Robust Indoor Positioning in WLAN Networks

Juan Manuel Castro-Arvizu

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

Robust Indoor Positioning in WLAN Networks

Thesis for the Degree of Doctor of Philosophy

Juan Manuel Castro-Arvizu

Department of Signal Theory and Communications
Universitat Politècnica de Catalunya

Academic advisors:
Prof. Juan A. Fernández Rubio[1]
Dr. Pau Closas[2]

[1]Department of Signal Theory and Communications
Universitat Politècnica de Catalunya
Barcelona, Spain

[2]Department of Electrical & Computer Engineering
Northeastern University. College of Engineering
Boston MA

Barcelona, Spain, 2017

Robust Indoor Positioning in WLAN Networks

Short abstract:

Due to the vast increase of location-based services, currently there exists an actual need of robust and reliable indoor localization solutions. Received Signal Strength (RSS) for indoor localization is widely used due to its simplicity and availability in most mobile devices. The path-loss channel model for RSS measurements is usually parameterized by the propagation losses and the shadow fading. These parameters might vary over time because of changes in the environment. In this thesis, the problem of tracking a mobile node in an indoor environment is addressed while an online estimation of the path loss exponents and shadowing of the channel model. For this goal, different methodologies and technologies are presented in this Thesis. Numerical simulations results in realistic scenarios are provided to support the theoretical discussion and to show the enhanced performance for the new robust indoor localization approach. Additionally, experimental results using real data are reported to validate computational results.

Keywords: Indoor localization, robust filtering, two-slope path loss model, channel model calibration, received signal strength, channel model calibration, Bayesian filtering

Dedicated to my family and Mexico

List of Papers

I J. Manuel Castro-Arvizu, Pau Closas, J. A. Fernández-Rubio
Cramér-Rao lower bound for breakpoint distance estimation in a path-loss model, Proceedings of IEEE International Conference on Communications (IEEE ICC 2014), 10-14 June 2014, Sidney Australia. Pages: 176 - 180, DOI: 10.1109/ICCW.2014.6881192. URL: http://ieeexplore.ieee.org/document/6881192/

II J. Manuel Castro-Arvizu, J. Vilà-Valls, Pau Closas, J. A. Fernández-Rubio
Simultaneous Tracking and RSS Model Calibration by Robust Filtering, Asilomar Conference on Signals, Systems and Computers November 2014, Pacific Grove CA. Pages: 706 - 710, DOI: 10.1109/ACSSC.2014.7094539 URL: http://ieeexplore.ieee.org/document/7094539/

III J. Manuel Castro-Arvizu, Ana Moragrega, Pau Closas, Juan A Fernandez-Rubio
Assessment of RSS Model Calibration with Real WLAN Devices, The Twelfth International Symposium on Wireless Communication Systems August 2015, Brussels, Belgium. Pages: 746 - 750, DOI: 10.1109/ISWCS.2015.7454449 URL: http://ieeexplore.ieee.org/document/7454449/

Attention¡

Pages VII-XXIII of the thesis, containing the texts mentioned above, should be consulted on the web pages of the respective publishers

IV J. Manuel Castro-Arvizu, J. Vilà-Valls, Pau Closas, J. A. Fernández-Rubio
RSS-based Indoor Localization Using a Robust IMM-EKF Algorithm, International Journal of Distributed Sensor Networks. Hindawi Publishing Corporation. *Accepted in December 2016.*

V J. Manuel Castro-Arvizu, Pau Closas, J. A. Fernández-Rubio
Bayesian Calibration for a RSS channel model using Real Data, Wireless Communications and Mobile Computing, Hindawi Publishing Corporation. *Submitted in January 2017.*

Abstract:

Navigation and location technologies have been reaching in a major interest where Global Navigation Satellite System (GNSS) is mostly adopted. The limitation of this technology is that direct sky view is needed for reliable positioning. In indoor environments, however, it is difficult for GNSS technology to provide a reliable performance in positioning due to the signal attenuation and blocking caused by buildings and construction materials. For this reason, the growth in indoor applications has focused the research in new techniques for attempting mitigate the poor GNSS performance on this type of environments.

In the context of indoor positioning, multitude of emerging technologies for localization based on ultrasound, infrared, Ultra Wide Band (UWB), Zigbee, inertial navigation and other non-GNSS technologies have been proposed but special equipment is required and a large number of signal sources are needed. However, Wireless Local Area Network (WLAN) technology is widely used in indoor positioning. While the same requirements are also needed as the other technologies in order to improve the positioning accuracy, in terms of cost and ability, Wireless-based indoor location is widely used due to the already deployment of Anchor Points (AP) in urban and indoor areas.

There are several methods for indoor positioning purposes e.g ToA (Time of Arrival), Received Signal Strength (RSS) measurements, AoA (Angle of Arrival), fingerprinting and so on. Most of the network-based location estimations use RSS measurements because almost all mobile devices are afforded to use this type of measurements. So, this thesis is centered in WLAN RSS-based positioning systems.

Although fingerprinting technique is widely used, WLAN-based positioning systems based on fingerprinting techniques has some disadvantages, such as low positioning accuracy due to the RSS offset between reference and user devices and long-duration fingerprinting updates. Hence, this thesis is focused only on geometric or statistical techniques that are based on previous knowledge of the radio propagation channel model.

The first step for indoor positioning is the distance estimation between the user and the AP. Theoretical and empirical propagation channel models are used to translate the difference between the transmitted and Received Signal Strength into an estimated range.

A Propagation channel model built the radio map and also report changes in the environment. There are several models in the literature to characterize this channel. Indoor RSS-based localization has become a popular solution, but standard techniques still consider a time invariant simple single slope path loss channel model with a priori known constant channel parameters. While some contributions considered the RSS-based localization problem using a single path loss model with unknown parameters, the general solution that considers

a generalized distance dependent measurement model is an important missing point.

While the single path loss model is adequate in free space propagation, a multi-slope piece-wise linear propagation model appears more suitable in indoor environments and in the presence of strong signal reflections. This thesis considers the two-slope channel model and proposes a robust indoor positioning solution based on a parallel architecture using a set of Interacting Multiple Models (IMM), each one involving two Extended Kalman filters (EKF) and dealing with the estimation of the distance to a given AP. Within each IMM, the two-slope path loss model parameters are sequentially estimated with Maximum Likelihood Estimate (MLE) to provide a robust solution. Finally, the set of distance estimates are fused into a standard EKF-based solution to mobile target tracking.

In addition, the benchmarks derived in this thesis to evaluate the performance of our IMM-EKF algorithm are the Cramér Rao Lower Bound (CRLB) and the Posterior Cramér Rao Lower Bound (PCRLB) providing a guidance in the improvement of the experimental design. The CRLB is used to assess the estimation of model parameters and the PCRLB for tracking solution. This, combined with a path-loss exponent estimation, the channel calibration algorithm is validated with an online range estimation.

The central theme throughout this thesis is to develop a completely online two-slope channel calibration and, simultaneously, a mobile target tracking algorithm. The performance of the method is assessed through realistic computer simulations and validated with real RSS measurements obtained from experimental tests in a typical office environment

---

Resumen:

Las tecnologías en navegación y localización han estado obteniendo un gran interés en los últimos años donde el Sistema Global de Navegación por Satélite (Global Navigation Satellite System, GNSS) aparece como el más utilizado para estos fines. No obstante, una de las limitaciones del GNSS es la necesidad de tener una visión directa al cielo para así garantizar un posicionamiento bastante fiable. También, al utilizar solamente tecnología GNSS en espacios interiores (más conocidos en el mundo científico por entornos *indoor*), se es complicado conseguir un buen desempeño en términos de posicionamiento debido a la atenuación e interferencia de la señal causada por los edificios y materiales de construcción. Por esta razón, y debido al crecimiento en aplicaciones dentro de entornos *indoor*, la investigación de nuevas tecnologías para posicionamiento en interiores se ha centrado en intentar mitigar el mal desempeño de la tecnología GNSS en este tipo de ambientes.

En el contexto de posicionamiento en interiores (indoor positioning), se han propuesto multitud de tecnologías emergentes para localización basadas en ultrasonido, infrarrojo, Banda Ultra Ancha (Ultra Wide Band UWB), Zigbee, navegación inercial y otras tecnologías que no sean GNSS. Sin embargo, se requiere de equipo especial y un gran número de fuentes de señal. A pesar de ello, la tecnología en Redes de Área Local Inalámbricas (Wireless Local Area Network WLAN) es ampliamente utilizada en el posicionamiento en interiores. Aunque la tecnología WLAN tenga los mismos requerimentos que el resto de tecnologías, en términos de coste y practicidad, los sistemas de posicionamiento basados en redes inalámbricas se utilizan con mayor frecuencia debido al ya existente despliegue de estaciones base (AP) en áreas urbanas e interiores.

Existen varias técnicas que sirven para fines de posicionamiento en interiores. Por ejemplo, utilizando el tiempo de llegada de la señal (time of arrival TOA), las mediciones de la potencia de la señal recibida (Received Signal Strength, RSS), el ángulo de llegada (Angel of Arrival AoA), la técnica *fingerprinting* entre otras. La mayoría de las estimaciones de posicionamiento usan mediciones RSS porque muchos de los dispositivos móviles que existen en el mercado actual utilizan este tipo de medidas. Por lo tanto, esta tesis está centrada en sistemas de posicionamiento basados en mediciones WLAN-RSS.

