrival and the user position, and we use this framework to empower a learning algorithm that learns information about the channel bias.

Two variations of the algorithm are presented: the first uses a single channel bias distribution for each base station, the other utilizes multiple ones, dividing the cell into regions and assigning to each of them a different distribution.

A simulator was developed to test the algorithm, showing good improvements in the positioning precision. The algorithm and the simulator are modelled in the context of an LTE network, but they can be adapted to any wireless system that employs arrival times.

Contents

Chapter 1

Introduction

Today positioning based on Global navigation satellite systems (GNSS) is the most precise available in outdoor enviornment. However its performance degrades in urban area, and it is often convinient to switch if off to save battery of the mobile devices. Mobile network positioning, although less precise than GNSS, is an energy-efficient solution that comes with its advantages. It can be used for example to automatically find the presence of coverage holes by the network provider, minimizing the need of drive tests [1], [2]. Mobile positions are also used in emergency situations [3], in the USA when an user call the 911 the network operator passes his position to the nearest Public Safety Answering Point. Additionally mobile positioning can be used to enhance satellite positioning [4], [5] or for location based services (LBS) [6].

Time-of-Arrival (ToA) positioning techniques are used both in (GNSS) and in LTE cellular networks. In GNSS such as GPS and GLONASS, satellites are precisily synchronized and simultaneously transmit signals known to the receivers. The receiver (not synchronized with the transimetters) is able to estimate the ToA for each signal by using a sliding correlator for each transmitting satellite. The ToA estimate is infact obtained as the maximum value of the correlator output. With each ToA estimate the receiver is then able to perform a multilateration calculation estimating its own position [7]. The same principle is used for Observed Time Difference Of Arrival (OT-

DOA) in LTE networks where the transmitters are synchronized terrestrial base stations [8].

Satellite systems and cellular network share characteristics such as carrier frequency (order of gigaHertz), signal bandwidth (order of megaHertz) and transmitter power (order of tens of Watts). However the main differences that come into play in ToA positioning are the distance between transmitters and receivers and the different Line-of-Sight (LOS) conditions. In particoular the LOS can often be treated as a clear line of sight channel in satellite systems, and the range distance can be calculated directly using the estimated ToA and the speed of light. This is not true in cellular systems where the urban environment and the relatively low position of base stations often create a Non-Line-of-Sight (NLOS) environment.

In cellular networks the majority of the signal energy can arrive through a NLOS reflection that bounces on buildings and obstacles [8]. As a result the signal can travel up to a hundred of meters more than an hypothetical LOS signal, causing the measured ToA to have a bias of hundreds of nanoseconds (the signal moves at the speed of light). Furthermore when multipath fading occurs in these channels, unresolvable multipath can also result in a positive bias in the ToA [9]. In particular given the bandwidth W of the signal the multipath componets cannot be resolved if the delays between paths is less than $1/W$, in this scenario the correlator create a blurred peak that makes the LOS component indistinguishable from the next multipath component. In ultrawide bandwidth (UWB) systems delays as small as nanoseconds can be resolved [10], but in cellular networks the multipath bias can reach the order of microseconds due to the smaller bandwidth (megaHertz). This bias in the ToA measurement due to the NLOS reflections and to unresolvable multipath fading is reffered to as channel bias. The channel bias is the main obstacle that any ToA-based positioning technique must face in cellular networks [11].

In this work we present a positioning technique for LTE systems where the mobile network can improve the accuracy of the position by estimating the channel bias distribution for each base station. With the creation of a probabilistic framework it is possible to learn information about the channel bias and use its distribution to improve the position estimation.

1.1 Thesis outline

Chapter 2 presents the general framework of ToA-based positioning in LTE networks. We then present the system model, the distribution used to model the channel bias, the position estimator and the learning algorithm. Chapter 3 present the simulator that was implemented to test the algorithm aswell as the adopted assumptions and the parameters choosen for the simulations. In Chapter 4 we present the results while Chapter 5 contains conclusions and possible extensions to this approach.

Chapter 2

Framework

This chapter presents all the information about how the system model is created. First we introduce LTE positioning, describing the positioning problem and the methods used in current mobile networks, then we introduce our model.

The system model uses the same framework used in LTE OTDOA but goes a step further by using ToA samples not only to estimate the position but also to learn information about the channel bias. As in all learning algorithms the tradeoff is the necessity of a training time. However the strength of the algorithm is that it does not need a fixed training set of data, instead, it separates the data available in ToA batches and processes them in consecutive iterations. When sufficient ToA samples are gathered a new iteration is performed, improving the estimation of the channel bias distribution. Moreover, once the parameters of the channel bias distribution are sufficiently valid, the learning can be suspended, allowing the location server to perform the position estimation using directly the learned parameters.

Since real datasets of LTE networks are usually protected by intellectual property rights, we present some realistic assumptions that have been adopted to create a simulator.

Finnaly it is important to note that while in this work we consider the case of an LTE network, the approach used can be easily adapted to any kind of wireless network.

2.1 LTE positioning

In mobile networks Reference Signal (RS) are used in the down-link to estimate the radio channel characteristics. These signals, transmitted at specific times and frequencies, can be used for positioning purposes. In LTE the Cell-specific Reference Signal (CRS) had a poor Signal to Interfirence plus Noise Ratio (SINR) [12]. Therefore a new reference signal called Positioning Reference Signal (PRS) was defined in the release of the standard [13] specifically for OTDOA positioning.

PRS are designed to increase the reference signal power while reducing the interference of other BS through isolation. PRS are transmitted in predefined bandwith and configuration parameters: subframe offset (Δ_{PRS}), periodicity (T_{PRS}), duration (N_{PRS}), muting pattern and muting sequence periodicity (T_{REP}). The period T_{PRS} between positioning occasions can be 160, 320, 640, 1280 subframes, and the number of consecutive subframes N_{PRS} during a positioning occasion can be 1, 2, 4 or 6 subframes. An example is presented in figure 2.1.

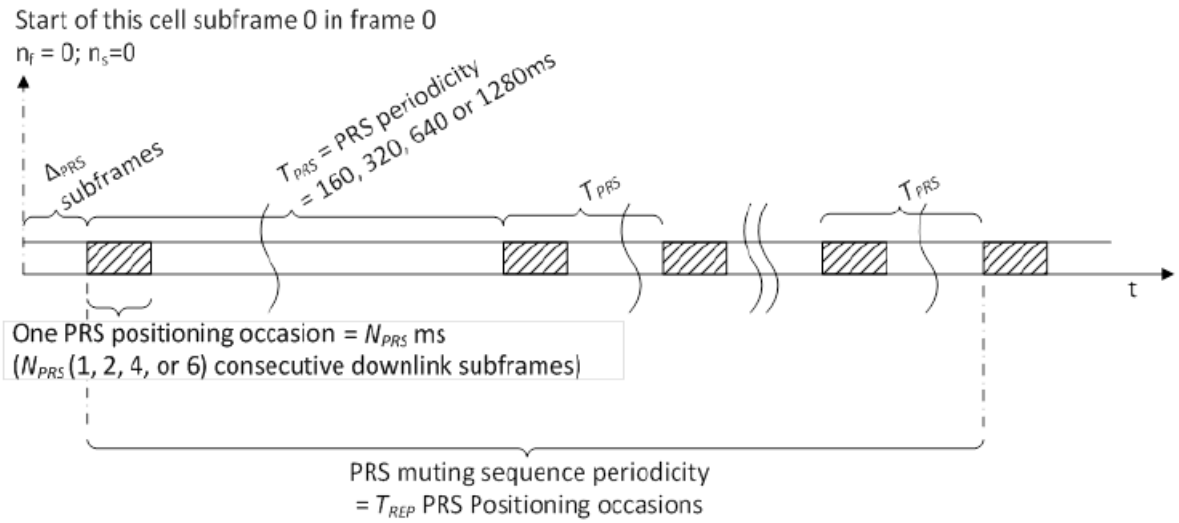


Fig. 2.1: Example of PRS transmissions (Source: [14])

Isolation to improve OTDOA performance can be achived in 3 domains:

- Code domain: each cell transmit a different PRS sequence, orthogonal

to other PRS sequences in the code domain.

- Frequency domain: six possible frequency offsets are defined in the PRS bandwidth. If two PRS collide in the frequency domain, orthogonal sequences grant isolation.

- Time domain: time based blanking (muting), can also be used to prevent collisions in the frequency domain.

PRS are configurable for each cell and a good Physical Cell Identity (PCI) planning distributes the frequencies shift in a smart way; in this way it is avoided the risk of having the same frequency in adjacent cells, or pointing at each other. An example is presented in figure 2.2, where each color represents a different frequency.

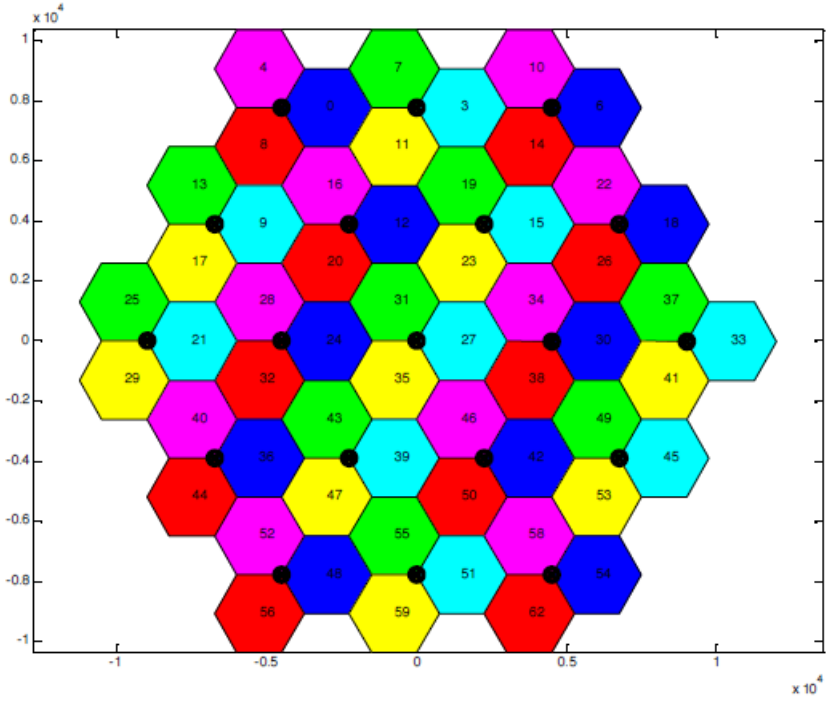


Fig. 2.2: PCI planning for PRS (Source: [14]).

The UE is informed about the PRS configurations through the LTE Positioning Protocol (LPP). The LPP provides the necessary communication

between the Location Server (LS) and the UE, aswell as position requests when a new measurement is necessary. A typical positioning routine is presented in figure 2.3.

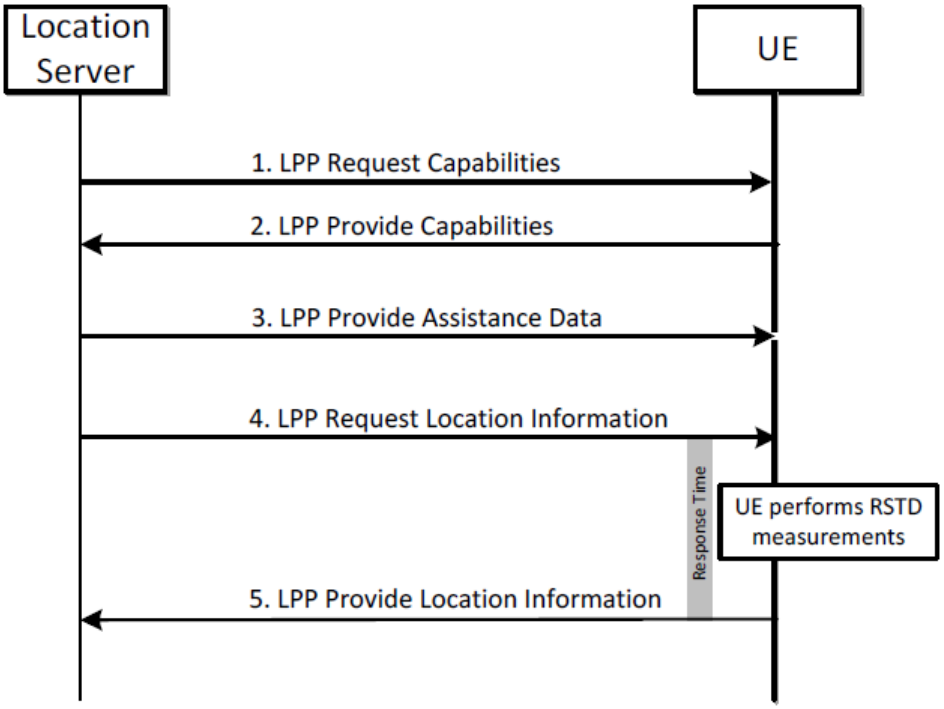


Fig. 2.3: Example of LPP Positioning Procedure (Source: [14]).

2.1.1 OTDOA

Observed Time Difference Of Arrival (OTDOA) is the most precise positioning technique in LTE [15]. It is a multilateration method in which the UE measures at least three ToA of PRS simultaneously transmitted by nearby BS [16], [17]. The UE select a ToA of a reference BS and subtract the other ToAs from it. For example with 3 BS and their ToAs, t_1, t_2, t_3, the obtained Time Differences Of Arrival (TDOA) are $TDOA_{2,1} = (t_2 - t_1)$ and $TDOA_{3,1} = (t_3 - t_1)$. In LTE this TDOA is also called Reference Signal Time Difference (RSTD). As shown in the next section each of these time

difference corresponds to a hyperbola of possible user positions [18]. Interesecting two hyperbolas we obtain the estimated user position, as illustrated in figure 2.4.

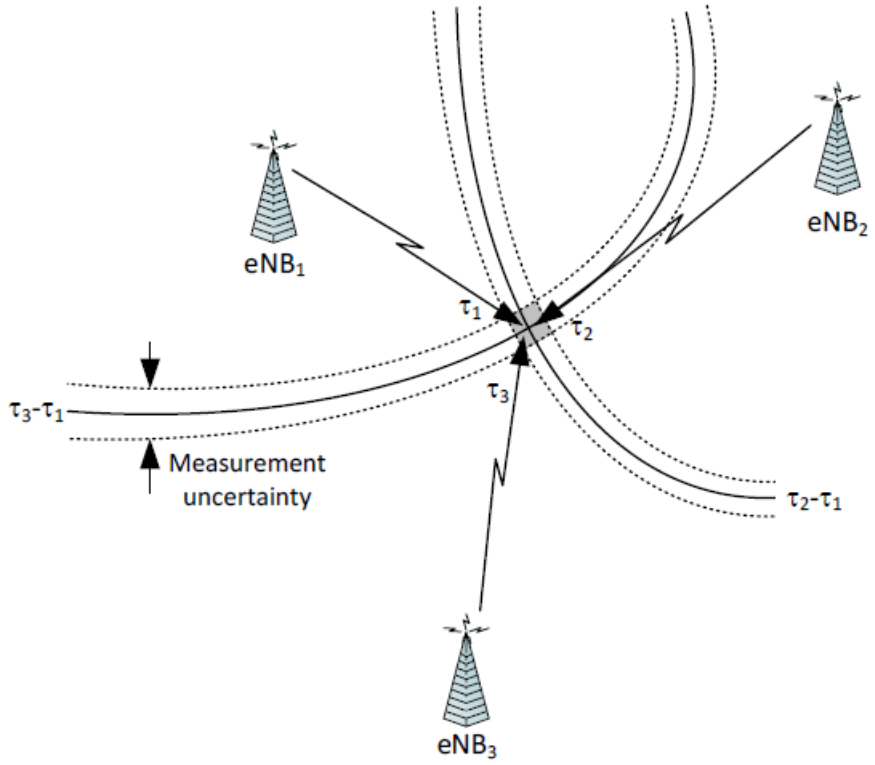


Fig. 2.4: OTDOA multilateration (Source: [14]).