Aunque la técnica *fingerprinting* es tambieén ampliamente utilizada. Los sistemas de posicionamiento basados en WLAN que utilizan la técnica de *fingerprinting* tienen algunas desventajas, como una baja precisión en el cómputo de la posición debido al offset existente en la potencia de la señal recibida entre los dispositivos de referencia utilizados y la larga duración en las actualizaciones de la base de datos del *fingerprinting*. Por conseguiente, esta tesis se centra sólo en técnicas geométricas o estadísticas basadas en el conocimiento previo del modelo de radio propagación.

El primer paso para el posicionamiento en interiores es estimar la distancia entre el usuario (o móvil) y el AP. Los modelos teóricos y empíricos de canales de

atenuación de interiores se utilizan para convertir la diferencia entre la intensidad de la señal recibida y transmitida en una distanca estimada.

Un modelo de canal de atenuación de interiores contruye un mapa de cobertura y también es capaz de reportar los cambios en el entorno *indoor*. El posicionamiento *indoor* basado en mediciones RSS se ha convertido en una solución bastante popular, pero las técnicas comunes consideran un modelo de pérdidas por trayectoria de una pendiente (one-single path loss channel model), invariante en el tiempo y con un conocimiento previo de los parámetros del canal que se consideran constantes.

Aunque el modelo de atenuación de una pendiente se adecua para la propagación en el espacio libre, un modelo de pérdidas por trayectoria de múltiples pendientes es más adecuado en entornos *indoor* y también cuando hay grandes reflexiones de la señal. Esta tesis considera el modelo de pérdidas por trayectoria de pendiente dual (two-slope path loss channel model) y propone una solución robusta para posicionamiento en interiores basado en una arquitectura paralela conformada por un conjunto de algoritmos de Interacción de Múltiples Modelos (Interacting Multiple Models IMM) donde cada IMM involucra dos Filtros de Kalman Extendidos (Extended Kalman Filter EKF) para el proceso de estimación de la distancia entre el AP y el usuario. Dentro de cada IMM, los parametros del modelo de pérdidas por trayectoria de pendiente dual se estiman secuencialmente utilizando la estimación por máxima verosimilitud (Maximum Likelihood Estimate MLE) y así proveer una solución robusta. Finalmente, el conjunto de distancias estimadas se fusionan en un EKF para tener una solución final de la posición del usuario.

Además, las cotas de referencias que son derivadas en esta tesis y que sirven para evaluar el rendimiento del algoritmo IMM-EKF son la Cota Inferior de Cramér Rao (Cramér Rao Lower Bound CRLB) y la Cota Inferior de Cramér Rao Posterior (Posterior Cramér Rao Lower Bound PCRLB) que servirán de guía para el perfeccionamiento del diseño experimental. El CRLB se utiliza para evaluar la estimación de los parámetros del modelo del canal y el PCRLB para la solución final de posicionamiento. Ésto, combinado con la estimación del exponente de pérdidas por trayectoria, el algoritmo de calibración de canal se valida con una estimación de la distancia de usuario hacia el AP en forma *online*.

El tema central de esta tesis es desarrollar un algoritmo online para posicionamiento indoor que simultáneamente sea capaz de hacer la calibración del canal de propagación. El desempeño del método se evalúa mediante simulaciones por computadora que se validan con mediciones RSS reales obtenidas a partir de pruebas experimentales.

Contents

List of Figures

List of Tables

List of Acronyms and Abbreviations

pdf	Probability Density Function
AP	Access Point
CDF	Cumulative Probability Function
CKF	Cubature Kalman Filter
CRLB	Cramér Rao Lower Bound
dB	Decibel
DCM	Direct Cosine Matrix
EKF	Extended Kalman Filter
FIM	Fisher Information Matrix
GDoP	Geometric Dilution of Precision
GNSS	Global Navigation Satellite Systems
IMM	Interacting Multiple Model
IMU	Inertial Measurement Unit
INS	Inertial Navigation System
IR	Infrared
KF	Kalman Filter
LBS	Location-Based Service
LOS	Line-of-Sight
MCMC	Markov Chain Monte Carlo
MLE	Maximum Likelihood Estimation
PCRLB	Posterior Cramér Rao Lower Bound
PF	Particle Filter
QKF	Quadrature Kalman Filter
RF	Radio Frequency
RFID	Radio-Frequency Identification
RMSE	Root-Mean-Square Error
RSS	Received Signal Strength
RToF	Roundtime Time-of-Flight
SoC	system on a chip
TDoA	Time Difference-of-Arrival
ToA	Time-of-Arrival
UKF	Unscented Kalman Filter
UWB	Ultra Wide Band

WLAN Wireless Local Area Networks

List of Symbols

$\mathbb{R}$	Set of Real numbers
$\mathbb{E}\{\cdot\}$	Expected value
$\mathbf{x}$	A vector
$\mathbf{X}$	A matrix
$\boldsymbol{f}(\cdot)$	A vector function
$\mathcal{N}$	Identifier for Normal or Gaussian probability density function.
Γ	Identifier for Gamma distribution
$\mathcal{U}$	Identifier for Uniform distribution
$\mathbf{x} \sim \mathcal{N}(\mathbf{x}\|\boldsymbol{\mu}, \boldsymbol{\Sigma})$	Shortcut for $\boldsymbol{x}$ is Gaussian with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$
$\hat{\boldsymbol{x}}$	An estimate of the random variable $\mathbf{x}$
$\boldsymbol{\nu}, \boldsymbol{\omega}$	Random perturbations or noises
α_x	Path Loss Parameter
σ_x	Standard deviation of the variable $\mathbf{x}$
$\chi_{\sigma_x^2}$	Random variable with distribution $\mathcal{N}(0, \sigma_x^2)$
$\nabla_{\mathbf{f(x)}}$	Gradient vector of $\mathbf{f}(\mathbf{x})$ with respect to $\mathbf{x}$
$\Delta^{\mathbf{f(x)}}_{\mathbf{f(x)}}$	Hessian Matrix of $\mathbf{f}(\mathbf{x})$ with respect to $\mathbf{x}$
$\mathbf{H}_{\boldsymbol{\theta}}$	Jacobian matrix $\mathbf{h}(\boldsymbol{\theta})$ with respect to $\boldsymbol{\theta}$
$\mathbf{x}^\top$, $\mathbf{X}^\top$	Transpose of $\mathbf{x}$; of $\mathbf{X}$
$\mathbf{X}^{-1}$	Inverse of $\mathbf{X}$
$\mathcal{I}$	Fisher Information Matrix (FIM)
$\log(\cdot)$	Natural Logarithm
g	Gravity, $g = 9.807 \frac{m}{s^2}$
c	Speed of light, $c = 299.792.458 \frac{m}{s}$

Introduction

In everyday life, Global Navigation Satellite Systems (GNSS) are widely used for positioning and navigation purposes [7, 8, 9, 10, 11]. Generally, GNSS is a key element in a multitude of outdoor applications. The limitation of this technology is that direct sky view is needed for reliable positioning. As a consequence, GNSS cannot satisfy the high accuracy positioning requirements of many applications in engineering and mining surveying, structural monitoring, urban or indoor positioning.

In the context of indoor positioning, there have been a multitude of emerging technologies for localization based on WLAN (IEEE 802.11x), UWB (IEEE 802.15.4), Zigbee and other non-GNSS technologies. This thesis is focused on using signals of the IEEE 802.11x as the primary source of information to approach the localization problem due to the inexpensive hardware and the already dense deployment of WLAN Anchor Points (APs) in urban and indoor areas.

Several techniques are widely used for this goal in literature. The type and quality of measurements have a considerable effect on the performance of a positioning algorithm. Different types of measurements have been considered for the positioning problem, e.g., Received Signal Strength (RSS), Angle of Arrival (AOA), Time of Arrival (TOA), and Time Difference Of Arrival (TDOA). Designing an estimator for the positioning problem strongly depends on the model of measurements, therefore it is of great importance to use an accurate model [12, 13, 14].

This thesis is mainly focused on RSS localization due to its availability in most mobile devices. The RSS channel model is characterized by the propagation losses and shadow fading, modeling weak signal conditions and signal blockage scenarios.

Although fingerprinting techniques are widely used [15, 16], this thesis is focused on geometric or statistical techniques [17] that are based on a certain knowledge of the radio propagation channel model.

There are several channel models in the literature to characterize the indoor propagation environment [18, 19, 20, 21]. A propagation model represents the radio map and also accounts for any changes in the environment. An analysis based on simulated models reduces the cost of developing a complex system by reducing the amount of hardware that has to be developed for performance evaluation. In some cases, this is the approach followed in this thesis, which is complemented with experiments on real data to validate the proposed methods.

This work considers the IEEE 802.11x channel model to develop a novel calibration algorithm based on Bayesian statistics as well as an indoor mobile tracking method. IEEE 802.11x channel model does not require an accurate floor plan of the indoor scenario [22].

In this work, a two-slope RSS model [6, 23] is investigated, which is a mathematical model of RSS that relates the path loss attenuation with distance. This means that the distance between the mobile node and the corresponding AP can be described by two models which depend on a breakpoint distance. It is observed in practice that for close distances (below breakpoint distance), the RSS measurements obey the first model and in opposite manner, for far distances (above breakpoint distance) the second model, resulting in an enhanced best performance in distance estimation when compared to using only the single-slope model.

This thesis proposes a solution to the generalized robust indoor localization problem, considering a two-slope RSS propagation model and the estimation of the path loss exponents and shadowing of the model.

1 Objectives

The general ojectives of this thesis are:

Determination/tracking of a mobile node path in different indoor environments should be intensively explored, tested and demonstrated. Analysis of existing integration architectures and methods for combined navigation utilizing inertial measurement units, as well as available positioning algorithms for using RSS measurements.

Development of novel algorithms to perform such integration based on the Bayesian filtering framework.

1.1 Specific Objectives

- ⋆ Mobile path tracking in an indoor environment using existing WLAN infrastructure where RSS measurements are available.
- ⋆ Investigation of joint online model calibration and mobile tracking methods.

- ⋆ Derivation of performance bounds when two-slope path loss model is considered.
- ⋆ Enhancement of the proposed methods by combining RSS measurements with external sensors (e.g. accelerometers or gyros), with the goal of obtaining a better tracking accuracy in terms of locating and tracking.
- ⋆ Addressing the positioning when the RSS measurements are taken from Access Points (APs) whose positions are not known.
- ⋆ Development, exploration, verification and demonstration of concepts developed within the thesis with real experiments.

2 Thesis Outline

This section gives a brief description of this Dissertation that could be of interest to readers in the areas of Signal Processing, Cognitive Radio, Wireless Networks, and Indoor Localization and Navigation. Methodology and novel contributions are presented along this thesis, which is organized in five chapters. In addition, the list of acronyms and symbols used in this work is presented at the beginning of this manuscript.

This thesis presents a survey of the technologies and algorithms used along this document for indoor positioning. This material is provided in **Chapter 1**. The state of the art presented in this chapter, gives a review of the most promising approaches and different contributions in the area of interest that can be found in the literature. This chapter is divided into three parts: The first part is referred to the different technologies used in indoor positioning defining the different measures which can be obtained using Wireless devices. The second part regards the different methodologies for positioning considering Range-based methods and Bayesian filtering being remarkable the use of the Extended Kalman Filter (EKF) that is the non-linear version of the Kalman Filter. The last part of this chapter is devoted to explain how inertial measurements are employed to obtain a better accuracy in terms of indoor positioning. Further, a brief description of the test-bed used to obtain real data to verify the computational simulations in this thesis is presented.

In **Chapter 2** a review of general concepts in statistical inference is given. Since this thesis is also focused in an online channel model calibration, different techniques in parameter estimation are presented in this part of the document. A brief explanation of the Maximum Likelihood Estimation (MLE) principle and the main differences with Bayesian techniques, mainly focused on Markov

Chain Monte Carlo (MCMC) method, is given. The benchmark to judge the accuracy of the estimator is the Cramér Rao Lower Bound (CRLB). Thus, its background theory is documented and referred in this chapter also

In **Chapter3** the channel model is presented. The focus is on a model having two path loss regions depending on the distance to the transmitter. The consideration of this channel model and not the classical one-slope model along this thesis is explained in this Chapter. In indoor positioning research, the importance of a channel model calibration is to obtain a good accuracy in terms of positioning. Different approaches have been studied in literature for channel model estimation where MCMC method gives a good performance, providing a moderate computational complexity and can be used in a tracking algorithm. This thesis considers the Gibbs sampler method as an approach for a channel model estimation just considering RSS measurements as observations. The benchmark against which can be compared the performance of the estimator is by means of its theoretical bound. The CRLB of the estimator of such model is also formulated in this chapter. Further, numerical results are presented in this work validated with real RSS measurements.

The online model calibration along with the mobile target tracking technical solution is presented in **Chapter 4**. Illustrative simulations and results with real data are discussed also. This Chapter presents the EKF-IMM algorithm developed in this Dissertation. The proposed methodology is based on a parallel Interacting Multiple Model architecture for distance estimation, which is coupled with an on-line calibration of the two-slope RSS channel model and the final position determination. The online channel model calibration is accomplished with MLE. The validation through computational simulations is complemented by the experimental tests using real data. The results with real data are compared when we use only RSS measurements and with the addition of INS measurements to the system.

The conclusions of this thesis and the proposed future research work is presented in 5.

1

State of the Art in Indoor Positioning

Target tracking is one of the most fundamental tasks in indoor positioning systems. Different indoor tracking algorithms have been introduced based on Bayesian filtering such as Kalman Filter and Particle filter. These algorithms are capable of probabilistically combine the device's motion dynamic which can be measured by embedded sensors readings (such as inertial measurement units) and the measurements obtained from wireless technologies deployed in a typical indoor area.

Contents

1.1 Background to Indoor Positioning

The success of outdoor positioning and applications based on the Global Navigation Satellite Systems (GNSS) provides an incentive to the research and development of positioning systems. But the need for localization is not just confined to persons or vehicles of transportation in outdoor environments where GNSS plays an important role. While accurately estimating location outdoors relying only on GNSS, the features of the system remains a difficult problem mainly because of signal blockage or severe attenuations. Because of the advent of Location-Based Service (LBS), there exists an actual need of robust and reliable indoor localization methodologies.

Indoor positioning techniques are needed to support, augment, and replace satellite-based navigation so that, for example, pedestrian positioning will be seamless from outdoors to indoors and of sufficient accuracy. The potential augmenting navigation technologies include inertial navigation methods and wireless network positioning approaches.

Compared to outdoor, indoor environments could be more complex and impose different challenges on location estimation. In indoor areas, there are various obstacles as walls, equipment, human beings influencing the propagation of electromagnetic waves which leads to a dense multipath fading[1] effect. For this reason, the need for fundamental studies of the characterization of indoor radio propagation and its impact on the accuracy of indoor positioning systems has been taken into consideration [23, 21]. Such analysis, reduces the cost of developing a complex system by reducing the amount of hardware that has to be developed for performance evaluation.

Due to the ubiquitous availability of powerful mobile computing devices, the bloom of location-aware applications has become an active field of research. A way of localization in indoor environments is using signals of opportunity such as Wireless Local Area Networks (WLAN), Zigbee, Ultra Wide Band (UWB), etc.

Optical, Magnetic, Infrared (IR), Radio Frequency Radio Frequency (RF) and ultra sound signals are the major wireless network positioning approaches used for indoor positioning systems [24]. To detect these signals, different types of sensors are required. Then, a sensing process is done to convert these signals

[1] Multipath fading occurs in a environment that results when the direct path to a receiver is blocked and the signal from the source is reflected at a scatter, causing a bias range estimate. Multipath fading may also cause distortion to the radio signal. As the various paths that can be taken by the signals vary in length, the signal transmitted at a particular instance will arrive at the receiver over a spread of times.

into useful quantities, such as distance or angle for later location determination and finally a Position Technique is needed to estimate the Mobile Target Position. This process is shown in Figure 1.1.

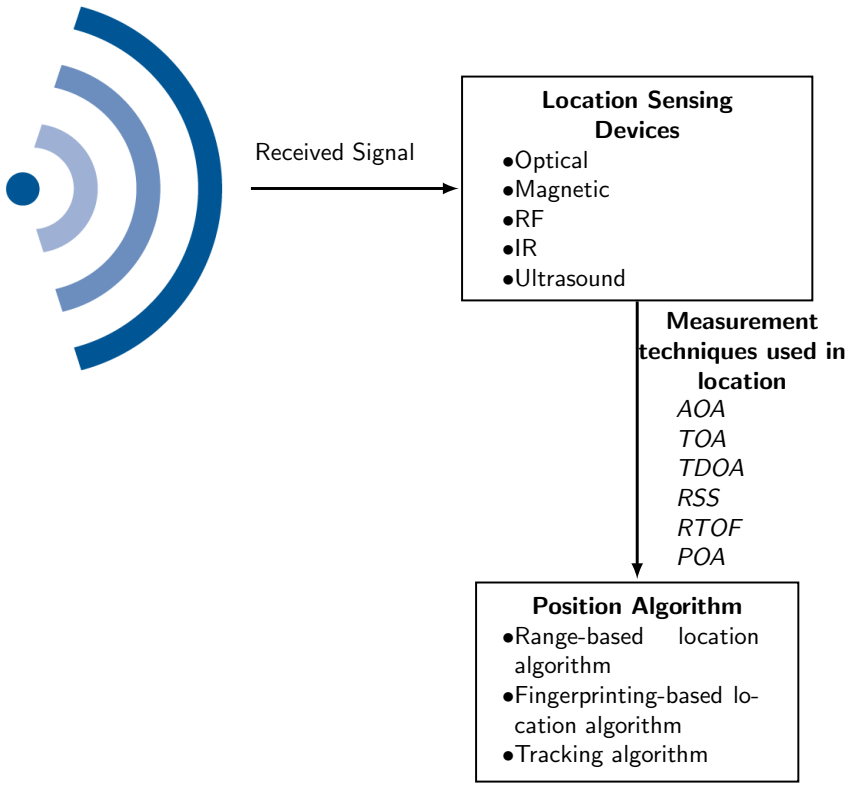


Figure 1.1: A functional block diagram of positioning system.