Since the ToA measurements have an intrinsic error the intersection define an area of possible positions rather than a single point. The error is caused by many factors: multipath fading, unresolvable multipath, synchronization error of the BS or simply ToA estimation error. The main parameter that influences the precision of the measurement is the bandwith of the signal W, greater bandwiths come with more precise measurments since the multipath components can be resolved with a precision of $1/W$ seconds. PRS are usually defined between 2.5 MHz and 10 MHz and the RSTD error can go up to hundreds of nanoseconds, leading to a position error that can be as high as a hundred meters [8], [14].

Other positioning techniques worth mentioning are: angle-based positioning [19], [20]; Received Signal Strength (RSS) [21], [22]; Round Trip Time (RTT) positioning [23], [24]; Enhanced Cell ID [25].

2.2 System model

The system model considers a wireless system with B transmitters (BS) which are time synchronized, they transmit simultaneously mutually ortogonal PRS. The receiver UE uses a sliding correlator matched to each signal waveform to estimates ToA from the corresponding BS. If the BS are not synchronized it is possible to find the Relative Time Difference (RTD) by time measurments at known positions [26]. There are not other stringent requirements for the network itself.

The receiver is not synchronized with the transmitters, for this reason the transmit time τ of the base stations cannot be known in advance and must be accounted in the equation. In the ideal case of LOS transmission and perfect estimation the ToA for each base station $b \in \{1...B\}$ would be equal to:

$$t_{b-ideal} = \frac{||\mathbf{x} - \mathbf{x}_b||}{c} + \tau. \tag{2.1}$$

- $\mathbf{x} = [\mathrm{x}_1, \mathrm{x}_2]$ is the unknown two-dimensional vector giving the position of the receiver
- $\mathbf{x}_b = [\mathrm{x}_{b1}, \mathrm{x}_{b2}]$ is the known two-dimensional vector giving the position of base station b
- c is the known speed of light
- τ is the unknown transmit time, assumed to be the same for each base station.

However in a real case scenario we have to account other factors. Hence the ToA estimate for each base station $b \in \{1...B\}$ is equal to:

$$t_b = \frac{||\mathbf{x} - \mathbf{x}_b||}{c} + \tau + \gamma_b + n_b. \tag{2.2}$$

- γ_b is the random variable that describes the channel bias. As explained in the previous chapter non line-of-sight transmission and unresolvable multipath create an unknown channel bias γ_b, which is modelled as a random variable. It holds $\gamma_b \geqslant 0$, in fact the channel bias can be thought of as a delay of the line-of-sight transmission.

- n_b an additional unknown random variable that accounts for the residual estimation error and the base stations synchronization error.

The localization problem consists in estimating the device position vector $\mathbf{x}$ and the transmit time τ given the positions of B base stations $\mathbf{x}_b$ and the ToA t_b from each of them ($b = 1, ..., B$). As shown in the next sections the learning algorithm takes care of estimating the distribution of γ_b and n_b, therefore we need at least three ToA measurements to solve for the three unknowns x_1, x_2 and τ, hence it must be $B \geqslant 3$. It is possible to include height using three-dimensional position vectors, in this case the requirement becomes $B \geqslant 4$.

In OTDOA the transmit time τ is eliminated from the equations through the difference between the ToAs. Given a reference ToA t_1 the TDOA becomes:

$$TDOA_{i,1} = t_i - t_1 = \frac{||\mathbf{x} - \mathbf{x}_i||}{c} - \frac{||\mathbf{x} - \mathbf{x}_1||}{c} + (\tau_i - \tau_1) + (\gamma_i - \gamma_1) + (n_i - n_1), \tag{2.3}$$

where the transmit time offset $\tau_i - \tau_1$ is zero if the network is correctly synchronized.

Here we opted for ToA instead of TDOA because this approach simplifies the mathematical modelling of the algorithm. As shown in [27] these approachs are equivalent in terms of location estimate variance for LOS transmission.

2.2.1 Channel bias probability distribution $p(\gamma_b)$

In the general case the random variable γ_b that describes the channel bias has an unknown probability distribution. The channel bias is given by how the transmitted signals are reflected on obstacles like buildings and other objects in the enviornment. For this reason it is impossible to know in advance the distribution that γ_b will have in a certain region and in particoular the distribution can change from one area to another. In literature there are many solutions to approximate the unknown distribution of γ_b, for example using uniform distribution [28], exponential distribution [29] or gaussian distributions [30] [31].

In this work we use a gaussian mixture to describe the probability distribution $p(\gamma_b)$, following the approach presented in [8]. This is advantageous compared to other approaches because a gaussian mixture makes it possible to more accurately describe the real distribution and most importantly because it will better adapt to different scenarios.

A gaussian mixture of R_b gaussians is used for each base station b, each component i of the gaussian mixture has weight π_{ib}, mean μ_{ib} and variance $\tilde{\sigma}_{ib}^2$. It holds $\sum_{i=1}^{R_b} \pi_{ib} = 1$ as normalization condition. The PDF of the Gaussian Mixture is hence expressed as

$$p(\gamma_b) = \sum_{i=1}^{R_b} \frac{\pi_{ib}}{\tilde{\sigma}_{ib}\sqrt{2\pi}} e^{\left[\frac{-1}{2\tilde{\sigma}_{ib}^2}\left(\gamma_b - \mu_{ib}\right)^2\right]}. \tag{2.4}$$

The learning algorithm learns the parameters of the gaussian mixture starting from an initial set of parameters. It is important to note that each base station will have its own channel bias distribution $p(\gamma_b)$ and that these distributions can be different from one another.

2.2.2 Probability distribution of the error n_b

The residual error n_b $(b = 1, ..., B)$ is the sum of the estimation error and the BS synchronization error. These two error can be characterized by two zero-mean Gaussian random variables with unknown variance σ_{syn}^2 and σ_{est}^2.

Since these two errors are independet n_b is also a zero-mean Gaussian random variable with variance $\sigma_b^2 = \sigma_{syn}^2 + \sigma_{est}^2$ and PDF

$$p(n_b) = \frac{1}{\sigma_b\sqrt{2\pi}} e^{\left[\frac{-1}{2\sigma_b^2} n_b^2\right]}. \tag{2.5}$$

2.2.3 Likelihood distribution $p(\mathbf{t}|\mathbf{x}, \tau)$

Using $p(n_b)$ the conditional likelihood of t_b can be written as:

$$p(t_b|\mathbf{x}, \tau, \gamma_b) = \frac{1}{\sigma_b\sqrt{2\pi}} e^{\left[\frac{-1}{2\sigma_b^2}\left(t_b - \frac{||\mathbf{x}-\mathbf{x}_b||}{c} - \tau - \gamma_b\right)^2\right]}. \tag{2.6}$$

Marginalizing over the probability distribution $p(\gamma_b)$ we obtain:

$$\begin{aligned} p(t_b|\mathbf{x}, \tau) &= \int p(t_b|\mathbf{x}, \tau, \gamma_b) p(\gamma_b) d\gamma_b \\ &= \sum_{i=1}^{R_b} \frac{\pi_{ib}}{\sigma_{ib}\sqrt{2\pi}} e^{\left[\frac{-1}{2\sigma_{ib}^2}\left(t_b - \frac{||\mathbf{x}-\mathbf{x}_b||}{c} - \tau - \mu_{ib}\right)^2\right]}, \end{aligned} \tag{2.7}$$

where we used a new variable σ_{ib}^2 that represents the variance of the corresponding gaussian mixture component $\tilde{\sigma}_{ib}^2$ plus the variance of the estimation error, σ_b^2:

$$\sigma_{ib}^2 = \tilde{\sigma}_{ib}^2 + \sigma_b^2. \tag{2.8}$$

Since the ToA measurements are independent it is possible to write the joint likelihood distribution of the ToAs as:

$$p(\mathbf{t}|\mathbf{x},\tau) = \prod_{b=1}^{B} p(t_b|\mathbf{x},\tau), \tag{2.9}$$

where $\mathbf{t}$ is a vector containing the B ToAs, $\mathbf{t} = [t_1, ..., t_B]$.

2.2.4 Prior $p(\mathbf{x}, \tau)$

For each BS we define the prior $p_b(\mathbf{x}, \tau)$ proportional to the reciprocal of the range $||\mathbf{x} - \mathbf{x}_b||$. In this way positions farther away from the BS are weighted less. The overall prior is equal to the product of the priors of each base:

$$p(\mathbf{x},\tau) = K \prod_{b=1}^{B} p_b(\mathbf{x},\tau) = K \prod_{b=1}^{B} \frac{1}{(||\mathbf{x} - \mathbf{x}_b|| + \epsilon)}. \tag{2.10}$$

Here ϵ is a parameter to prevent numerical instability, it must be true that $||\mathbf{x} - \mathbf{x}_b|| \gg \epsilon$, and $\epsilon > 0$; K is a normalization parameter.

2.2.5 Position MAP estimator

A Maximum a Posteriori (MAP) estimator is used for the position vector $\mathbf{x}$ and transmit time τ. Using Bayesan inference the MAP probability is proportional to the product of the likelihood $p(\mathbf{t}|\mathbf{x},\tau)$ and the prior $p(\mathbf{x},\tau)$.

$$p(\mathbf{x},\tau|\mathbf{t}) \propto p(\mathbf{t}|\mathbf{x},\tau)p(\mathbf{x},\tau). \tag{2.11}$$

The estimate $(\hat{\mathbf{x}}, \hat{\tau})$ is obtained as:

$$(\hat{\mathbf{x}}, \hat{\tau}) = \underset{\mathbf{x},\tau}{\operatorname{argmax}}\, p(\mathbf{x},\tau|\mathbf{t}) = \underset{\mathbf{x},\tau}{\operatorname{argmax}}\, p(\mathbf{t}|\mathbf{x},\tau)p(\mathbf{x},\tau). \tag{2.12}$$

Using the equations obtained in the previous sections we get:

$$(\hat{\mathbf{x}}, \hat{\tau}) = \underset{\mathbf{x},\tau}{\operatorname{argmax}} \prod_{b=1}^{B} \sum_{i=1}^{R_b} \frac{\pi_{ib}}{(||\mathbf{x} - \mathbf{x}_b|| + \epsilon)\sigma_{ib}\sqrt{2\pi}} e^{\left[\frac{-1}{2\sigma_{ib}^2}\left(t_b - \frac{||\mathbf{x} - \mathbf{x}_b||}{c} - \tau - \mu_{ib}\right)^2\right]}. \tag{2.13}$$

Figure 2.5 shows the typical shape of the position posterior distribution for a single base station. Figure 2.6 presents a countour graph of three different posterior distributions, each belonging to a different base stations; the position MAP probability is the product of those distributions.

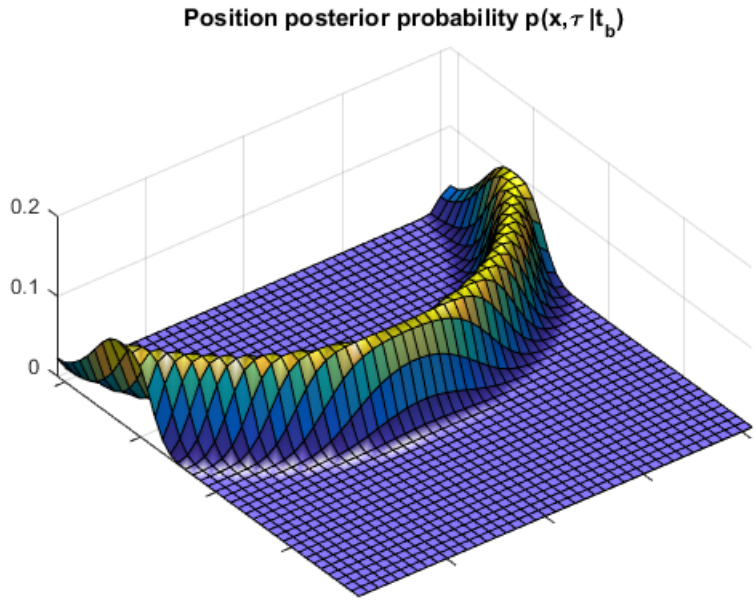


Fig. 2.5: Portion of the position posterior distribution for a single BS.

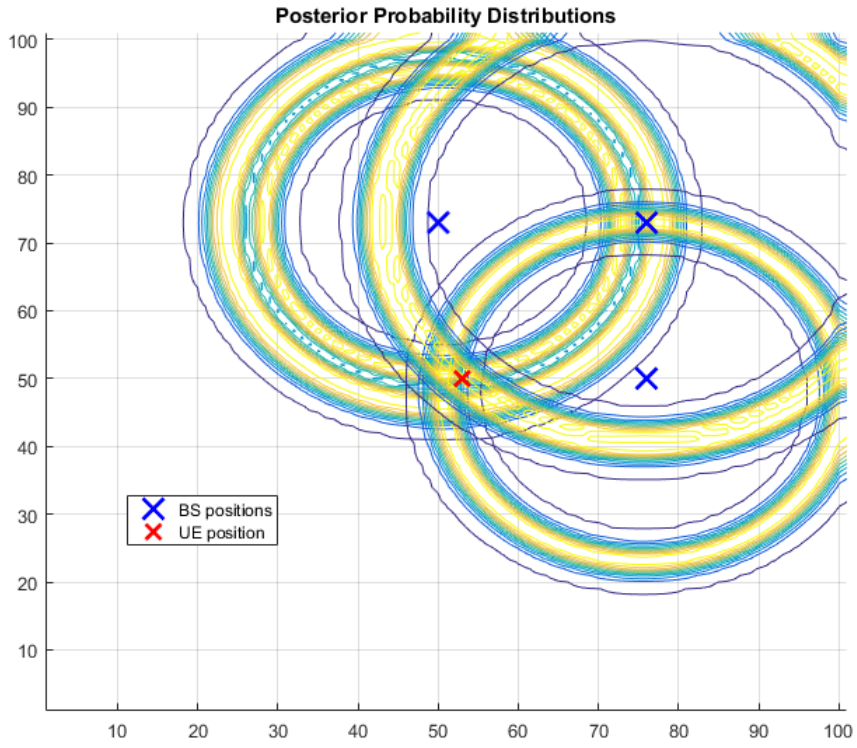


Fig. 2.6: Countour graph of three posterior distributions.

2.3 Learning algorithm for the channel bias distribution

The learning algorithm takes care of learning the parameters of the channel bias probability distributions $p(\gamma_b)$. The algorithm uses K iterations, each iteration requires a set of N ToA measurements for each of the B base stations. At iteration k the parameters of $p(\gamma_{k,b})$ $(b = 1, ..., B)$ are $\pi_{ib,k}$, $\mu_{ib,k}$ and $\sigma^2_{ib,k}$ $(i = 1, ..., R_b)$. We group these parameters into a vector $v_{b,k}$ and these vectors into a matrix $V_k = [v_{1,k}, ..., v_{B,k}]$. The idea behind the algorithm is to use the parameters V_k to estimate the position $(\hat{\mathbf{x}}, \hat{\tau})$, and then use this position to update the parameters V_{k+1} which will be used in the next iteration.

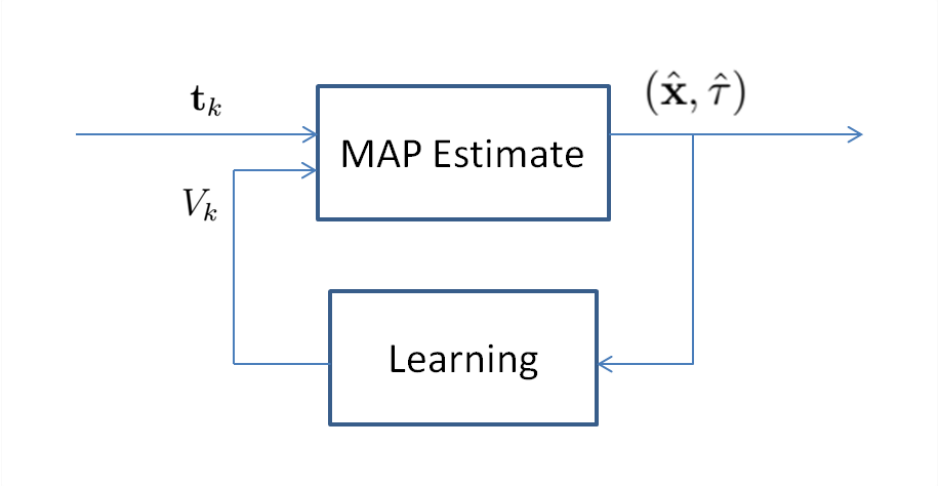


Fig. 2.7: Scheme of the MAP estimation and learning algorithm.

LEARNING ALGORITHM

At every iteration k the following procedure is repeated for each base station b using the parameters from V_k:

- For each ToA measurement vector $\mathbf{t}_n$ $(n = 1, ..., N)$ estimate the location and transmit time without using the measurements from base b (to prevent instabilities):

$$(\hat{\mathbf{x}}_{k,n}^{\neg b}, \hat{\tau}_{k,n}^{\neg b}) = \underset{\mathbf{x},\tau}{\operatorname{argmax}} \prod_{j=1, j\neq b}^{B} p(t_{j,n}|\mathbf{x}, \tau) p(\mathbf{x}, \tau). \tag{2.14}$$

- Compute the channel bias $\gamma_{n,k,b}$ $(n = 1, ..., N)$, using $(\hat{\mathbf{x}}_{k,n}^{\neg b}, \hat{\tau}_{k,n}^{\neg b})$ and (2.2) as if the position estimates were the true location:

$$\gamma_{n,k,b} = t_{b,n} - \frac{||\hat{\mathbf{x}}_{k,n}^{\neg b} - \mathbf{x}_b||}{c} - \hat{\tau}_{k,n}^{\neg b}. \tag{2.15}$$

- Use the N measurements $\gamma_{n,k,b}$ to update the parameters for the next iteration.

$$[\gamma_{1,k,b}, ..., \gamma_{N,k,b}] \rightarrow \mathbf{v}_{b,k+1}. \tag{2.16}$$

This update is a clustering problem for Gaussian Mixtures and can be done in different ways. Here we opted for Expectation Maximization. Other options are k-means clustering [32], or dirichlet processes [33].

It is important to note that since the position estimation is done with a leave-out procedure that excludes b the minimum number of base stations must be increased by 1. Hence it is required $B \geqslant 4$ for two-dimensional position vectors and $B \geqslant 5$ for three-dimensional vectors.

2.3.1 Expectation maximization

Given a set of channel bias samples γ_n $(n = 1, ..., N)$ the clustering problem consists in finding the R_b gaussians that better fit the distribution of the samples. Which means finding the weights π_i, means μ_i and variances σ_i^2 of this Gaussian Mixture $(i = 1, ..., Rb)$, for simplicity we indicate this group of parameters as v.

Each γ_n is treated as an observation from one of the gaussian mixture components. We define a label $z = (z_1, ...z_{R_b})$ for each observation; z is a vector of R_b binary indicators that are mutually exclusive (only one z_i is equal to 1 and the other are equal to 0). For each sample γ_n this indicator represents the identity of the gaussian mixture component that generated it. The Gaussian Mixture weights can in fact be written as $\pi_i = p(z_i)$. If these indicators are known then it is easy to compute the parameters v using the classic maximum likelihood estimation for each of the gaussian component. Since z is not know, the goal is estimating the correct one for each γ_n. The Expectation Maximization algorithm solve this problem by first estimating the probability of the indicators z_i and the using this probability to calculate new parameters values for v [34].

We define the membership weight of the n^{th} sample as:

$$w_{ni} = p(z_{ni} = 1|\gamma_n, v) = \frac{p_i(\gamma_n|z_i, v_i)\pi_i}{\sum_{m=1}^{R_b} p_m(\gamma_n|z_m, v_m)\pi_m}, \tag{2.17}$$

here $p_i(\gamma_n|z_i, v_i)\pi_i$ is the probability density for the i^{th} gaussian.

The EM algorithm is an iterative algorithm that updates the parameters v untill when convergence is detected. It can be divided in two steps:

- Compute w_{ni} for all data points γ_n and for all mixture components $i = 1, ..., R_b$. This gives an $N \times R_b$ matrix of membership weights where each row sum to 1 (because $\sum_{i=1}^{R_b} w_{ni} = 1$).
- Use the membership weights and the data to calculate new parameter values. We define $N_i = \sum_{n=1}^{N} w_{ni}$ as the sum of the membership weights for the i^{th} component, this is the effective number of samples assigned to component i. For each i the updated parameters are:
  - $\pi_{i,new} = \dfrac{N_i}{N}$,
  - $\mu_{i,new} = \left(\dfrac{1}{N_i}\right) \sum_{n=1}^{N} w_{ni}\gamma_n$
  - $\sigma^2_{i,new} = \left(\dfrac{1}{N_i}\right) \sum_{n=1}^{N} w_{ni}(\gamma_n - \mu_{i,new})^2$

2.4 Range-Aware learning algorithm

It is possible to further expand the model by using multiple channel bias distributions for each base station. In a real enviornment the channel bias can change from one region to another due to many factors. This can be taken into account defining regions within the cell and assigning a channel bias distribution to each of such regions. Theoretically there is no limit to the number of regions that can be used or to their shapes. However to avoid overfitting we opted for three circular regions $R1_b$, $R2_b$, $R3_b$ defined by two radius $r1$ and $r2$ for each BS b. These circles are defined using the BS as center of the regions. Figure 2.8 shows an example for BS 1.

At iteration k each base station has three channel bias distributions: $p(\gamma_{k,b})|R1$ for $R1$, $p(\gamma_{k,b})|R2$ for $R2$ and $p(\gamma_{k,b})|R3$ for $R3$. Consequently each BS has three parameter vectors $v_{b,k,R1}$, $v_{b,k,R2}$, $v_{b,k,R3}$ with the parameters of the corresponding distribution.

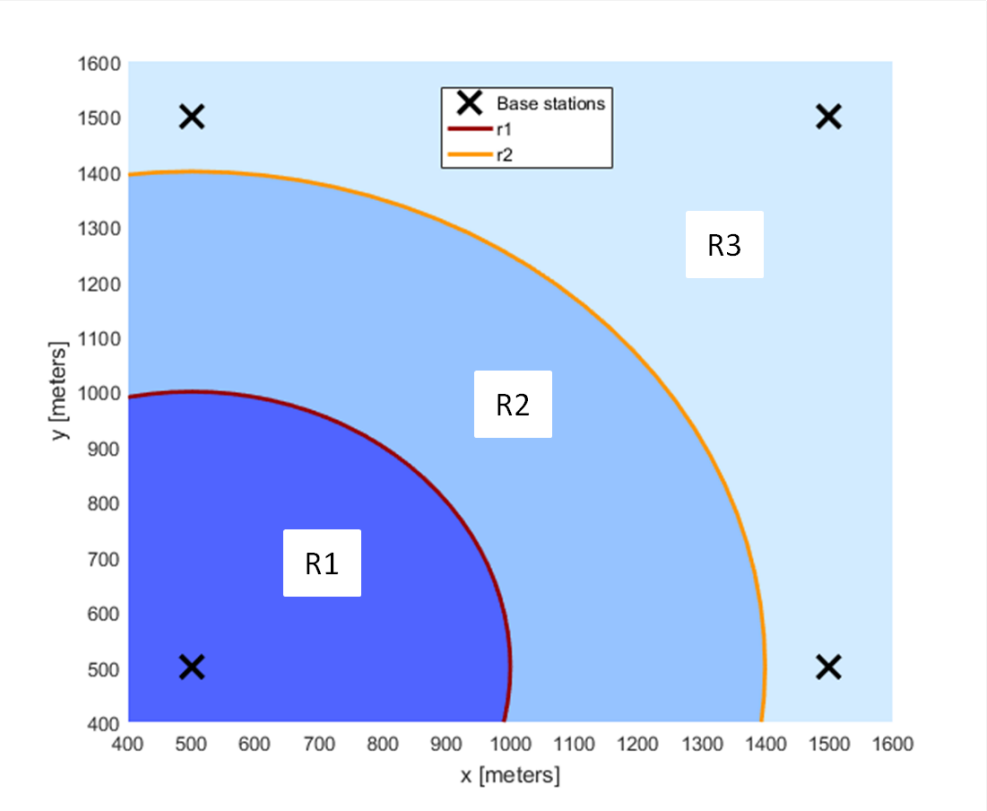


Fig. 2.8: Example of 3 regions for BS 1.

RANGE-AWARE LEARNING ALGORITHM (RALA)

At every iteration k the following procedure is repeated for each base station b:

- For each ToA measurement vector $\mathbf{t}_n$ $(n = 1, ..., N)$ estimate a preliminary position $\hat{\mathbf{x}}_p$ using the MAP estimation of the normal learning algorithm or TDOA positioning.

- Find which region contains $\hat{\mathbf{x}}_p$ for each BS b:

  - if $||\hat{\mathbf{x}}_p - \mathbf{x}_b|| < r1 \Rightarrow \hat{\mathbf{x}}_p \in R1_b$.
  - else if $||\hat{\mathbf{x}}_p - \mathbf{x}_b|| < r2 \Rightarrow \hat{\mathbf{x}}_p \in R2_b$.
  - else $\hat{\mathbf{x}}_p \in R3_b$.

- For each BS b select the correct parameters vector $v_{b,k,Ri}$, knowing the region from the previous step.
- perform a MAP position estimation again, this time using the selected parameter vectors $v_{b,k,Ri}$. The obtained $\hat{\mathbf{x}}_n$ is the position estimate of the Range-Aware algorithm for the n^{th} time sample.
- For each BS b and for each region Ri update the parameters of $v_{b,k,Ri}$ like in the learning algorithm. This time using only the positions $\hat{\mathbf{x}}_n$ that belong to Ri.

Chapter 3

Simulation setup

LTE/4G datasets are not easily accessible since they can contain sensible information for the network operators. To test the performance of the algorithm we created a simulator that first generates ToAs, and then run the positioning algorithm to estimate the position and to learn the distribution of the channel bias.
The structure of the simulator is presented in figure 3.1. The data generation requires only the position of the BS. A total of N_T random users position U_{xy} are selected within the cell, for each of them B ToA are generated (one from each BS), these ToA samples are created deducing a realistic model for the channel bias. The obtained matrix M_{ToA} of size $B \times N_T$ is passed to the positioning algorithm, togheter with the matrix B_{xy} of the BS positions. The positioning algorithm uses M_{ToA} to perform the learning and the MAP estimation. The learning does not require to know the true user positions U_{xy}, nor the model that generated the ToA. In fact the MAP estimate $\hat{\mathbf{x}}$ is enough to correctly perform expectation maximization on the channel bias distribution.

3.1 Channel bias samples

We consider a channel with h clusters, where the ith cluster arrives with a delay γ_i, for $i = 1, \ldots, h-1$ with respect to the first cluster, for which we

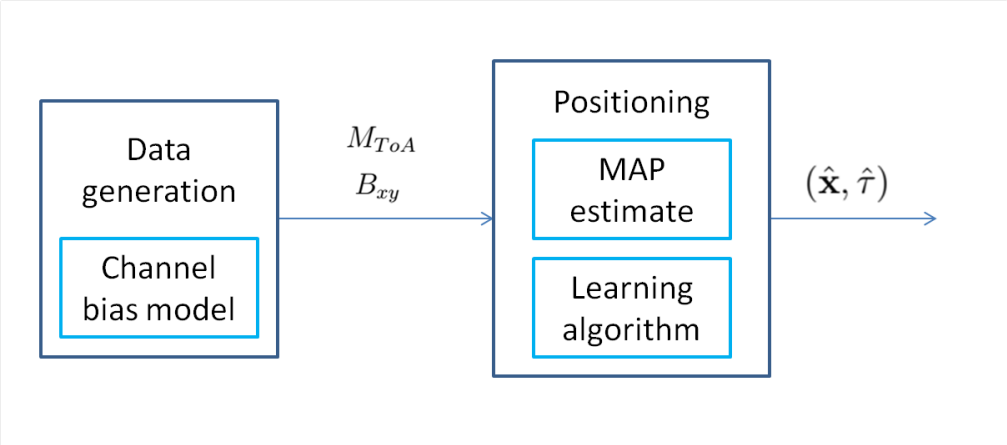


Fig. 3.1: Scheme of the simulator.

set $\gamma_0 = 0$. Each cluster consists of a random number of rays, which arrive with random phases but very close delay. Therefore, assuming a Rayleigh fading model, the aggregate power of the ith cluster can be modelled as an exponential random variable P_i with mean $\bar{P}_i$ [35]. A typical power delay profile is exponential with parameter α, so that the *average* received power of the clusters can be expressed as $\bar{P}_i = \bar{P}_0 \exp(-\alpha\gamma_i)$, $i = 0, \ldots, h-1$.

Now, the channel bias γ will take values in the set $\{\gamma_0, \gamma_1, \ldots, \gamma_{h-1}\}$, with $\gamma = \gamma_j$ if the power of the jth cluster is larger than that of the other clusters. Denoting by $p_\gamma(j) = \Pr[\gamma = \gamma_j]$ we hence have

$$p_\gamma(j) = \prod_{i \neq j} \Pr[P_i \leq P_j] = \int_0^\infty \prod_{i \neq j} \Pr[P_i \leq a\bar{P}_0 e^{-\alpha\gamma_j}] e^{-a} da \tag{3.1}$$

$$= \int_0^\infty \prod_{i \neq j} \left(1 - e^{-ae^{-\alpha(\gamma_j - \gamma_i)}}\right) e^{-a} da \tag{3.2}$$

A simplified model may be obtained by approximating (3.2) with

$$\tilde{p}_\gamma(j) = F \Pr[P_j \leq P_0] = F \int_0^\infty \left(1 - e^{-ae^{-\alpha(\gamma_j - \gamma_0)}}\right) e^{-a} da \tag{3.3}$$

$$= F\left(1 - \int_0^\infty e^{-a\left(1 + e^{-\alpha(\gamma_j - \gamma_0)}\right)} da\right) = \frac{F}{e^{\alpha(\gamma_j - \gamma_0)} + 1} \tag{3.4}$$

where F is a normalization constant.

Considering $h = 20$ clusters uniformly distributed in the set $[0, \gamma_{MAX}]$, with parameters $\alpha = 12 \times 10^6$ and $\gamma_{MAX} = 500$ ns, by using the approximate expression given by (3.4), we get the PDF in figure 3.2. The parameters were choosen respecting the location uncertainties presented by the Technical Specification Group for Radio Access Network in [26] (a channel bias of 500 ns corresponds to a position error of 150 meters in the worst case scanario).