This thesis is focused in using WLAN signals. The popularity of WLANs opens a new opportunity for LBS. The advantage of working with WLAN signals as the primary source of information to approach the localization problem is the inexpensive hardware and the already dense deployment of WLAN Access Points(APs) in urban areas, therefore WLAN technology can be applied to provide indoor location service without deploying additional equipment.

1.2 Performance metrics

In the literature of position location, four areas of challenges are identified [25]: *Accuracy, precision, cost and complexity, applications requirements* and *security*. These metrics are briefly described in this section.

1.2.1 Accuracy

Accuracy (or positioning error) that is usually reported as an error range between the estimated location and the Mobile Target location. This performance metric also includes the confidence interval or percentage of successful location detection which is called the location precision [26]. Accuracy is the most important requirement of positioning systems. Usually, Root-Mean-Square Error (RMSE) is adopted as the performance metric:

$$RMSE_k = \sqrt{\mathbb{E}\{\mathrm{err}^2(\mathbf{p}_k)\}} \tag{1.1}$$

where $\mathrm{err}(\mathbf{p}_k)$ is the position estimation error at time instant k of average Euclidean distance between the estimated position $\hat{\mathbf{p}}_k$ and the true position $\mathbf{p}_k$ expressed as $\mathrm{err}(\mathbf{p}_k) = \|\hat{\mathbf{p}}_k - \mathbf{p}_k\|$.

An affordable method to compare and predict the performance of a positioning method with the theoretically best achievable, is the derivation of the CRLB. The CRLB gives a lower bound on the variance of the estimated position. The CRLB is widely mentioned along this thesis and its description is extended in Chapter 2.

1.2.2 Precision

Accuracy only considers the value of mean distance errors. However, positioning precision is a measure of the robustness of the positioning technique as it reveals the variation in its performance over many trials. Literature defines the location precision as the standard deviation in the location error or the Geometric Dilution of Precision (GDoP), but in this thesis, the distribution

of distance error between the estimated location and the true location is used [17, 27].

Usually, the Cumulative Probability Function (CDF) of the error is used for measuring the precision of a system. When two positioning techniques are compared, if their accuracies are similar, the CDF plot is preferred to select the one which reaches high probability values faster, with the error concentrated in small values. In practice, CDF is described by the percentile format. [17, 28]

All of these performance metrics depend on the choice of sensing technologies, characteristics of the radio channel and environment, the bandwidth of sensing signal, system's infrastructure capabilities, positioning algorithm, and complexity of signal processing techniques employed to estimate the location information.

1.2.3 Cost and Complexity

The cost incurred by a positioning system involves important factors as money, time, space, weight, energy, needs for extra infrastructure, additional bandwidth, fault tolerance and reliability, and nature of deployed technology. If a positioning system can reuse an existing communication infrastructure, some part of infrastructure, equipment, and bandwidth can be saved. Energy is an important cost factor of a system also. Some mobile units (e.g., electronic article surveillance (EAS) tags and passive Radio-Frequency Identification (RFID) tags) are completely energy passive. These units only respond to external fields and, thus, could have an unlimited lifetime. But other mobile units (e.g., devices with rechargeable battery) have a lifetime of several hours without recharging.

The complexity of the used signal processing algorithms to estimate the location is another issue that needs to be balanced with the performance of positioning systems. Complexity of a positioning system can be attributed to hardware, software, computational complexity and operation factors.

1.2.4 Application Requirements

The major application requirements for the location information are

1. The *granularity* that is subdivided into *temporal granularity* and *spatial granularity*. Temporal granularity determines the rate at which the location information is requested while spatial granularity determines the level of detail of location information.

2. The *performance* requirements that include any combination of performance metrics discussed above.

3. The *availability* of the location information that may be required at different entities within a wireless network leading to a two approaches for location systems: self-positioning (the mobile terminal determines its own location) and remote-positioning (an entity in the backbone infrastructure determines the location of a mobile terminal) [25]. Remote-positioning can be found in literature as localization.

1.2.5 Security

Location information should be made available only to those with authorized access. This issue represents the privacy concern of mobile users who do not want to reveal their location or be tracked. In the case of remote positioning systems, this is particularly significant. Here, a mobile terminal transmits signals with the intent to enable their capture and processing to derive position location information making it very difficult to secure such signals. Unfortunately, the security of the system is limited by the location sensing technique. For instance, a positioning system that reuses the communication signals for the purpose of location detection cannot completely secure the Mobile Target's privacy because of its active nature.

1.3 Techniques for Indoor Wireless Network positioning

Utilizing the RF signals that are readily available for communication purposes, can complement the data networking service with user positioning and tracking capabilities

There are various approaches to position (or locate) a Mobile Target. A basic functional block diagram of wireless positioning system is suggested by [26]. It consists of a number of location sensing devices, a positioning algorithm, and

finally the mobile target location. Figure 1.1 illustrates these components and their relationships.

Indoor positioning systems use location techniques to provide location information of persons and devices [29]. The main positioning algorithms used in the literature are classified in:

- Range-based location methods.
- Fingerprinting-based location technique.
- Tracking algorithm.

and they are described in the following subsections.

1.3.1 Range-based location methods.

These are sensing techniques that use the geometric properties of the measured ranges to compute the Mobile Target location. These techniques are also called *range measurement techniques* and they are divisible into two subcategories *lateration* and *angulation*.

Lateration technique, computes the position of an object by measuring its distance to multiple APs. The *Lateration* technique uses distance measurements and it is based on different types of measurements, such as Time-of-Arrival (ToA) [27], Roundtime Time-of-Flight (RToF) and Time Difference-of-Arrival (TDoA) [30]. An alternative is using the Received Signal Strength (RSS). Since most of mobile devices are equipped with wireless capacity, no additional hardware is required. There exist a logaritmic relation between distance and the power measurement received in the Mobile Target also [31, 16, 32, 15, 33, 34]. Hence, the distance between the Mobile Target and the AP is derived by computing the attenuation of the emitted signal strength or multiplying the radio signal velocity by travel time.

To compute a Mobile target's position in two dimensions, it is required the distance measurements from at least three non-collinear points. In three dimensions, distance measurements from four non-coplanar points are required [35].

The *Angulation* uses primarily angle or bearing measurements and it is based mainly on the glsAoA [36, 31, 37] of the signal. AOA methods utilize antenna arrays to estimate the direction of arrival of the signal of interest. Thus a single AOA measurement restricts the source location along a line in the estimated

AOA. Usually multiple AOA measurements are used to improve the estimation accuracy by using the redundant information.

The advantage of using AOA-based system as a location technique is that a position may be estimated with as a few as three measurements for 3-D positioning or two measurements for 2-D positioning and time synchronization is not required. On the other hand, the most important disadvantages of this technique is the hardware complexity that is required and the location estimate degradation that happens when the mobile target moves farther from the AP [36, 37].

1.3.1.1 Time Of Arrival (TOA)

For TOA-based systems, one-way signal propagation time between the Mobile Target and the AP is measured. This travel time is proportional to the distance between both devices. TOA-based systems affront basically two problems:

1. The transmitters and receivers in the system should be synchronized.
2. A timestamp should be labeled in the transmitting signal in order for the measuring unit to discern the distance that the signal has traveled.

The TOA measurement model is developed as follows. Let $\mathbf{p}_j$ the known position of the $j-th$ AP, $j = \{1, 2, .., r\}$ and $\mathbf{p}$ the position of the mobile target. The distance between the source and the sensor is $d_j = \|\mathbf{p}_j - \mathbf{p}\|$. A geometric representation is shown in Figure 1.2.

The Mobile Target transmit a signal at time 0 and the $j-th$ AP receives this signal at time t_j. The relationship between time t_j and d_j is given by:

$$t_j = \frac{d_j}{c_s}, \tag{1.2}$$

where c is the speed of light in $\frac{m}{s}$. So, the range measurement (r^j_{TOA}) can be modeled as:

$$r^j_{TOA} = d_j + \varepsilon^j_{TOA} = \|\mathbf{p}_j - \mathbf{p}\| + \varepsilon^j_{TOA}, \tag{1.3}$$

where ε^j_{TOA} is a random error modeled as $\mathcal{N}\left(0, \sigma^2_{TOA,j}\right)$.

To represent the range measurements of every AP, (1.3) can be represented in vector as:

$$\mathbf{r}_{TOA} = \mathbf{f}(\mathbf{p}) + \boldsymbol{\varepsilon}_{TOA}, \tag{1.4}$$

where

$$\mathbf{r}_{TOA} = \left[r^1_{TOA}, r^2_{TOA}, ..., r^r_{TOA}\right]^\top, \tag{1.5}$$

and

$$\boldsymbol{\varepsilon}_{TOA} = \left[\varepsilon^1_{TOA}, \varepsilon^2_{TOA}, ..., \varepsilon^r_{TOA}\right]^\top. \tag{1.6}$$

.

$\mathbf{f}(\mathbf{p})$ is a known function parameterized by $\mathbf{p}$ and represented as:

$$\mathbf{f}(\mathbf{p}) = \mathbf{D} = \begin{bmatrix} \|\mathbf{p}_1 - \mathbf{p}\| \\ \|\mathbf{p}_2 - \mathbf{p}\| \\ \vdots \\ \|\mathbf{p}_r - \mathbf{p}\| \end{bmatrix} \tag{1.7}$$

1.3.1.2 Round-Trip time-of-Flight RTOF

RTOF consists on measuring the time of flight of the signal from the transmitter to the receiver and back. RTOF like TOA is based on the intersection of circumferences which radius are the calculated distance from the fixed station to the mobile station. The main difference between TOA and RTOF is a more moderate clock synchronization requirement, in the latter, replacing the above synchronization requirement in TOA [38].

1.3.1.3 Time Difference of Arrival (TDOA)

TDOA techniques are based on estimating the difference in the arrival of times of the signal from the AP at multiple receivers and how this range difference corresponds to a hyperbolic function [38]. This is usually accomplished by taking a snapshot of the signal at a synchronized time period at multiple receivers. The cross-correlation of the two versions of the signal at pairs of AP is done and the peak of the crosscorrelation output gives the time difference for the signal arrival at those two base stations.

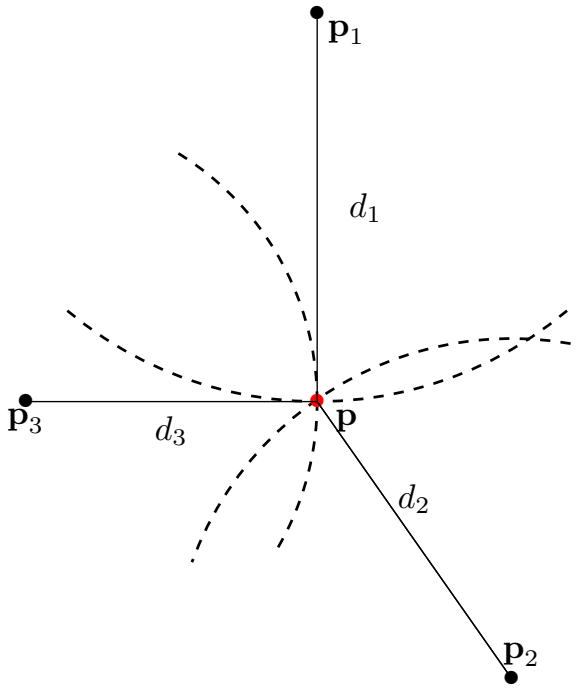


Figure 1.2: Geometric representation of the Positioning based in TOA measurements where $\mathbf{p}_j$ represents the AP positions and $\mathbf{p}$ the Mobile Target position to estimate.

The equation of the hyperbole is given by:

$$\Delta d_{i-j} = \|\mathbf{p}_i - \mathbf{p}\| - \|\mathbf{p}_j - \mathbf{p}\|, \tag{1.8}$$

where $\mathbf{p}_i$, $\mathbf{p}_j$ are the AP i and j fixed positions and $\mathbf{p}$ is the Mobile target position. The two intersections of two or more TDOA measurements, is shown in Figure 1.3.

Supposing that we have a transmitted signal $s(t)$ is corrupted by a zero mean Gaussian noise ε and delayed by τ. Then, the received signal for the $i-th$ AP is:

$$x_i(t) = s(t - \tau_i) + \varepsilon_i, \tag{1.9}$$

Similarly, for the $j-th$ AP, we have:

$$x_j(t) = s(t - \tau_j) + \varepsilon_j. \tag{1.10}$$

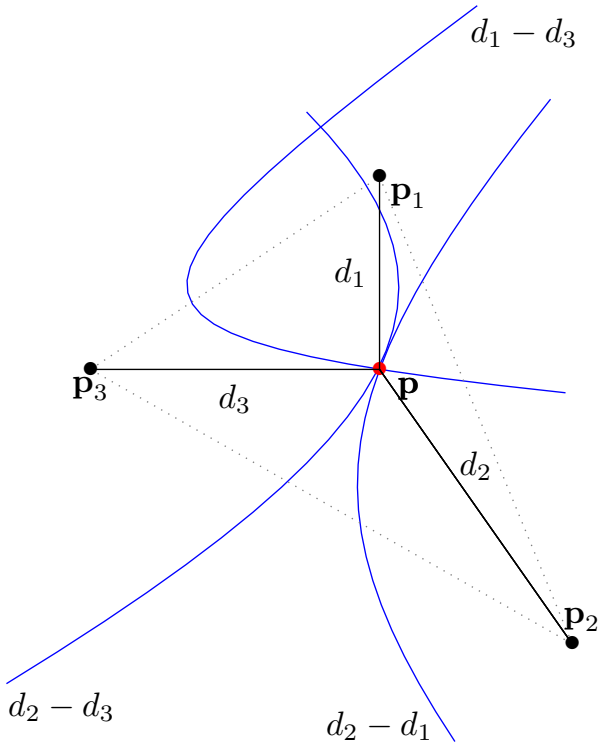


Figure 1.3: Geometric representation of the Positioning based in TDOA measurements. Two hyperbolas are formed from TDOA measurements at three fixed AP ($\mathbf{p}_1$, $\mathbf{p}_2$, and $\mathbf{p}_3$) to provide an intersection point, which is the Mobile Target $\mathbf{p}$ position.

A correlation analysis between (1.9) and (1.10) is used to provide a time delay $\tau_i - \tau_j$ corresponding to provide an estimate of Δd_{i-j} as follows:

$$\Delta d_{i-j} = c_s(\tau_i - \tau_j), \quad i \leq i < j \leq r. \tag{1.11}$$

Each Δd_{i-j} corresponds to a position along a hyperbola. Relating (1.8) with (1.11) we have a delay measurement-based TDOA as:

$$c_s(\tau_i - \tau_j) = \|\mathbf{p}_i - \mathbf{p}\| - \|\mathbf{p}_j - \mathbf{p}\|. \tag{1.12}$$

This approach requires a precise time reference and reference signals. [39] proposes a solution for IEEE802.11 WLAN which eliminates the requirement of initial synchronization in the conventional TDOA methods [38].

The path-loss effect is also utilized in order to calculate the mobile target's position. It is based on the fact that the received signal has been attempted inversely proportional tot the traveled distance. In indoor environments due to multipath effect, the path-loss model do not always hold so the parameters employed in this model the site-specific. In the literature, different theoretical and empirical models approaches have been used to translate the difference between the transmitted signal strength and the received signal strength into a range estimate (Figure 1.4).

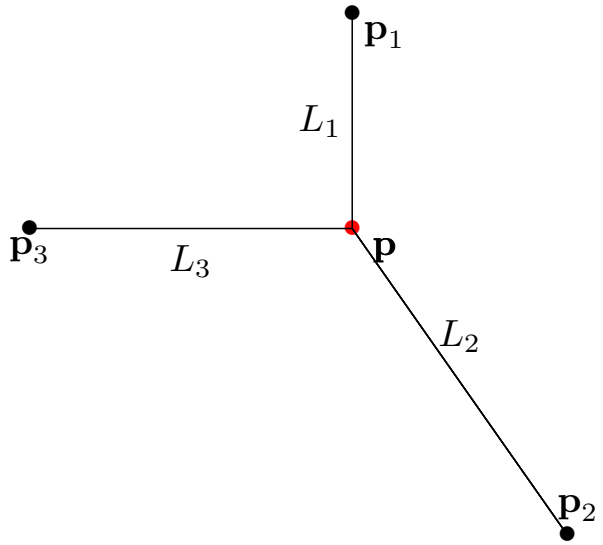


Figure 1.4: Geometric representation of the Positioning based in RSS measurements. L_1, L_2 and L_3 represent the measured path loss.

One of these approaches is the one-slope model that is widely used in RSS based indoor positioning [23, 40, 33]. A proper propagation model (or a propagation model calibrated to best fit the specific environment in the area of interest)[5] is used to translate RSS values to distances from the respective neighboring APs. A comparative between the one-slope model and the two-slope model is presented in [5].

Random shadowing effects occurring over a large number of measurement locations which have the same Transmitter-Receiver separation, but different levels of clutter on the propagation path isreferred to as Log-Normal Distribution.

The expression for free space path loss L_j comes from the Friis equation with path loss α [22]:

$$L_j = -10 \log_{10} \left[\frac{G_t G_r \lambda^2}{(4\pi d_j)^2} \cdot \frac{1}{d_j} \right] , \tag{1.13}$$

where G_t, G_r are the transmitter and receiver antenna gains, respectively, d_j is the distance between the jth transmitter and the receiver in meters, and λ is the wavelength of the transmitted carrier frequency. Assuming isotropic antennas, (1.13) becomes:

$$L_j = L_o + 10\alpha \log_{10}(d_j), \tag{1.14}$$

that corresponds to the expression for the one-slope model. L_o statistically describes the path loss model for an arbitrary location having a specific Transmiter-Receiver [41, 6], α is the path loss exponent and d_j is the distance between the jth AP and the Mobile Target given by:

$$d_j = \|\mathbf{p}_j - \mathbf{p}\|. \tag{1.15}$$

For any given transmitter/receiver configuration, the regions surroundings the AP and the Mobile Target can differ, resulting in the received signal with strength differing from the nominal (1.14). This variation is known as shadow fading or log-normal shadow fading and it is modeled as an additive zero-mean Gaussian random variable $\chi_{\sigma_j^2} \sim \mathcal{N}(0, \sigma_j^2)$. Then, including the shadow fading effect to (1.14), the path loss model results in:

$$L_j = L_o + 10\alpha \log_{10}(d_j) + \chi_{\sigma_j^2}, \tag{1.16}$$

The function L_j was obtained by measurement campaigns using radio signal ray tracing methods, premeasured RSS contours centered in the receiver or multiple measurements at several base stations [22, 31, 42].