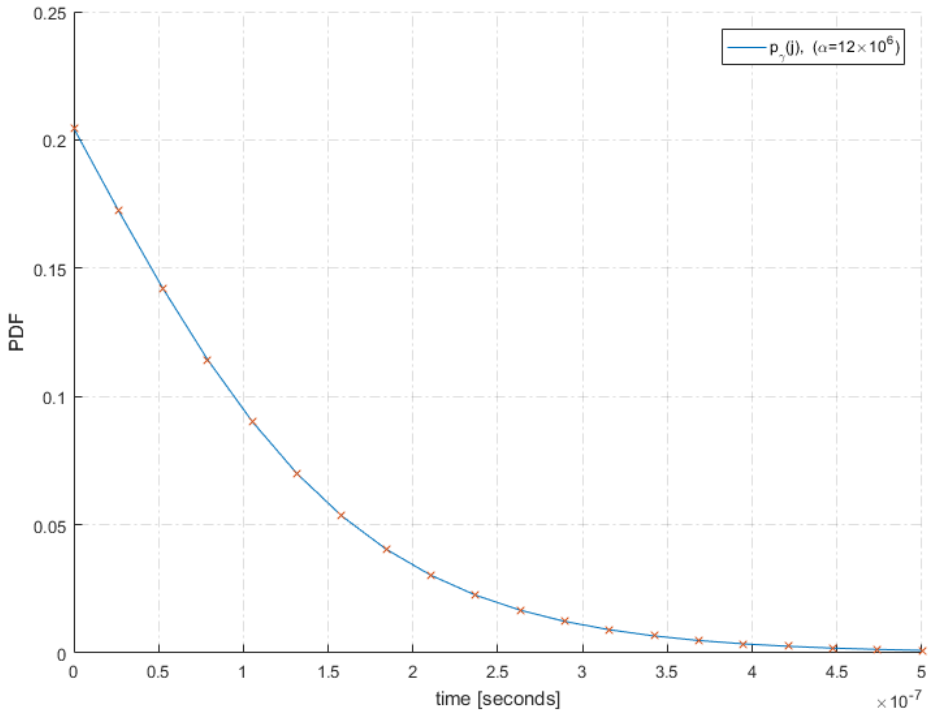


Fig. 3.2: $\tilde{p}_\gamma(j)$ used to generate channel bias samples ($\alpha = 12 \times 10^6$).

We sample the channel bias PDF to generate ToA using the know distance between the user position U_{xy} and the BS. For the sample n ($n = 1, ..., N_T$) and BS b the ToA is:

$$t_{b,n} = \frac{||U_{xy|n} - \mathbf{x}_b||}{c} + \gamma_{b,n}. \tag{3.5}$$

3.2 Network geometry

To test the learning algorithm and his accuracy we consider a network with 4 base stations at the vertices of a square of side $s = 1000m$. The user positions are selected randomly within this square as in figure 3.3.

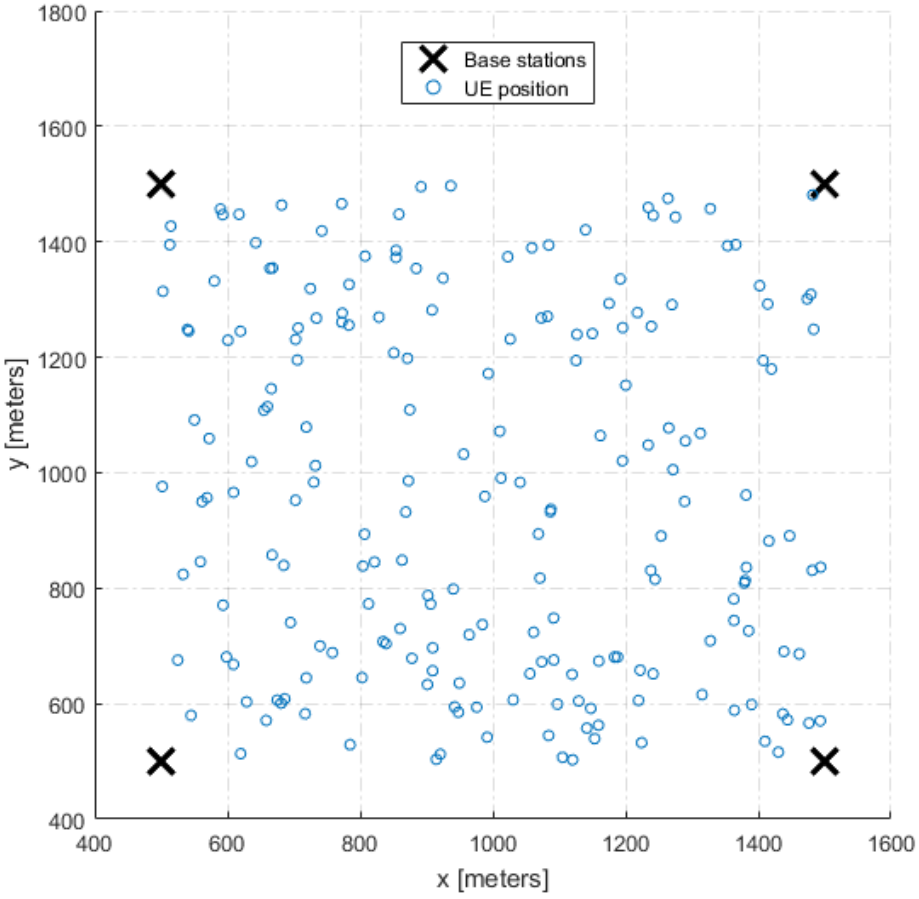


Fig. 3.3: Example of 200 random UE positions with 4 BS at the corner of a square area

3.3 Range-Aware algorithm simulation parameters

To test the Range-Aware algorithm we suppose that the channel bias is prone to be bigger as the distance from the BS increases. This is obtained creating a function that defines the parameter α over the range of possible distances. This is equivalent to say that as the distance of the user from the BS increase, so increases the probability of having a NLOS transmission (with LOS channels that become obstructed). In more general terms this means that the probability of receiving the signal with a bigger delay increases over distance. The channel bias behaves like this in cities with tall buildings that bounce the signal, a worst-case scenario for positioning algorithms.

We used a polynomial function of order two to define the relation between α and the user distance $||\mathbf{x}_b - \mathbf{x}||$, remembering that a lower α corresponds to a higher probability of a big channel bias:

$$\alpha = f(||\mathbf{x}_b - \mathbf{x}||) = \alpha_1 + \alpha_2 \Big(\frac{d_{MAX} - ||\mathbf{x}_b - \mathbf{x}||}{d_{MAX}} \Big)^2. \tag{3.6}$$

In this way we let α vary from:

- $\alpha = \alpha_1$ when $||\mathbf{x}_b - \mathbf{x}|| = d_{MAX}$
- $\alpha = \alpha_1 + \alpha_2$ when $||\mathbf{x}_b - \mathbf{x}|| = 0$.

We define $\alpha_1 = 4 \times 10^6$ and $\alpha_2 = 28 \times 10^6$ and the maximum distance allowed in the simulation to $d_{MAX} = 1500m$. Figure 3.4 presents the function $f(||\mathbf{x}_b - \mathbf{x}||)$. Figure 3.5 presents $\tilde{p}_\gamma(j)$ when $||\mathbf{x}_b - \mathbf{x}|| = d_{MAX}$, and figure 3.6 when $||\mathbf{x}_b - \mathbf{x}|| = 0$. Finally figure 3.7 presents an example of channel bias samples generated for 1000 random user positions.

3.3.1 Regions radius

We set $r1 = 500m$ and $r2 = 900m$, obtaining the regions in figure 3.8.

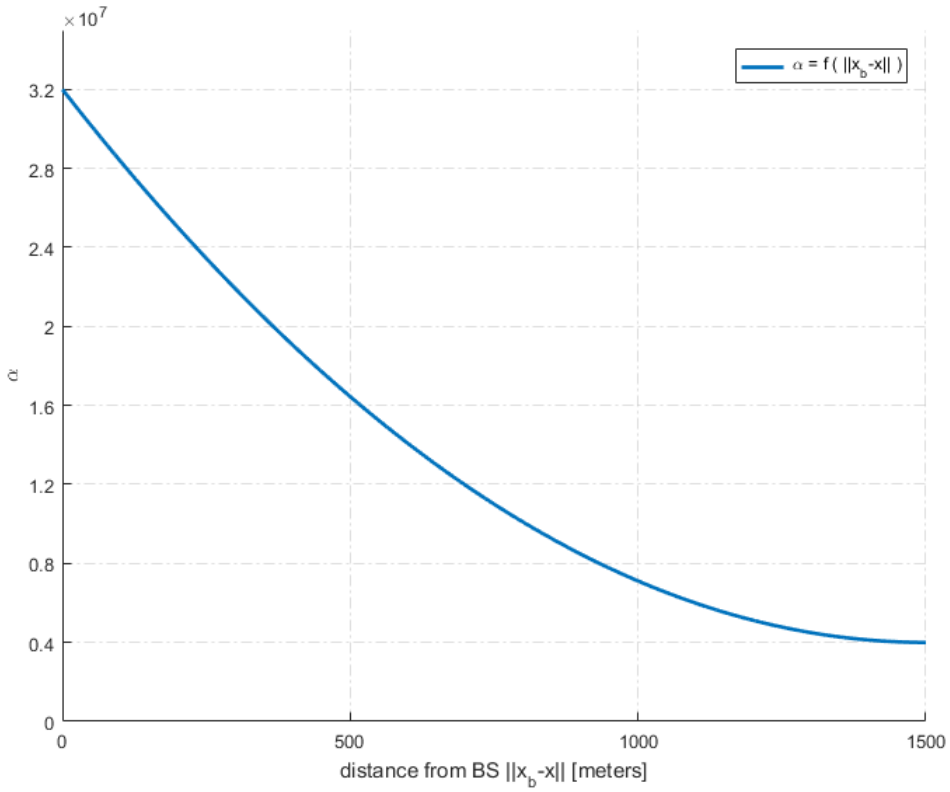


Fig. 3.4: α as a function of $||\mathbf{x}_b - \mathbf{x}||$

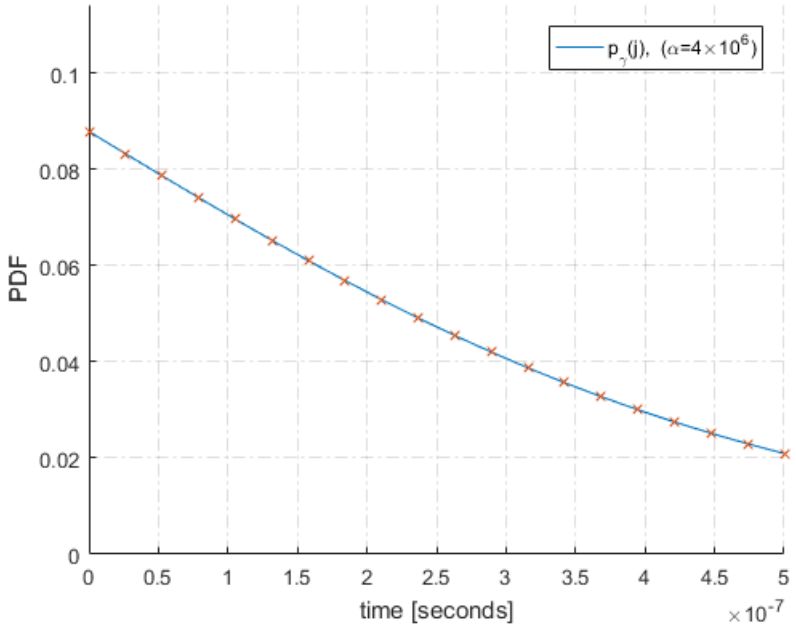


Fig. 3.5: $\tilde{p}_\gamma(j)$, with $\alpha = 4 \times 10^6$

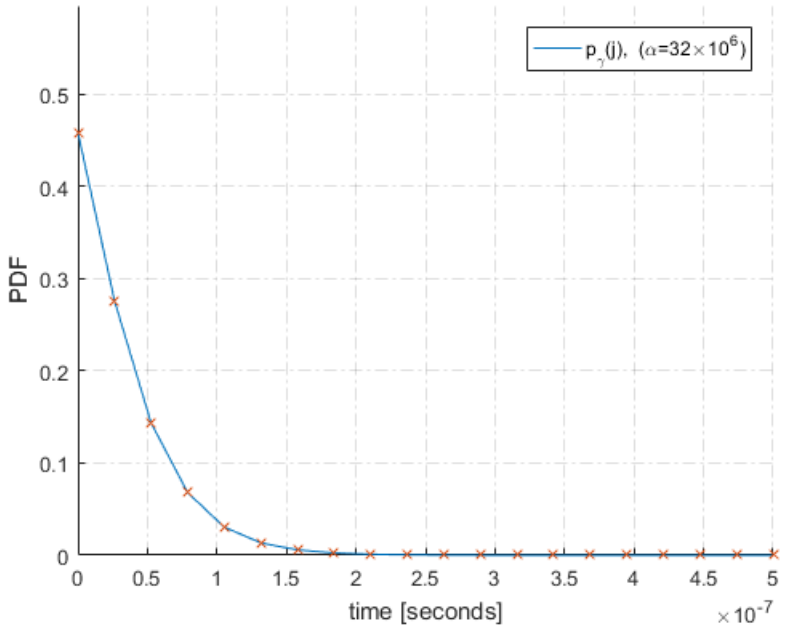


Fig. 3.6: $\tilde{p}_\gamma(j)$, with $\alpha = 32 \times 10^6$

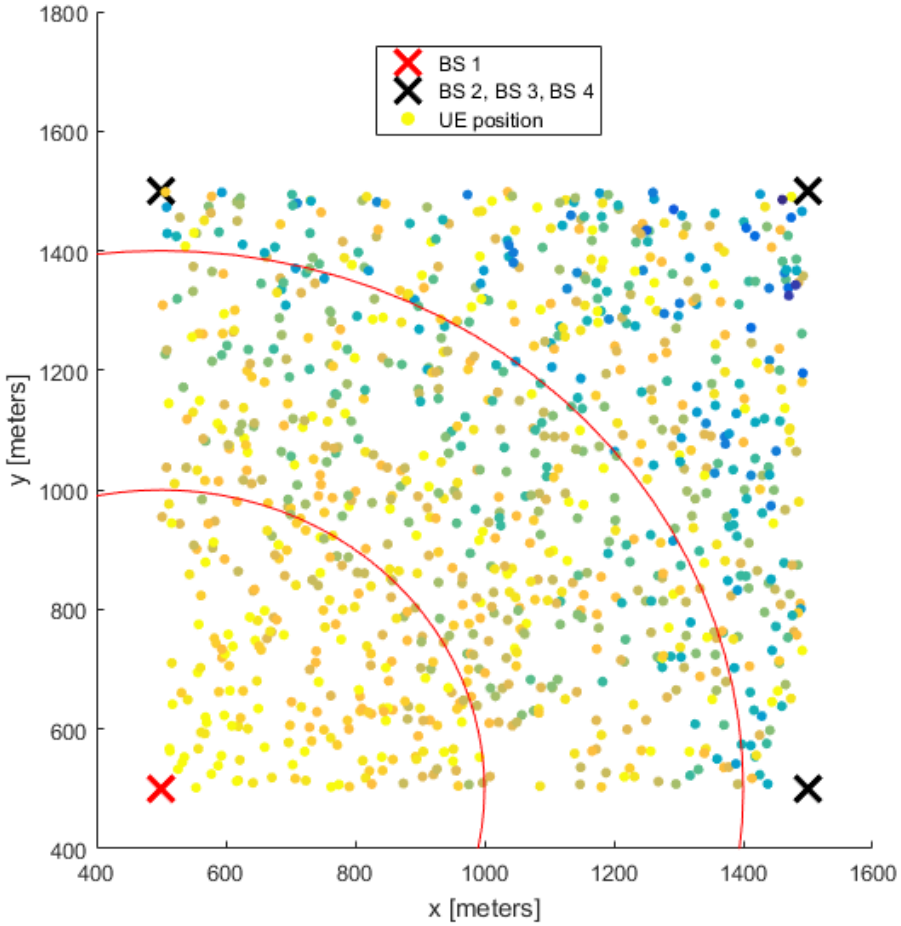


Fig. 3.7: Channel bias samples for base station 1 in 1000 random positions. Darker colors represent bigger channel bias.

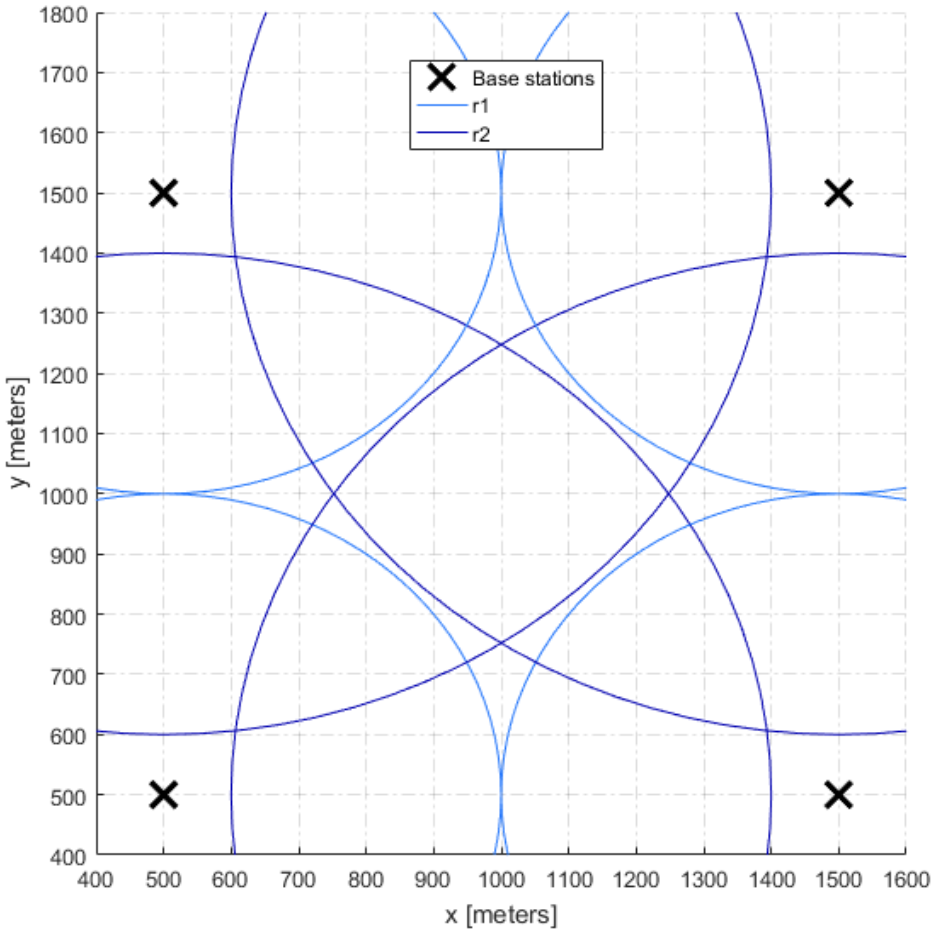


Fig. 3.8: Regions of the RA Learning Algorithm

Chapter 4

Results

The simulator was used to collect a variety of results, studying different perfomance metrics of the algorithm. In particular we investigated:

- The improvement of the performance when $p(\gamma_b)$ is trimmed for negative values, and the convergence of the algorithm in both cases.

- The clustering of the channel bias gaussian mixture for different values of R_b (number of components of the gaussian mixture) and the consequent change in perfomances.

- The comparison with a TDOA direct position estimation that uses the three nearest BS.

- The performance of the Range-Aware learning algorithm compared to the normal one.

The size of each iteration was 1000 samples and to test the convergence we used a total of 8000 samples (8 iterations). The initial parameters of the Gaussian Mixtures are $\mu_{ib} = 50ns$ and $\sigma_{ib} = 10ns$ for each component i, with weights uniformely divided, $\pi_{ib} = 1/R_b$.

4.1 Trimmed $p(\gamma_b)$

When generating data, and in real measurements, the channel bias is always greater than zero. In fact a negative channel bias could be obtained only with a signal moving faster than the speed of light. This means that the perceived ToA is always over-estimated and never under-estimated. The same is not true for the estimated user position and the distance from the base stations, which can be greater or smaller than the true distance. Since the learning algorithm uses the estimated user position to reverse engineer the channel bias distribution it is possible, and in fact it often happens, that the channel bias probability distribution has positive values for negative channel bias. This can be seen for example in figure 4.1. Since we know that these values represent a case which is physically impossible, we can optimize the algorithm by trimming the channel bias distribution $p(\gamma_b)$. To do so we multiply it for the unit step function $1(\gamma_b)$ and we renormalize it. An example is presented in figure 4.2.

Figure 4.3 compares the CDF of the position error for the two versions of the algorithm. Figures 4.4, 4.5 present the position error mean and standard deviation at each iteration when $p(\gamma_b)$ is not trimmed. While figures 4.6, 4.7 present these results with a trimmed $p(\gamma_b)$. Since this adjustment brings massive improvements to the algorithm we have used it for the rest of the simulations. We keep referring to the trimmed distribution as $p(\gamma_b)$ because this adjustment is done only before each MAP estimate and the rest of the algorithm remains unchanged.

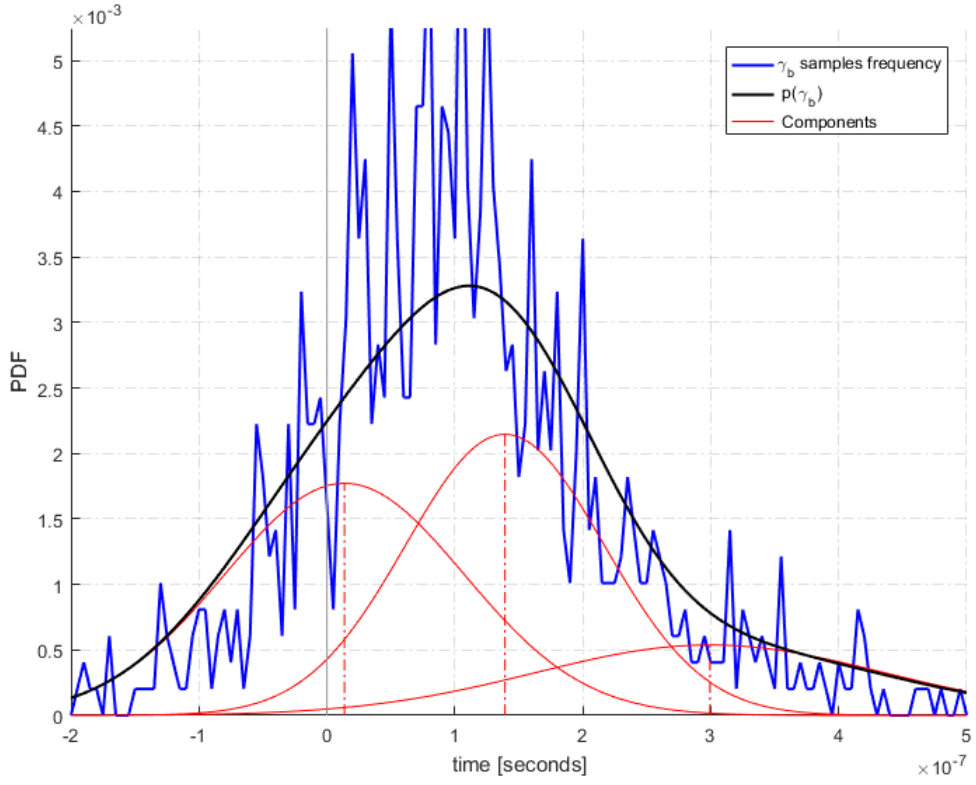


Fig. 4.1: $p(\gamma_b)$ and γ samples frequency, $Rb = 3$.

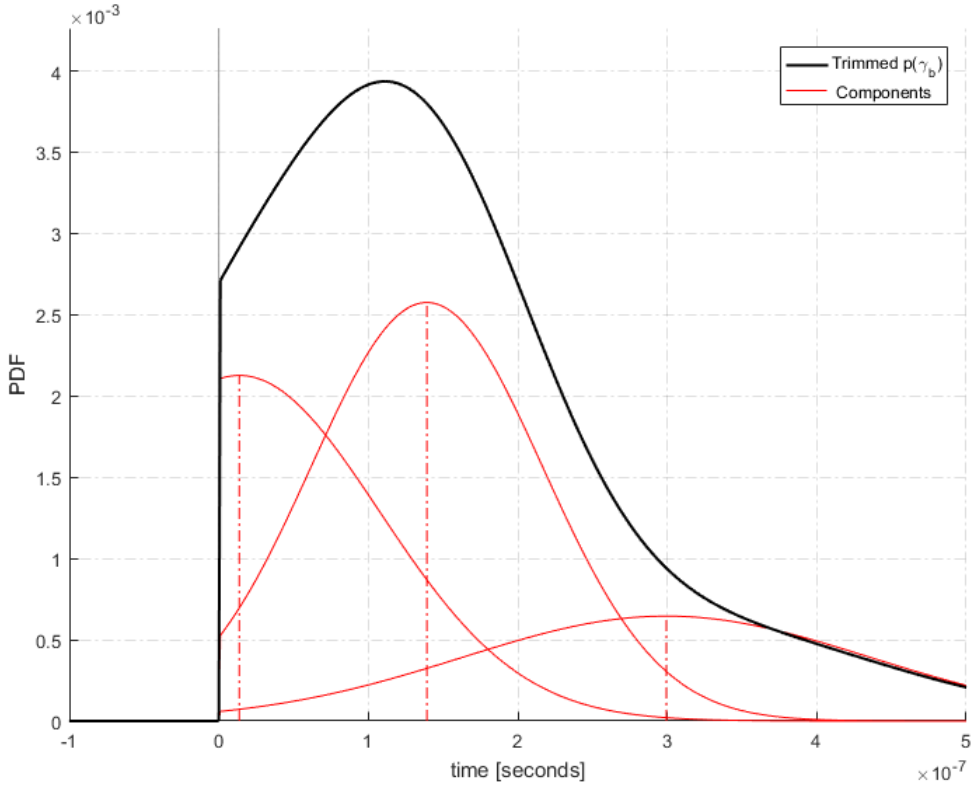


Fig. 4.2: Trimmed $p(\gamma_b)$, $Rb = 3$.

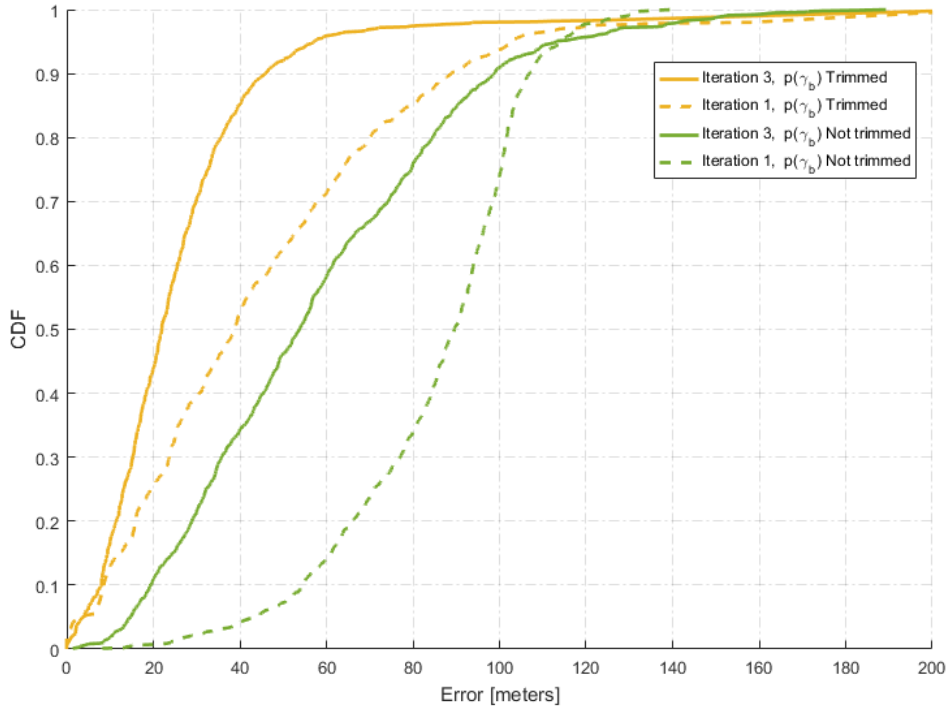


Fig. 4.3: CDF comparison between a trimmed and non-trimmed $p(\gamma_b)$.

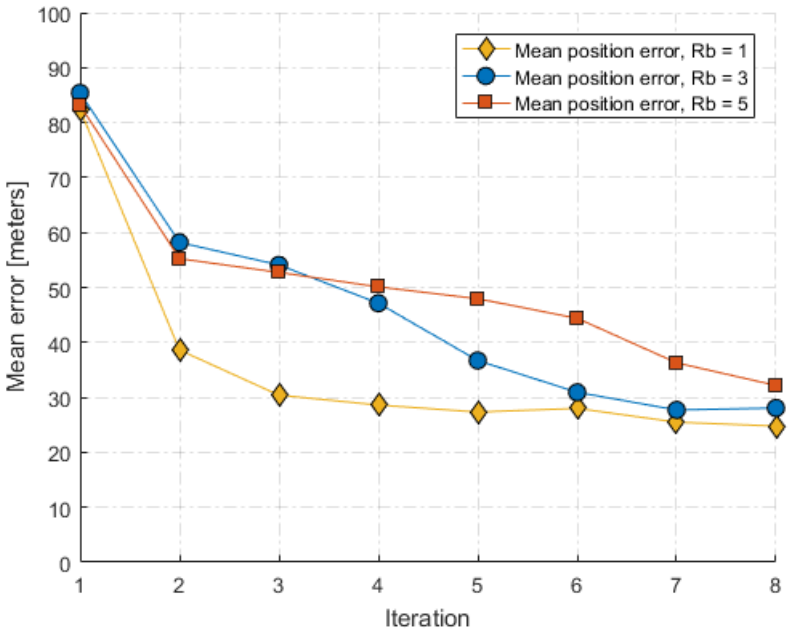


Fig. 4.4: Mean position error at each iteration. Non-trimmed $p(\gamma_b)$.

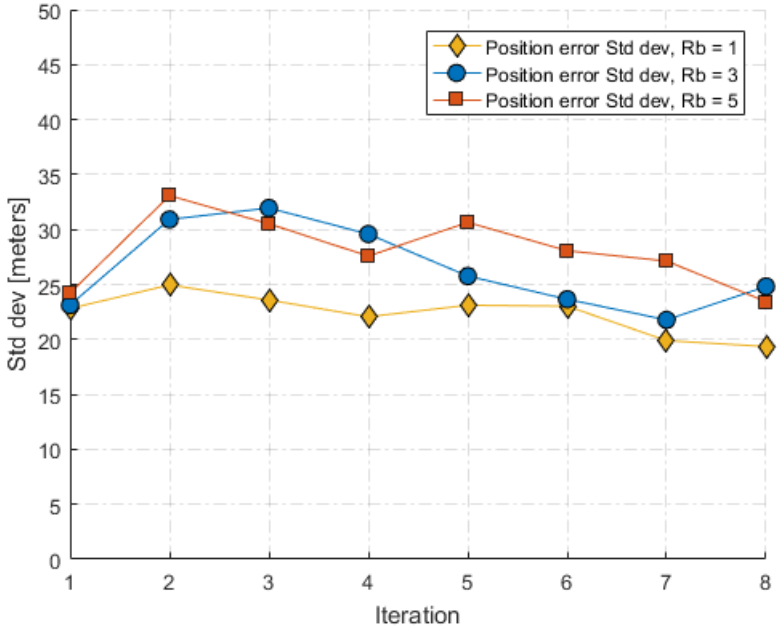


Fig. 4.5: Position error standard deviation at each iteration. Non-trimmed $p(\gamma_b)$.