A distance estimate can be obtained if we rearrange the terms in (1.16) and thus, we obtain:

$$\begin{aligned} \hat{d}_j &= 10^{\frac{L_o - L_j + \chi_{\sigma_j^2}}{10\alpha}} \\ &= 10^{\frac{L_o - L_j}{10\alpha}} \cdot 10^{\frac{\chi_{\sigma_j^2}}{10\alpha}}. \end{aligned} \tag{1.17}$$

Because $d_j = 10^{\frac{L_o - L_j}{10\alpha}}$, Equation 1.17 results in:

$$\begin{aligned} \hat{d}_j &= d_j \cdot 10^{\frac{\chi_{\sigma_j^2}}{10\alpha}} \\ &= d_j \cdot \xi_{\hat{d}_j}. \end{aligned} \tag{1.18}$$

It is remarkable that (1.18) depends on a log-normal random variable $\xi_{\hat{d}_j} \sim \mathcal{N}(\mu_{\xi_{\hat{d}_j}}, \sigma^2_{\xi_{\hat{d}_j}})$ where $\mu_{\xi_{\hat{d}_j}}, = 0$ and

$$\begin{aligned} \sigma^2_{\xi_{\hat{d}_j}} &= (e^{\sigma_\varepsilon^2} - 1)e^{2\mu_\varepsilon + \sigma_\varepsilon^2} \\ &= (e^{\sigma_\varepsilon^2} - 1)e^{\sigma_\varepsilon^2}. \end{aligned} \tag{1.19}$$

If $\xi_{\hat{d}_j}$ is a Log-Normal distribution, then ε is a normal random variables distributes as $\varepsilon = \log \xi_{\hat{d}_j} \sim \mathcal{N}(\mu_\varepsilon, \sigma^2_\varepsilon)$. where $\mu_\varepsilon = 0$ and $\sigma^2_\varepsilon = (\frac{\log 10 d_j \sigma_j^2}{10\alpha})^2$.

The variance of the distance estimate $\hat{d}_j$ may be expressed as:

$$\sigma^2_{d_j} = d_j^2(e^{\sigma_\varepsilon^2} - 1)e^{\sigma_\varepsilon^2}. \tag{1.20}$$

Indeed,the CRLB for a distance estimate from RSS measurements is [43]

$$\sigma^2_{d_j} \geq \left(\frac{\log 10 d_j}{10\alpha} \cdot \sigma_j\right)^2 \tag{1.21}$$

Because most of the network-based location estimations use RSS measurements, this thesis is centered in WLAN RSS-based positioning systems considering the two-slope model that it will be deeply discussed in the Chapter 3 .

1.3.2 Fingerprinting-based location technique

Fingerprinting refers to a technique that exploits the relationship between any measurable physical stimulus and a specific location. In the RADAR case [16], the RSS is the stimulus. This type of positioning technique is proposed to improve the accuracy of indoor position measurements by using premeasured location related data.

Fingerprinting based positioning systems usually work in two phases as shown in Figure 1.5: calibration phase (also called offline training phase) and positioning phase (also called online phase or run-time phase) [16, 44].

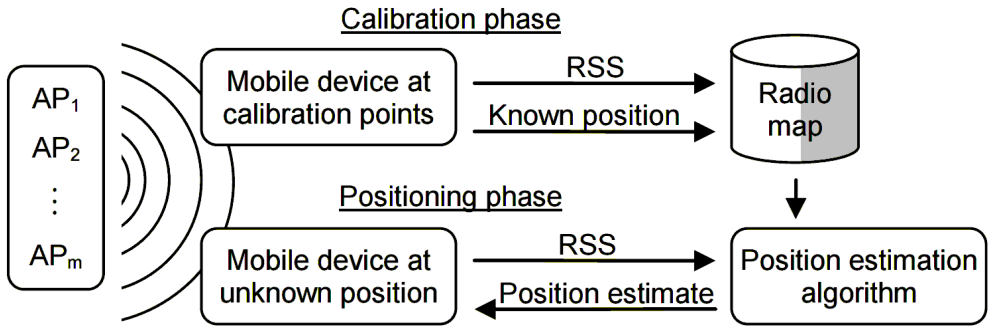


Figure 1.5: The two phases of location fingerprinting [1].

In the off-line phase, a mobile device is used to measure RSS values (in dBm) from APs at the chosen calibration points in the area of interest and this entire area is covered by a rectangular grid of points. The distance between two closest physical positions is called *grid spacing* and usually reported in meters or feet.

The RSS is measured with enough statistics to create a database or a table of RSS patterns (called a radio map) on the predetermined points of the grid. The vector of RSS values at a point on the grid is called the *location fingerprint* of that point.

During the on-line phase, the location related data of a target object is measured and compared with the the radio map collected in the offline phase to get a similar case in the database to make the location estimations. One of the most important algorithms used to estimate the location is the *k-nearest neighbor* algorithm [16, 45], which is based on estimating the position depending on the average (in physical space) of the coordinates of the k closest calibration points to the received RSS vector (radio map).

Other advanced algorithms such as neural networks [46] and Bayesian modeling [47] have been introduced for indoor positioning systems to determine the relationship between samples of RSS and the location fingerprint in the radio map.

1.3.3 Tracking Algorithm

Positioning is referred when the target's own positiong is self-located. Likewise, *tracking* is defined as the problem of estimating the trajectory of Mobile Target from the first to the most actual position in order to maintain an estimate of the Mobile target's current state using a set of successive measurements.

When tracking is done continuously in time, the dynamics of the Mobile Target and the measurements are most combined using an optimal filter or smoother. This filter is applied to calculate the mean and error covariance from converted measurements and they are based on linearized models and Gaussian noise approximations where the state vector contains the position and derivatives of the position,

Traditionally, target tracking is performed using the Kalman filter algorithm or its more sophisticated approaches such as the Extenden Kalman Filter (EKF), Unscented Kalman Filter (UKF) Cubature Kalman Filter (CKF), Quadrature Kalman Filter (QKF) or Particle Filter (PF) when the model is not linear[48]. These algorithms assume the target movement can be described by a dynamic system model where the target state (which typically consists of the position and velocity of the target) can be observed via a measurement model [49, 50].

The Kalman Filter (named after Rudolf E. Kálmán), is a recursive linear estimator which successively calculates an estimate for a continuous valued state, that evolves over time, on the bases of periodic observations of the state. It is over 50 years old but is still one of the most important and common data fusion algorithms in Navigation and Tracking area today. The Kalman Filter is derived exploiting the property that the product of two Gaussian distributions is another Gaussian distribution using vector algebra as a minimum mean squared estimator[51]. This filter is essentially a set of mathematical equations that implement a predictor-update type estimator that is optimal in the sense that it minimizes the estimated error covariance featuring a smoothing noisy data process and a parameter estimation [51].

The Kalman filter has a number of features which make it ideally suited to dealing with complex multi-sensor estimation and data fusion problems. Particularly, the explicit description of process and observations allows a wide variety of different sensor models to be incorporated within the basic algorithm. In addition, the consistent use of statistical measures of uncertainty makes it possible to quantitatively evaluate the role each sensor plays in overall system performance. Further, the linear recursive nature of the algorithm ensures that its application is simple and efficient. For these reason, the Kalman filter

has found wide-spread applications in many different data fusion problems [52, 53, 54, 55, 56, 57, 58].

In robotics, the Kalman filter is most suited to problems in tracking, localization and navigation; and less so to problems in mapping [59, 60]. This is because the algorithms works best with well defined state descriptions (position and velocities), and for states where observation and time-propagation models are also well understood

The Kalman filter may be considered as a specific instance of the recursive Bayesian state estimation for the case where the probability densities are linear-Gaussian [61, 62, 63]. However, reality often manifest itself as being very complex as nonlinear, non-Gaussian, non-stationary and with continuous-valued target states and in most practical situation, the Kalman Filter cannot be applied. Instead, ones is forced to use approximations or suboptimal solutions [61].

1.3.3.1 Conceptual solution to filtering

The Kalman Filter is derived for nonlinear systems with additive noise given by:

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}) + \boldsymbol{\nu}_{k-1} \tag{1.22}$$

$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k) + \boldsymbol{w}_k \tag{1.23}$$

The random sequences $\boldsymbol{\nu}_{k-1}$ and $\boldsymbol{w}_k$ are mutually independent, zero-mean white Gaussian with covariances $\mathbf{Q}_{k-1}$ and $\mathbf{R}_k$, respectively allowed to be time-variant.

The interest is the posterior density $p(\mathbf{x}_k|\mathbf{y}_k)$, where $\mathbf{Y}_k := \{\mathbf{y}_1, ..., \mathbf{y}_k\}$ is the set of measurements up to instant k and $\mathbf{x}_k$ is the state vector. The complexity of computing such a density grown exponentially with time; to make the computation tractable, the true state is being assumed an unobserved Markov process [62, 63].

Kalman Filter uses a prediction and an update step. The prediction step calculates the probability of the current state $\mathbf{x}_k$. Suppose that the required Probability Density Function (PDF) $p(\mathbf{x}_{k-1}|\mathbf{x}_{k-1})$ at instant $k-1$. The prediction state involves using the system model (1.22) to obtain the dynamic prior Probability Density Function (*pdf*) of the state at time k via the Chapman-Kolmogorov equation:

$$p(\mathbf{x}_k|\mathbf{Y}_{k-1}) = \int p(\mathbf{x}_k|\mathbf{x}_{k-1})p(\mathbf{x}_{k-1}|\mathbf{y}_{k-1})d\mathbf{x}_{k-1} \tag{1.24}$$

In Equation 1.24, use has been made of the fact that $p(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{Y}_{k-1}) = p(\mathbf{x}_k|\mathbf{x}_{k-1})$ as (1.22) describes a Markov process of order one. The probabilistic model of the state evolution $p(\mathbf{x}_k|\mathbf{x}_{k-1})$ is defined by the system equation 1.22 an the known statistics of ν_{k-1}, rendering the estimation recursive.

One that $p(\mathbf{x}_k|\mathbf{Y}_{k-1})$ is known, the update stage is carried out and the posterior or corrected estimate using a new measurement $\mathbf{y}_k$ is made via the Bayes' rule

$$\begin{aligned} p(\mathbf{x}_k|\mathbf{y}_k) &= p(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{y}_k), \mathbf{Y}_{k-1}) \\ &= \frac{p(\mathbf{y}_k|\mathbf{x}_k, \mathbf{Y}_{k-1})p(\mathbf{x}_k|\mathbf{Y}_{k-1})}{p(\mathbf{y}_k|\mathbf{Y}_{k-1})} \\ &= \frac{p(\mathbf{y}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{Y}_{k-1})}{p(\mathbf{y}_k|\mathbf{Y}_{k-1})} \end{aligned} \tag{1.25}$$

where the normalizing constant

$$p(\mathbf{x}_k|\mathbf{y}_{k-1}) = \int p(\mathbf{y}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{Y}_{k-1})d\mathbf{x}_k \tag{1.26}$$

depends on the likelihood function $p(\mathbf{y}_k|\mathbf{x}_k)$, defined by the measurement model (1.23) and the known statistics of $\boldsymbol{w}_k$. In the update stage (1.25), the measurement $\mathbf{y}_k$ is used to modify the prior density to obtain the required posterior density of the current state.

1.3.3.2 The Extended Kalman Filter

As mentioned in Section 1.3.3, in the category of suboptimals filters, the most used to deal with a non-linear model is the EKF. The main feature of the EKF is to provide a recursive state estimation of a dynamic system, or more precisely, linearize the nonlinear functions in the state dynamic and measurement models[62]. To obtain this process, the EKF calculates a Gaussian approximation to the true belief as shown in Figure 1.6. The EKF filter, derived using (1.22) and (1.23), makes the following assumptions about the respective *pdf*s:

$$p(\mathbf{x}_k|\mathbf{x}_{k-1}) \sim \mathcal{N}(\mathbf{f}(\mathbf{x}_{k-1}), \mathbf{Q}_k) \tag{1.27}$$

$$p(\mathbf{y}_k|\mathbf{x}_k) \sim \mathcal{N}(\mathbf{h}(\mathbf{x}_k), \mathbf{R}_k) \tag{1.28}$$

$$p(\mathbf{x}_{k-1}|\mathbf{y}_{k-1}) \sim \mathcal{N}(\mathbf{x}_{k-1}, \mathbf{P}_{k-1}) \tag{1.29}$$

where $\mathcal{N}(\mu, \mathbf{\Sigma})$ denotes the Gaussian distribution with mean μ and covariance matrix $\mathbf{\Sigma}$, $\mathbf{f}(\mathbf{x})$ the motion model and $\mathbf{h}(\mathbf{x})$ the measurement model. The mean and the covariance of the underlying Gaussian density are computed recursively. Detailed derivation may be found in textbooks on the subject [64, 61]. The main stages in the derivation of the EKF follow directly from those of the Kalman Filter with the additional step that the nonlinear functions (1.23) and (1.22) are approximated by the first term in their Taylor series expansion about the estimate and prediction, respectively. The algorithm has two stages:

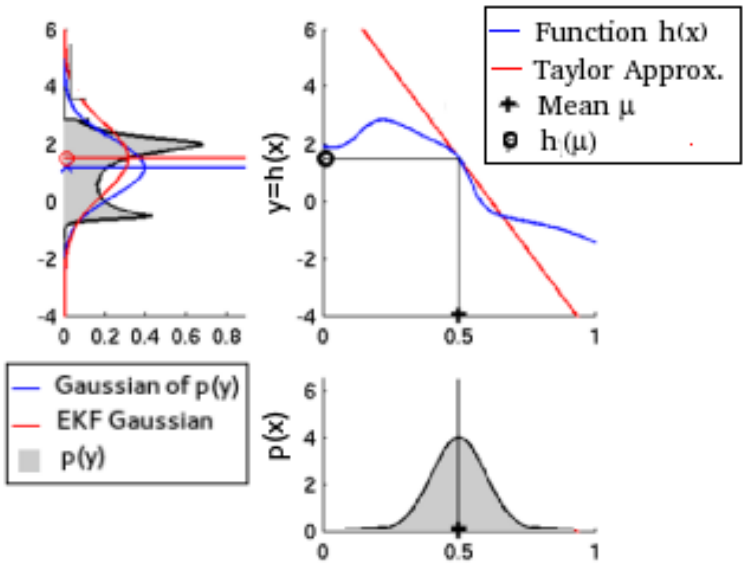


Figure 1.6: Illustration of EKF where the non-linear function $h(x)$ leads to a non Gaussian distribution $p(y)$. Thus, a linearization of $h(x)$ is needed to assure a Gaussian distribution [2].

Prediction. A prediction $\hat{\mathbf{x}}_{k|k-1}$ of the state at time k and its covariance $\mathbf{P}_{k|k-1}$ is computed according to

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{f}(\hat{\mathbf{x}}_{k-1|k-1}) \tag{1.30}$$

$$\mathbf{P}_{k|k-1} = \mathbf{F}\mathbf{P}_{k-1|k-1}\mathbf{F}^{\top} + \mathbf{Q}_k \tag{1.31}$$

Update. At time k an observation $\mathbf{y}_k$ is made and the updated estimate $\hat{\mathbf{x}}_{k|k}$ of the state $\mathbf{x}_k$, together with the updated estimate covariance $\mathbf{P}_{k|k}$ is computed from the state prediction and observation according to

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k\zeta_k \tag{1.32}$$

$$\hat{\mathbf{P}}_{k|k} = \hat{\mathbf{P}}_{k|k-1} - \mathbf{K}_k\mathbf{S}_k\mathbf{K}_k^{\top} \tag{1.33}$$

where the Kalman filter gain matrix $\mathbf{K}_k$ is given by

$$\mathbf{K}_k = \mathbf{P}_{k|k-1}\mathbf{H}_k^{\top}\mathbf{S}_k^{-1}, \tag{1.34}$$

the measurement residual (or innovations)

$$\zeta_k = \mathbf{y}_k - \mathbf{h}(\hat{\mathbf{x}}_{k|k-1}) \tag{1.35}$$

and the innovation covariance is,

$$\mathbf{S}_k = \mathbf{H}_k\mathbf{P}_{k|k-1}\mathbf{H}_k^{\top} + \mathbf{R}_k. \tag{1.36}$$

$\mathbf{F}$ (of dimension $n_x \times n_x$) and $\mathbf{H}_k$ (of dimension $n_z \times n_x$) are defined as the Jacobian matrices of $\mathbf{f}(\mathbf{x}_{k-1})$ and $\mathbf{h}(\mathbf{x}_k)$ evaluated in $\hat{\mathbf{x}}_{k-1|k-1}$ and $\hat{\mathbf{x}}_{k|k-1}$ respectively; that is

$$\mathbf{F}_k = \left[\nabla_{\mathbf{x}_{k-1}}\mathbf{f}^{\top}(\mathbf{x}_{k-1})\right]^{\top}\Big|_{\mathbf{x}_{k-1}=\hat{\mathbf{x}}_{k-1|k-1}} \tag{1.37}$$

$$\mathbf{H}_k = \left[\nabla_{\mathbf{x}_k} \mathbf{h}^\top(\mathbf{x}_k)\right]^\top \Big|_{\mathbf{x}_k = \hat{\mathbf{x}}_{k|k-1}} \tag{1.38}$$

where

$$\nabla_{\mathbf{x}_k} = \left[\frac{\delta}{\delta \mathbf{x}_k[1]} \quad \cdots \quad \frac{\delta}{\delta \mathbf{x}_k[n_x]}\right] \tag{1.39}$$

with $\mathbf{x}_k[i]$,$i = 1, ..., n_x$, bieng the ith component of vector $\mathbf{x}_k$. An element of $\mathbf{H}_k$ is given by:

$$\mathbf{H}_k[i,j] = \frac{\delta \mathbf{h}_k[i]}{\delta \mathbf{x}_k[j]} \Big|_{\mathbf{x}_k = \hat{\mathbf{x}}_{k|k-1}} \tag{1.40}$$

where $\mathbf{h}_k[i]$ denotes the ith component of vector $\mathbf{h}_k(\mathbf{x}_k)$.