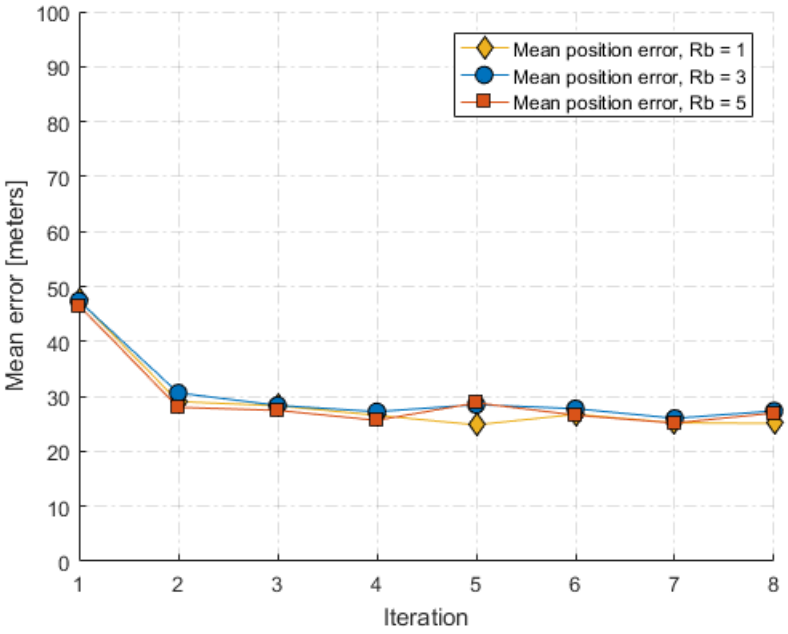


Fig. 4.6: Mean position error at each iteration. Trimmed $p(\gamma_b)$.

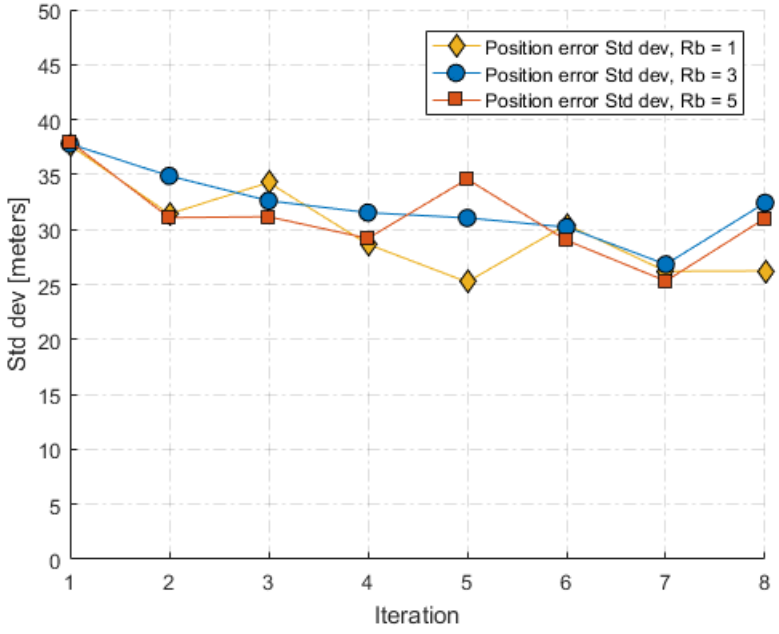


Fig. 4.7: Position error standard deviation at each iteration. Trimmed $p(\gamma_b)$.

4.2 Convergence and Gaussian Mixture components

As it is possible to see from figures 4.6 and 4.7 already at the third iteration the algorithm reachs convergence and stops improving. This can be also seen in figure 4.8 where the CDF of the fourth iteration overlaps the CDF of the third one. Table 4.1 presents mean and standard deviation of the position error at the third iteration, a higher number of componets R_b slightly improves the precision.

Position error, iteration 3			
	$R_b = 1$	$R_b = 3$	$R_b = 5$
Error mean [meters]	29.3	28.4	27.5
Error std dev [meters]	34.3	32.6	31.2

Table 4.1: Position error mean and standard deviation at the third iteration for different values of R_b.

Figures 4.9, 4.10, 4.11 present $p(\gamma_b)$ at the third iteration for different values of R_b ($R_b = 1$, $R_b = 3$, $R_b = 5$). Figure 4.12 presents the CDF at the first and third iteration for those values of Rb.

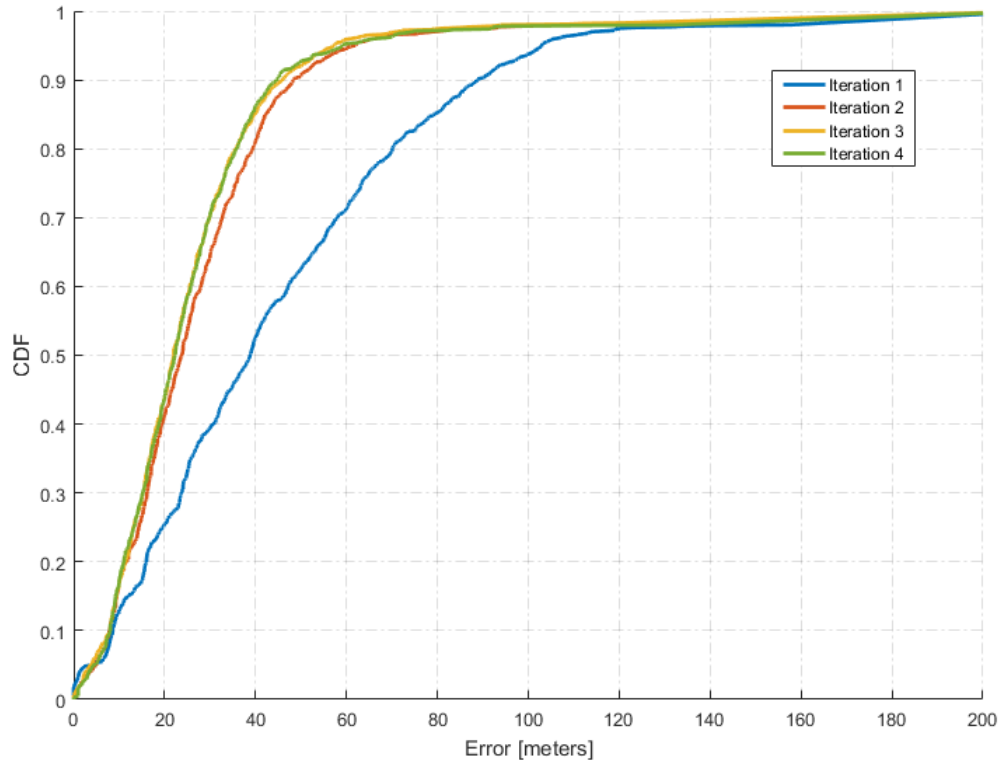


Fig. 4.8: CDF for different iterations, ($R_b = 3$).

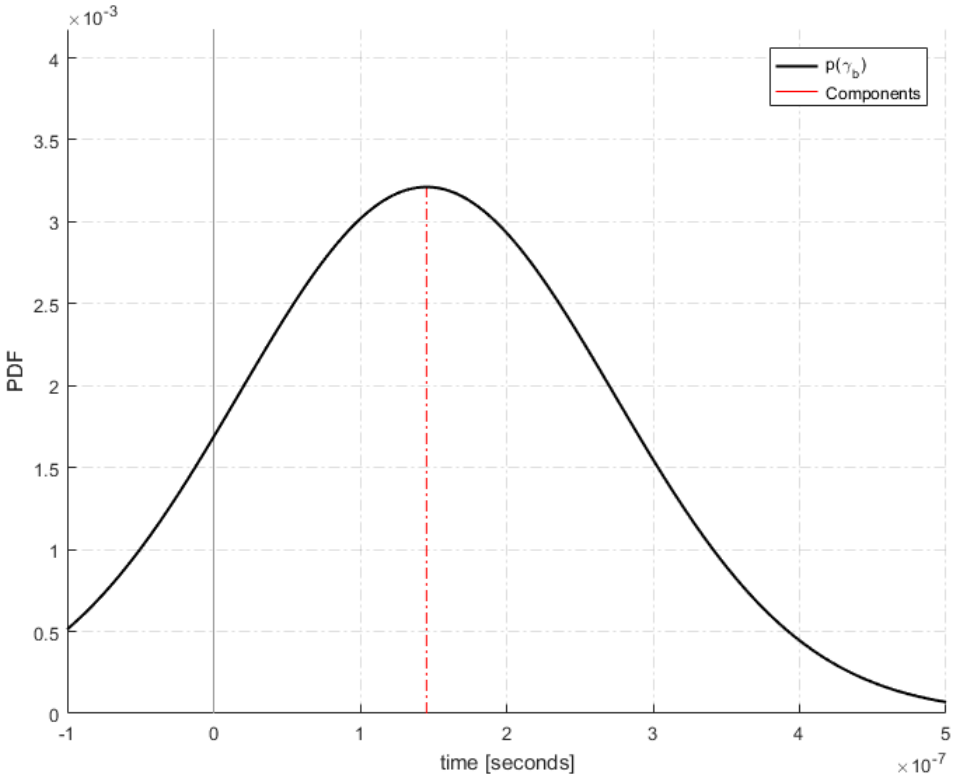


Fig. 4.9: $p(\gamma_b)$ for BS 1 (iteration 3, $R_b = 1$).

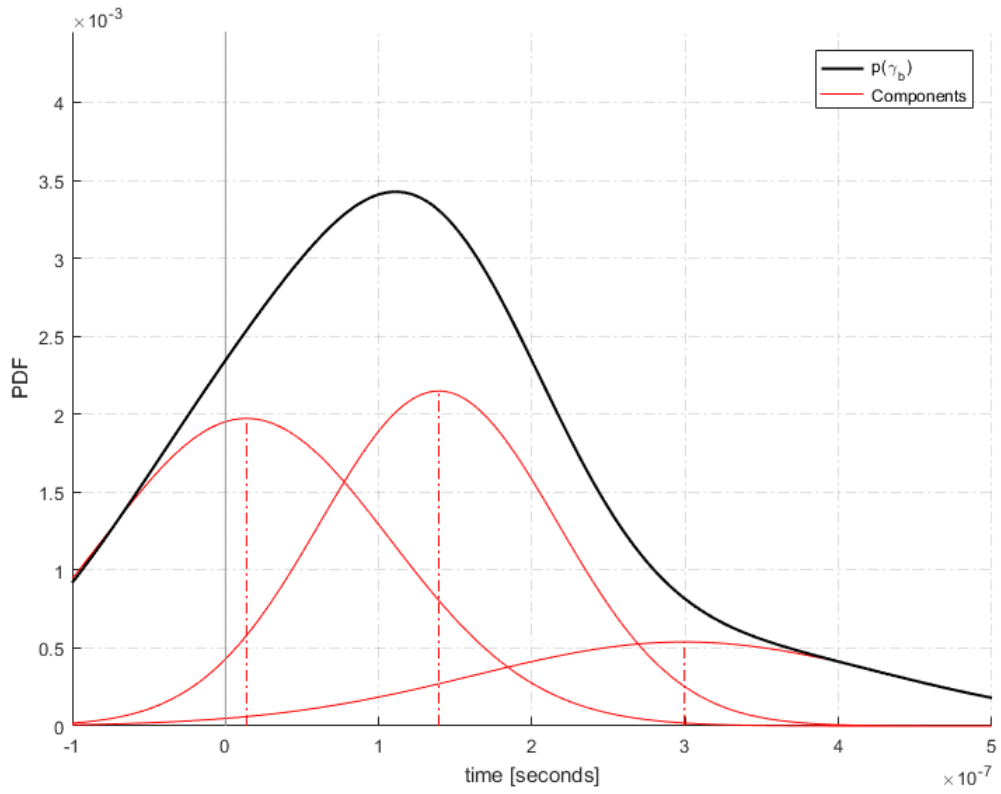


Fig. 4.10: $p(\gamma_b)$ for BS 1 (iteration 3, $R_b = 3$).

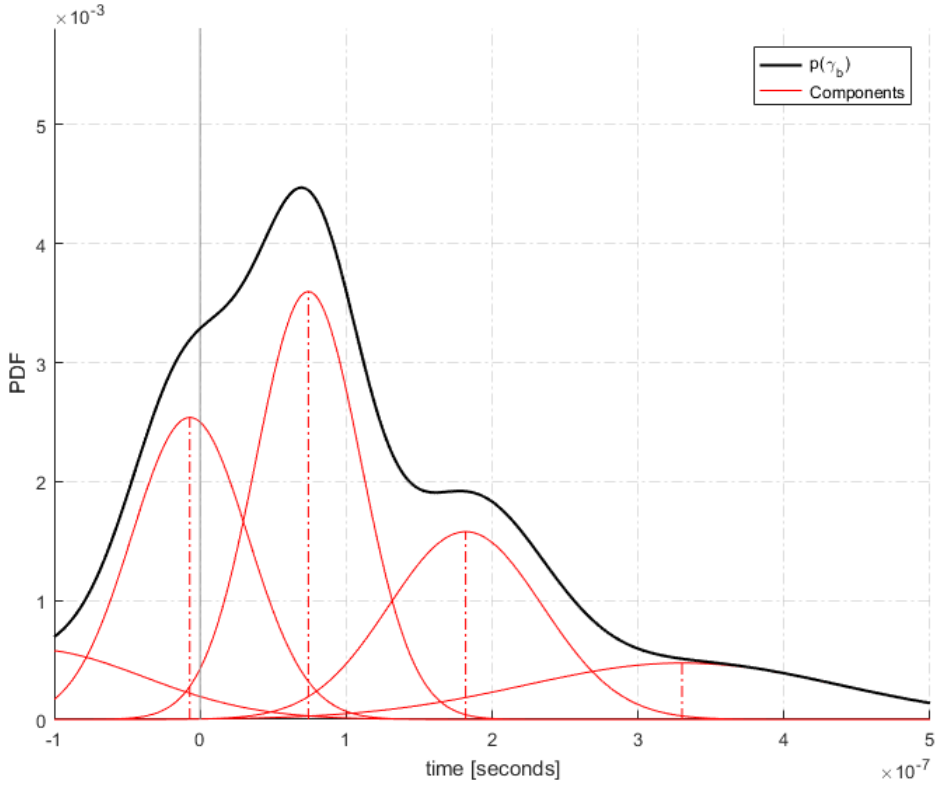


Fig. 4.11: $p(\gamma_b)$ for BS 1 (iteration 3, $R_b = 5$).

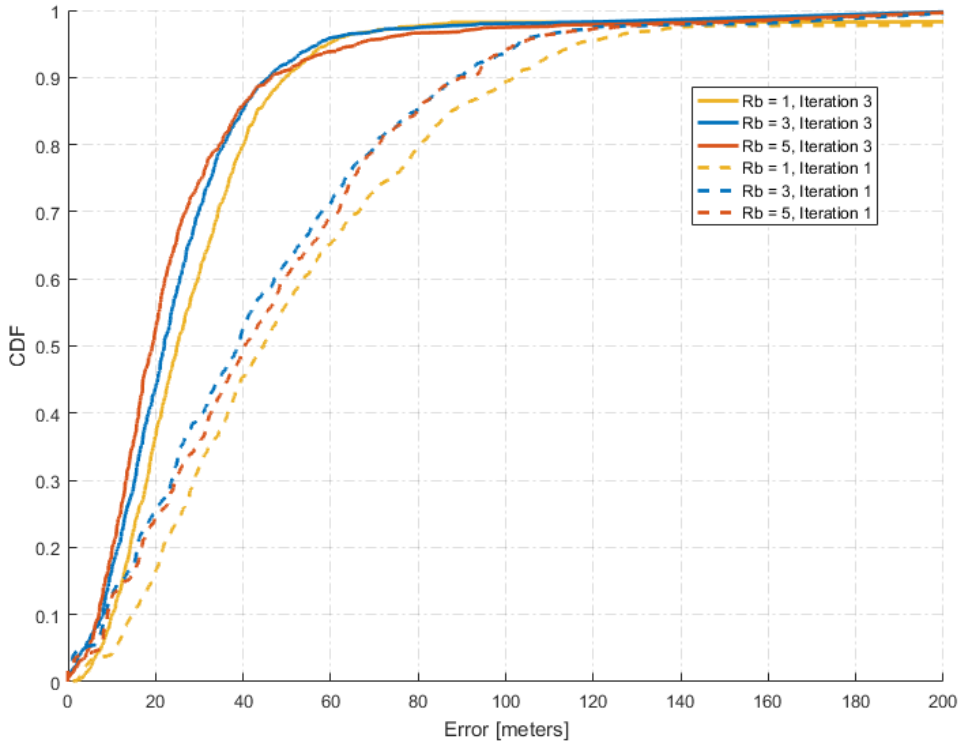


Fig. 4.12: CDF at first and third iteration for $R_b = 1$, $R_b = 3$, $R_b = 5$.

4.3 Comparison with TDOA

The learning algorithm was compared with a TDOA multilateration algorithm. TDOA multilateration uses three ToA to generate two hyperbolas and find the estimated position as intersection of these hyperbolas. Here TDOA uses only the three smallest ToA, corresponding to the three closest BS. For an honest comparison the LA was tested with both four and three BS (in this case all base stations are used for the learning but only three are used for the position MAP estimate). Figure 4.13 presents the CDF compared to the LA when 4 BS are used, while figure 4.14 when only 3 BS are used. Figures 4.15, 4.16 present the comparison of position error mean and standard deviation over multiple iterations. By iteration three the LA outperforms the TDOA multilateration (even if only three BS are used). The mean error and standard deviation at iteration three are presented in table 4.2.

Position error, iteration 3			
	TDOA	LA, 4 BS	LA, 3 BS
Error mean [meters]	46.9	28.4	31.4
Error std dev [meters]	44.5	32.6	26.5

Table 4.2: Position error mean and standard deviation for TDOA, LA with 4 BS and LA with 3 BS (iteration 3, $R_b = 3$).

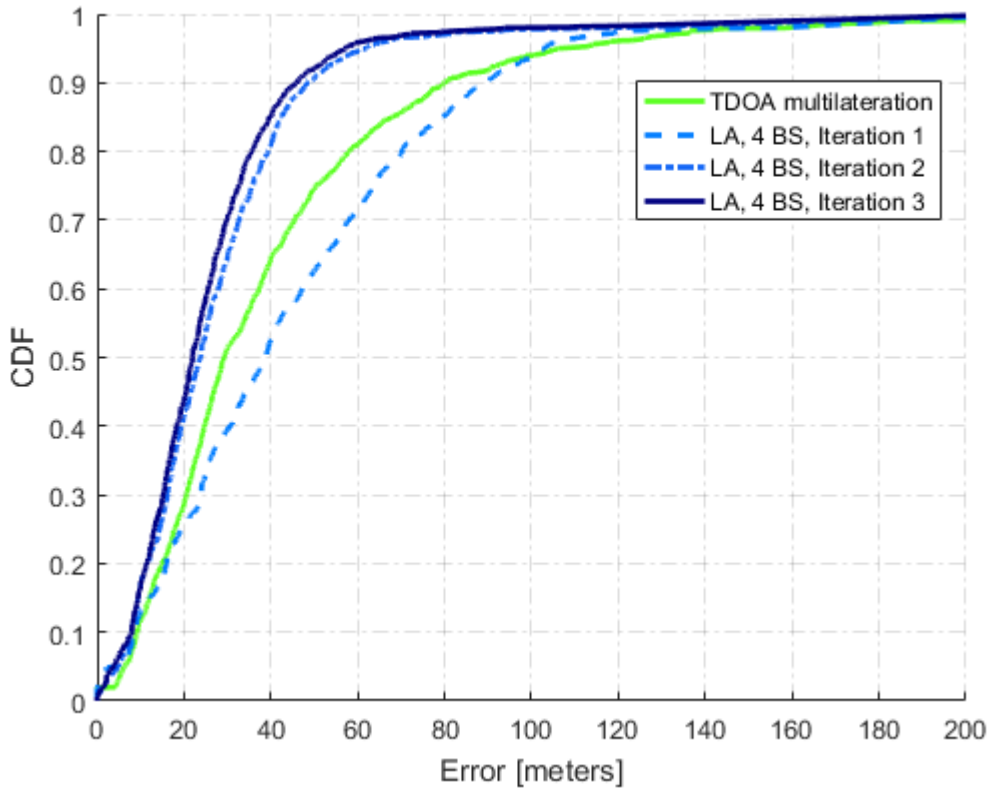


Fig. 4.13: CDF of TDOA and Learning Algorithm with 4 BS ($R_b = 3$).

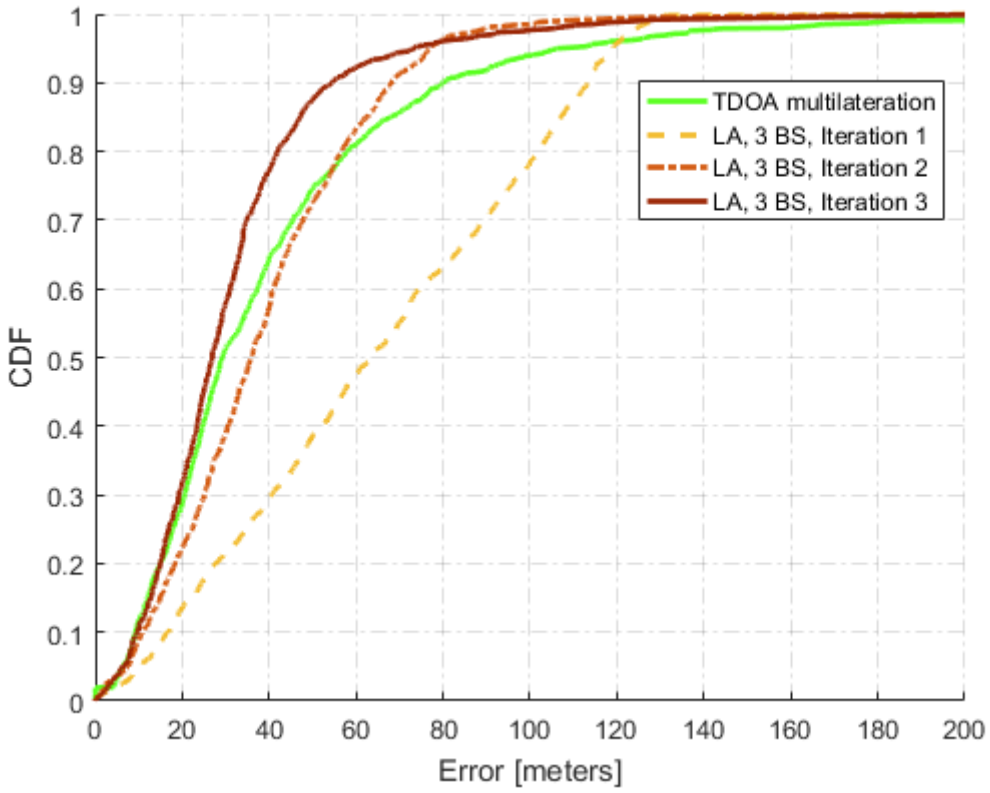


Fig. 4.14: CDF of TDOA and Learning Algorithm with 3 BS ($R_b = 3$).

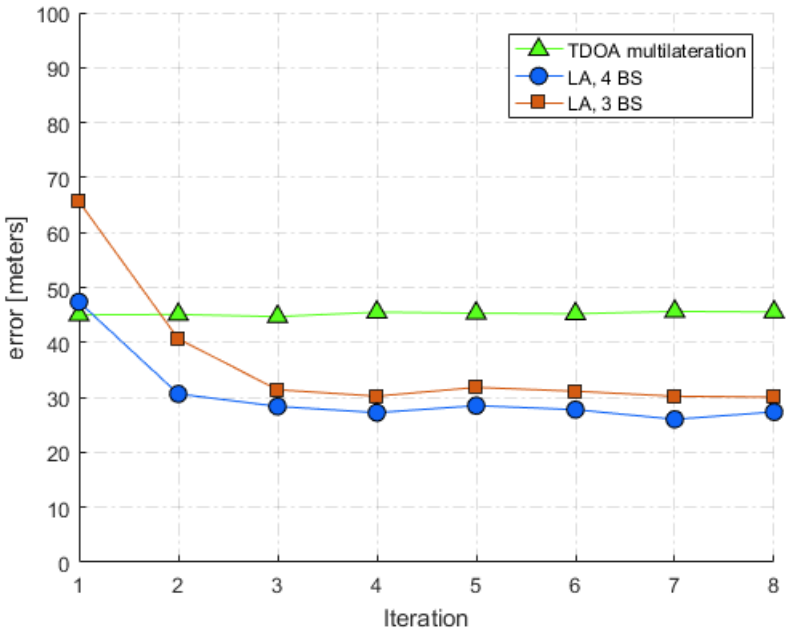


Fig. 4.15: Mean position error at each iteration for TDOA and LA ($R_b = 3$).

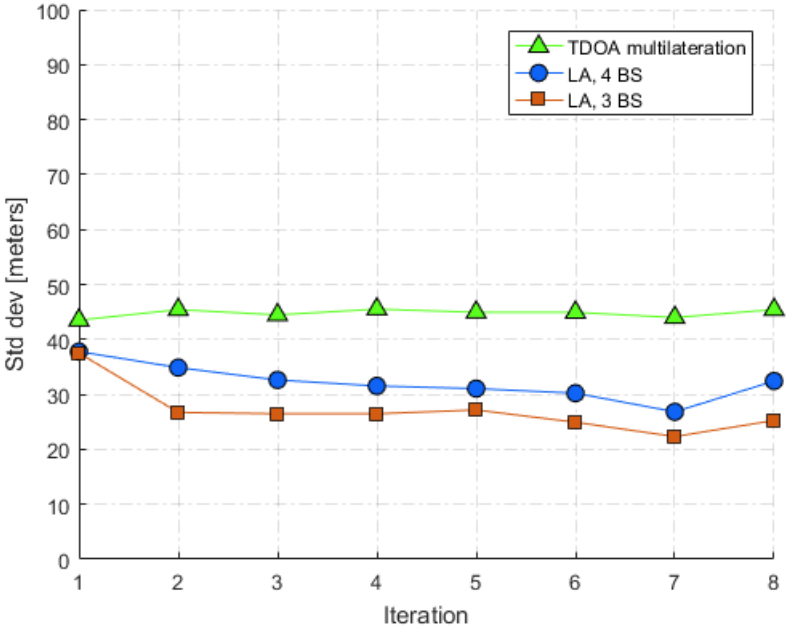


Fig. 4.16: Position error standard deviation at each iteration for TDOA and LA ($R_b = 3$).

4.4 Range-Aware algorithm results

The Range-Aware Learning Algorithm (RALA) was compared with the normal Learning Algorithm (LA). In this simulation we let the parameter α vary over distance as explained in the previous chapter. The Range-Aware algorithm uses multiple distributions for each base stations. This can be seen in figures 4.17, 4.18, 4.19 where the distributions change between a region and another. In this environment the normal algorithm is losing information by using only one distribution for each base station. Figure 4.20 presents the CDF of the two algorithms. Table 4.3 presents the position error precision at iteration three.

Position error, iteration 3		
	LA	RALA
Error mean [meters]	33.0	28.2
Error std dev [meters]	33.9	31.2

Table 4.3: Position error mean and standard deviation for RALA and LA (iteration 3, $R_b = 3$).

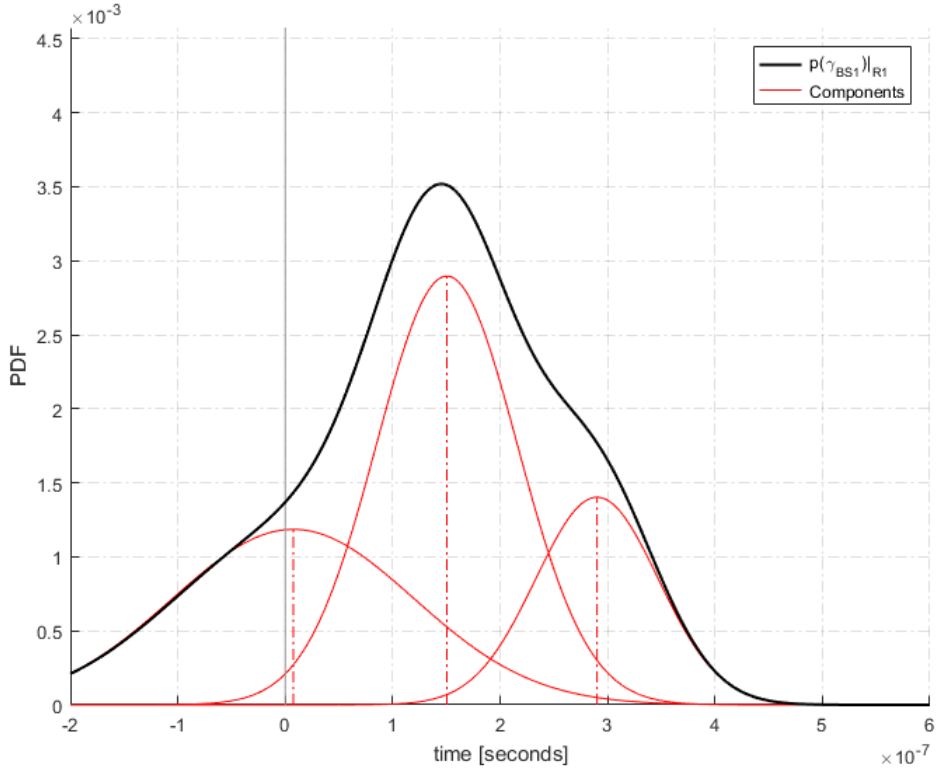


Fig. 4.17: $p(\gamma_b)$ for BS 1 in Region 1 (iteration 3, $R_b = 3$).

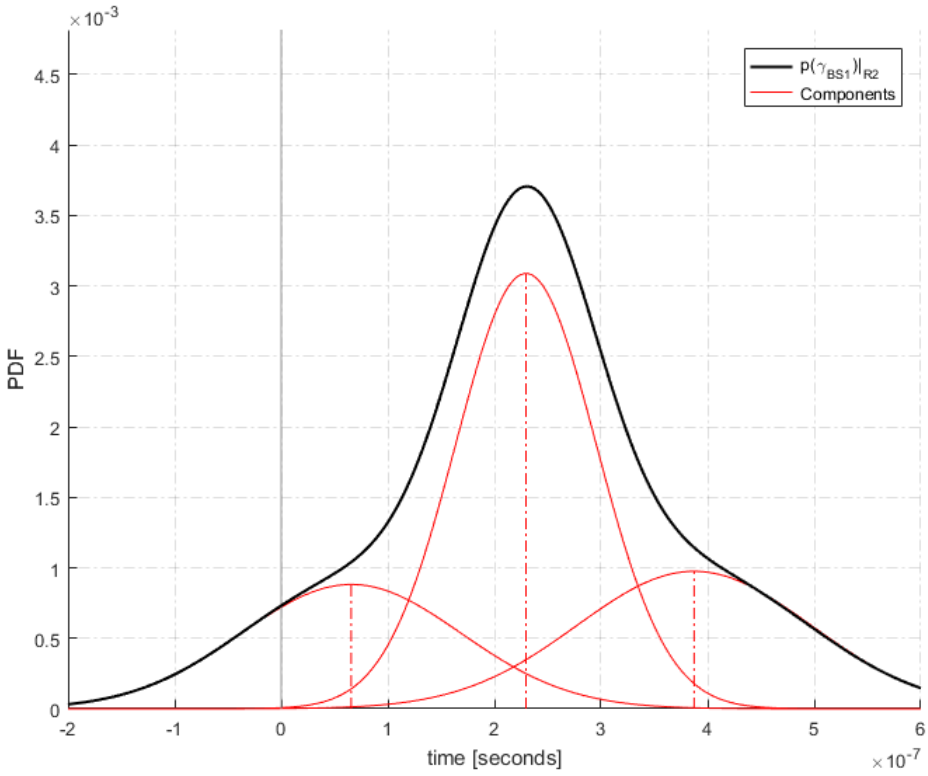


Fig. 4.18: $p(\gamma_b)$ for BS 1 in Region 2 (iteration 3, $R_b = 3$).

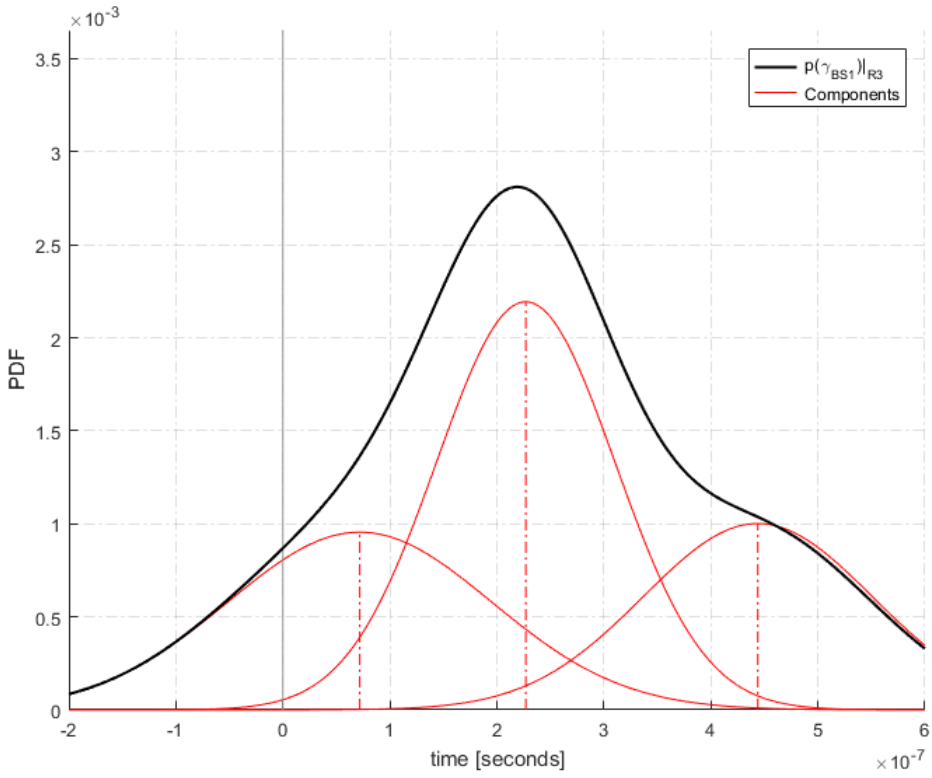


Fig. 4.19: $p(\gamma_b)$ for BS 1 in Region 3 (iteration 3, $R_b = 3$).

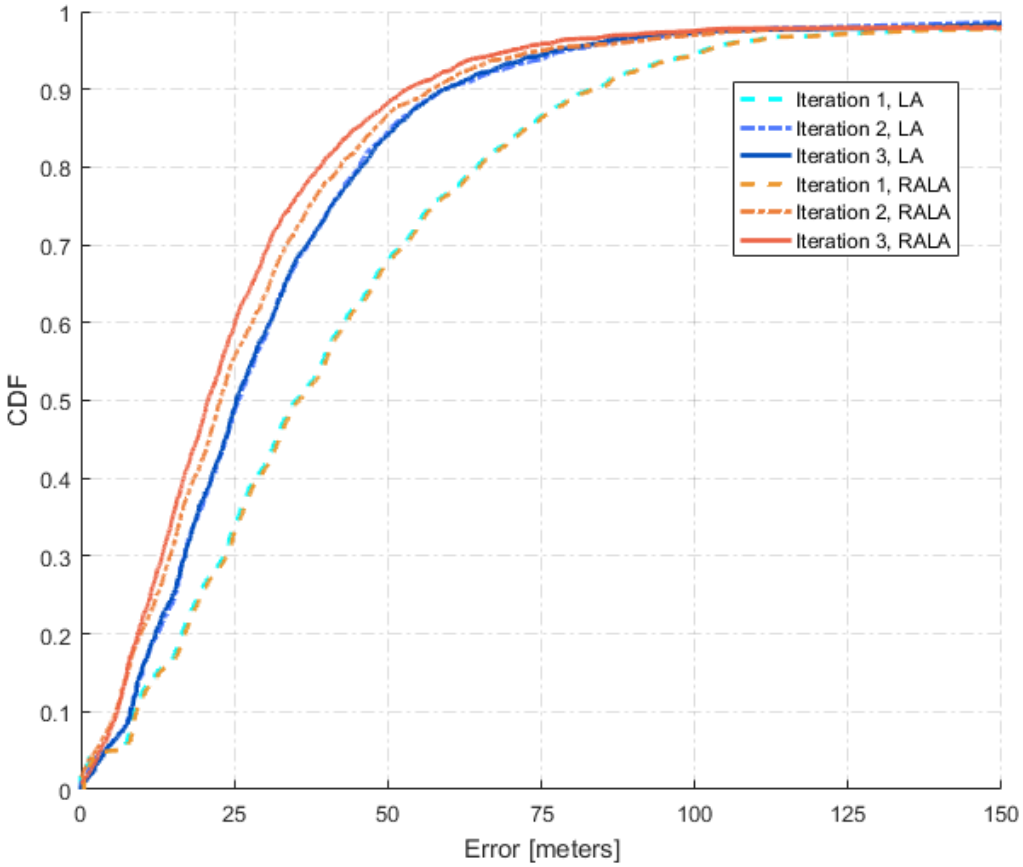


Fig. 4.20: CDF of the Range-Aware LA and normal LA.

Chapter 5

conclusion

The simulations proved that the learning algorithm is effective in improving the position estimation. The mean position error was reduced by 40% in respect to the TDOA multilateration, reducing the variance at the same time. The Range-Aware variation proved to be effective if the channel bias changes over space, reducing the mean position error by an additional 15% in respect to the normal learning algorithm. The results showed also that three base stations are enough to obtain good measurements if the learning is already performed.

One important feature that was found is the fast learning speed, in fact 2000 samples divided in two iterations are enough to reach convergence. Moreover it is important to remind the fact that a real location server using this algorithm needs to perform the learning only once in a while, either to learn the parameters for the first time or to recalibrate them if it is necessary. The learned parameters can be memorized and then used without the need to keep track of all the measurments collected during the learning.

5.0.1 Future works

The approach presented in this work can be extended in a plethora of ways and adapted to many different scenarios. In first place the algorithm, being based on learning and using little to no assumption about the network, is prone to adapt to different environments and conditions. Secondly the

Range-Aware variant opens the way to even more capabilities and possible extensions. In this work we opted for range-based regions, but it is possible to define angle-based regions, or even better, a combination of the two. Furthermore it is possible to exploit machine learning even more by letting the definition of these regions to the algorithm itself. One approach to achieve this would be by using clustering of the channel bias over space instead of clustering over time (the one used in this work). This approach can be adapted to IoT and UWB signals where frequent reflections easily degrade the position estimation.

Other possible extensions can be done to the probabilistic framework of the algorithm. Including for example a movement tracking feature and consequently a position predictor that can be used to weight the next position estimates.

Bibliography

[1] W. A. Hapsari, A. Umesh, M. Iwamura, M. Tomala, B. Gyula, and B. Sebire, "Minimization of drive tests solution in 3GPP," *IEEE Communications Magazine, vol. 50, no. 6*, p. 28–36, 2004.

[2] J. Johansson, W. Hapsari, S. Kelley, and G. Bodog, "Minimization of drive tests in 3GPP release 11," *IEEE Communications. Magazine, vol. 50, no. 11*, p. 36–43, 2012.

[3] I. Guvenc and C. C. Chong, "survey on ToA based wireless localization and NLOS mitigation techniques," *IEEE Communications Survey & Tutorials*, vol. 11, no. 3, pp. 107–124, 2009.

[4] J. A. del Peral-Rosado, J. A. López-Salcedo, F. Zanier, and M. Crisci, "Achievable localization accuracy of the positioning reference signal of 3GPP LTE," *012 International Conference on Localization and GNSS, Starnberg*, pp. 1–6, 2012.

[5] J. A. D. Peral-Rosado, "Performance analysis of hybrid GNSS and LTE Localization in urban scenarios," *2016 8th ESA Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing (NAVITEC)*, pp. 1–8, 2016.

[6] S. A. Ahson and M. Ilyas, "Location-based services handbook: Applications, technologies, and security," *CRC Press*, 2010.

[7] C. Jeffrey, "An introduction to GNSS, GPS, GLONASS, Galielo and other Global Navigation Satellite Systems," 2010.

[8] F. Perez-Cruz, C.-K. Lin, and H. Huang, "Blade: A universal, blind learning algorithm for ToA localization in NLOS channels," *IEEE Globecom Workshops (GC Wkshps), Washington, DC*, pp. 1–7, 2016.

[9] J. A. del Peral-Rosado, J. A. López-Salcedo, G. Seco-Granados, F. Zanier, and M. Crisci, "Evaluation of the LTE positioning capabilities under typical multipath channels," *2012 6th Advanced Satellite*

Multimedia Systems Conference (ASMS) and 12th Signal Processing for Space Communications Workshop (SPSC), pp. 139–146, 2012.

[10] J. Riba and A. Urrela, "A non-line-of-sight mitigation technique based on ML-detection," *IEEE Int. Conf. Acoustic Speech and Signal Processing (ICASSP)*, p. 153–156, 2004.

[11] B. Chen, "OTDOA or UTDOA? performance and implementation on lbs strategy," *Proceedings of the 2nd Invitational Workshop on Opportunistic RF localization for Next Generation Wireless Devices*, June 2010.

[12] 3GPP TSG RAN1 #56bis., "Study on hearability of reference signals in LTE positioning support," *TDoc R1-091336*, 2009.

[13] ETSI TS. 136 211 V13.1.0 (2016-03) LTE Technical Specification, "Evolved universal terrestrial radio access (E-UTRA); physical channels and modulation,"

[14] S. Fischer, "Introduction to OTDOA on LTE network," *Qualcomm*, 2014.

[15] M. Panchetti, C. Carbonelli, M. Horvat, and M. Luise, "Performance analysis of PRS-based synchronization algorithms for LTE positioning applications," *2013 10th Workshop on Positioning, Navigation and Communication (WPNC), Dresden*, pp. 1–6, 2013.

[16] J. Medbo, I. Siomina, A. Kangas, and J. Furuskog, "Propagation channel impact on LTE positioning accuracy: A study based on real measurements of observed time difference of arrival," *2009 IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications, Tokyo*, pp. 2213–2217, 2009.

[17] S. Hu, A. Berg, X. Li, and F. Rusek†, "Improving the Performance of OTDOA based Positioning in NB-IoT System," 2017.

[18] Technical Specification Group Radio Access Network, "Stage 2 functional specification of User Equipment (UE) positioning in UTRAN," *3GPP TS 25.305*, 2006.

[19] A. Kangas and T. Wigren, "Angle of arrival localization in LTE using MIMO pre-coder index feedback," *IEEE Communications Letters, vol. 17, no. 8*, pp. 1584–1587, 2013.

[20] C.-Y. Chen and W.-R. Wu, "Three-dimensional positioning for LTE systems," *IEEE Transactions on vehicular technology, vol. 66, no. 4*, pp. 3220 – 3234, 2017.

[21] C. H. Chen and K. T. Feng, "Statistical distance estimation algorithms with RSS measurements for indoor LTE-A networks," *IEEE Transactions on Vehicular Technology, vol. 66, no. 2*, pp. 1709–1722, 2017.

[22] C. H. Chen and K. T. Feng, "Enhanced distance and location estimation for broadband wireless networks," *IEEE Transactions on Mobile Computing, vol. 14, no. 11*, pp. 2257–2271, 2015.

[23] S. Bartelmaos, K. Abed-Meraim, and R. Leyman, "General selection criteria to mitigate the impact of NLoS errors in RTT measurements for mobile positioning," *IEEE International Conference on Communications, Glasgow*, pp. 4674–4679, 2007.

[24] D. Milic and T. Braun, "Optimizing dimensionality and accelerating landmark positioning for coordinates based RTT predictions," *Broadband Communications, Networks and Systems 2007. BROADNETS 2007. Fourth International Conference on, Raleigh, NC, USA*, pp. 631–640.

[25] G. Çelik, H. Çelebi, and G. Tuna, "A novel RSRP-based E-CID positioning for LTE networks," *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC), Valencia*, pp. 1689–1692, 2017.

[26] 3GPP TSG RAN WG1 #56bis, "Discussion on positioning support in LTE," *TDoc R1-091228*, 2009.

[27] T. Sathyan, M. Hedley, and M. Mallick, "An analysis of the error characteristics of two time of arrival localization techniques," *Proceedings of the 13th International Conference on Information Fusion*, 2010.

[28] P. C. Chen, "A non-line-of-sight error mitigation algorithm in localization estimation," *IEEE Int. Conf. Wireless Communications and Networking (WCNC)*, pp. 316–320, 1999.

[29] S. Venkatesh and R. M. Buehrer, "A linear programming approach to NLOS error mitigation in sensor networks," *IEEE Int. Symp. Information Processing in Sensor Networks (IPSN)*, p. 301–308, 2006.

[30] V. Dizdarevic and K. Witrisal, "On impact of topology and cost function on lse position determination in wireless networks," *Workshop on Positioning, Navigation and Communication (WPNC)*, p. 129–138, 2006.

[31] J. Medbo, I. Siomina, A. Kangas, and J. Furuskog, "Propagation channel impact on LTE positioning accuracy: A study based on real measurements of observed time difference of arrival," *IEEE Int. Symp. Personal, Indoor and Mobile Radio Communications (PIMRC)*, p. 2213 – 2217, 2009.

[32] B. Altintas and T. Serif, "Improving RSS-Based Indoor Positioning Algorithm via K-Means Clustering," *17th European Wireless 2011 - Sustainable Wireless Technologies*, pp. 1–5, 201.

[33] D. M. Blei and M. I. Jordan, "Variational inference for dirichlet process mixtures," *Bayesian Anal. 1, no. 1*, pp. 121–143, 2006.

[34] R. Sridharan, "Gaussian mixture models and the EM algorithm," 2014.

[35] K. Hassan, T. A. Rahman, M. R. Kamarudin, and F. Nor, "The mathematical relationship between maximum access delay and the R.M.S delay spread," *ICWMC 2011 : The Seventh International Conference on Wireless and Mobile Communications*, 2011.

Acronyms

BS Base Station.

CSR Cell-specific Reference Signal.

GNSS Global Navigation Satellite System.

LA Learning Algorithm.

LOS Line Of Sight.

LPP LTE Positioning Protocol.

LS Location Server.

LTE Long Term Evolution.

MAP Maximum A Posteriori.

NLOS Non Line Of Sight.

OTDOA Observed Time Difference Of Arrival.

PCI Physical Cell Identity.

PRS Positioning Reference Signal.

RALA Range-Aware Learning Algorithm.

RS Reference Signal.

RSTD Reference Signal Time Difference.

RTD Relative Time Difference.

SINR Signal To Interference plus Noise Ratio.

TDOA Time Difference Of Arrival.

ToA Time of Arrival.

UE User Equipment.

UWB Ultra Wide Band.

Acknowledgments

I want to truly thank prof. Zanella for his insightful advices during the last steps of the thesis and for being particularly understanding during these months.

A special "thank you" go to my parents who always supported me during my studies. I'm thankful and proud of having them.

Final thanks go to Demet, for her support, and to the friends that have been in my thoughts even if I haven't seen them in a while: Piero, Giacomo, Federico G., Laura, Federico C., Enrica, Leo and all the others from the old telecommunication group.