The EKF works in much the same way as the Kalman filter with some notable details:

- If equations $\mathbf{h}_k$ and $\mathbf{f}_k$ are linear, it is not necessary (1.37) and (1.38) thus, these equations are equal and exact.
- The EKF assumes that $p(\mathbf{x}_k|\mathbf{y}_k)$ is Gaussian. If the nonlinearity in models (1.22) and (1.23) is moderately severe, the non-Gaussianity of the true posterior density will be more pronounced. In such cases the performance of the EKF will be degraded significantly.
- The Jacobians $\hat{\mathbf{F}}_k$ and $\hat{\mathbf{H}}_k$ are typically not constant, being functions of both state and timestep. This means that the covariances and gain matrix must be computed on-line as estimates and predictions are made available. This significantly increases the amount of computation which must be performed on-line by the algorithm.
- As the linearised model is derived by perturbing the true state and observation models around a predicted or nominal trajectory, great care must be taken to ensure that these predictions are always close enough to the true state that second order terms in the linearisation are indeed insignificant. If the nominal trajectory is too far away from the filter will perform poorly. In extreme cases the filter may also become unstable.
- The Kalman Filter (KF) and EKF employs a linearised model which must be compute from an approximate knowledge of the state. Unlike the linear algorithm, this means that the filter must be accurately initialized at the star of operation to ensure that the linearised models obtained are

valid. If this is not done, the estimates computed by the filter will simple be meaningless

1.3.3.3 Interacting Multiple Model

The implementation of IMM based methods allows the possibility of using highly dynamic models just when required, diminishing unrealistic noise considerations in non maneuvering situations and the computational charge of the system [64]. Several maneuvering targets tracking algorithms are developed [65]. Among them, the Interacting Multiple Model (IMM) method based on the optimal Kalman Filter, yields good performance when the measurement and state models are linear. However, if the these measurements are nonlinear, the EKF generally substitutes the optimal Kalman Filter. For this reason, the standard IMM algorithm will be reviewed.

The IMM filter consists of a bank of s nonlinear filters, each matched to a particular model. A convenient choice for a model-matched nonlinear filter is, for example, the EKF, or indeed any other nonlinear filter that adopts relationships (1.27)-(1.29).The output of model-matched filter j is the model-conditioned state estimate $\hat{\mathbf{x}}^j_{k|k}$ and its associated covariance $\mathbf{P}^j_{k|k}$.

The IMM estimator is designed to solve the problem defined by a hybrid system described by a dynamic and measurements equations in every instant k as:

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}, m_k, \boldsymbol{\upsilon}_k) \tag{1.41}$$

$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k, m_k, \boldsymbol{n}_k) \tag{1.42}$$

where m_k is the model variable in effect during the sampling period $(k-1, k]$ that is with the impact of the new model, m_{k-1}. The model variable m_k is commonly modeled by a an s-state first-order Markov chain with transitional probabilities [66]:

$$\pi_{j,i} \triangleq \mathbb{P}\left\{m_k = j | m_{k-1} = i\right\} \tag{1.43}$$

where $(j, i) \in S$ and $\mathcal{S} = \{1, 2, .., s\}$. The transitional probability matrix $[\mathbf{\Pi}]_{j,i} = \pi_{j,i}$ is thus an $s \times s$ matrix with elements satisfying

$$\pi_{j,i} \geq 0 \text{ and } \sum_{j=1}^{s} \pi_{j,i} = 1, \tag{1.44}$$

for each $j, i \in S$.

The total probability theorem is used as initial approximation of the IMM yielding s filters running in parallel:

$$\begin{aligned} p(\mathbf{x}_k|\mathbf{Y}_k) &= \sum_{i=1}^{s} p(\mathbf{x}_k|m_k = j, \mathbf{Y}_k)\mathbb{P}\left\{(r_k = j|\mathbf{Y}_k)\right\} \\ &= \sum_{i=1}^{s} p(\mathbf{x}_k|m_k = j, \mathbf{Y}_k)w_k^j, \end{aligned} \tag{1.45}$$

where w_k^j is the mixture weight (probability model) and it is obtained from the Bayes' formula:

$$\begin{aligned} w_k^j &\triangleq \mathbb{P}\{m_k = j|\mathbf{Y}_k\} \\ &= \mathbb{P}\{m_k = j|\mathbf{y}_k, \mathbf{Y}_{k-1}\} \\ &= \frac{p(\mathbf{y}_k|\mathbf{Y}_{k-1}, m_k = j)\mathbb{P}\{m_k = i|\mathbf{Y}_{k-1}\}}{p(\mathbf{y}_k|\mathbf{Y}_{k-1})} \\ &= \frac{p(\mathbf{y}_k|\mathbf{Y}_{k-1}, m_k = j)w_{k-1}^i}{\sum_{j=1}^{s} p(\mathbf{y}_k|\mathbf{Y}_{k-1}, i)w_{k-1}^j} \end{aligned} \tag{1.46}$$

where $LF_k^j = p(\mathbf{y}_k|\mathbf{Y}_{k-1}, m_k = j)$ is the model conditioned likelihood function. Under the Gaussian assumption, this likelihood function is Gaussian:

$$LF_k^j = p(\mathbf{y}_k|\mathbf{Y}_{k-1}) = \mathcal{N}(\zeta_k^j; \mathbf{0}, \mathbf{S}_k^j) \tag{1.47}$$

where ζ_k^j and $\mathbf{S}_k^j$ are the innovation and its covariance from the model-matched filter j. In this case:

$$\lim_{k\to\infty} w_k^i = \begin{cases} 0 \text{ if } m_k \neq j \\ 1 \text{ if } m_k = j \end{cases} . \tag{1.48}$$

A decomposition of the first term on the right hand side of Equation 1.45 the occurs via Bayes' rule

$$p(\mathbf{x}_k|m_k = j, \mathbf{Y}_{k-1}) = \frac{p(\mathbf{y}_k|m_k = j, \mathbf{x}_k)}{p(\mathbf{y}_k|m_k = j, \mathbf{Y}_{k-1})} p(\mathbf{x}_k|m_k = j, \mathbf{Y}_{k-1}). \tag{1.49}$$

Equation 1.49 reflectes one cycle of the state estimation filter matched to model $m_k = j$ starting with the prior, which is the last term above and applying the total probability theorem to this term, yields to [64]:

$$p(\mathbf{x}_k|m_k = j, \mathbf{Y}_{k-1}) \approx \sum_{i=1}^{s} p(\mathbf{x}_k|m_k = j, m_k = i, \hat{\mathbf{x}}_{k-1}^i, \mathbf{P}_{k-1}^i)\eta_{k-1}^{j|i}. \tag{1.50}$$

Thus, Equation 1.49 becomes,

$$p(\mathbf{x}_k|\mathbf{Y}_k) \approx \sum_{j=1}^{s} \frac{p(\mathbf{y}_k|m_k = j, \mathbf{x}_k)}{p(\mathbf{y}_k|r_k = j, \mathbf{Y}_{k-1})} \times \sum_{i=1}^{s} p(\mathbf{x}_k|m_k = j, m_k = i, \hat{\mathbf{x}}_{k-1}^i, \mathbf{P}_{k-1}^i)\eta_{k-1}^{j|i} \tag{1.51}$$

and the calculation of Equation 1.51 is broken into four steps composing the IMM algorithm [67, 68] .

The term $\eta_k^{j|i}$ is defined as:

$$\eta_{k-1}^{j|i} \triangleq \mathbb{P}\{r_{k-1} = i|r_k, \mathbf{Y}_{k-1}\} \tag{1.52}$$

These model transition probabilities is a homogeneous Markov chain. The system (1.41), (1.42) and (1.52) is a generalized version of a hidden Markov model [67]. Using Bayes' formula, it follows that:

$$\eta_{k-1}^{j|i} = \frac{\pi_{j,i} w_{k-1}^j}{\sum_{j=1}^{s} \pi_{j,i} w_{k-1}^j} \tag{1.53}$$

where $\pi_{j,i}$ is defined in Equation 1.44 and w_{k-1}^j are the mode probabilities as before.

$\eta_k^{j|i}$ in (1.50) is referred to the mixing probabilities (weights), different for each current model $m_k = j$. This mixture is assumed as a mixture of Gaussian

*pdf*s (Gaussian sum) and then approximated via moment matching by a single Gaussian. Thus, before the model-matched filtering step, the IMM performs a Gaussian mixture of models. The mixed posterior density at $k-1$ for model i is represented by [64]

$$\mathcal{N} \sim \left(\mathbf{x}^i_{k-1}; \hat{\mathbf{x}}^i_{k-1|k-1}, \mathbf{P}^i_{k-1|k-1}\right), \tag{1.54}$$

where

$$\hat{\mathbf{x}}^i_{k-1|k-1} = \sum_{i=1}^{s} \eta^{j|i}_{k-1} \hat{\mathbf{x}}^i_{k-1|k-1} \tag{1.55}$$

$$\mathbf{P}^i_{k-1|k-1} = \sum_{j=1}^{s} \eta^{j|i}_{k-1} \left[\mathbf{P}^j_{k-1|k-1} + \left(\hat{\mathbf{x}}^j_{k-1|k-1} - \hat{\mathbf{x}}^j_{k-1|k-1}\right)\right. \tag{1.56}$$

$$\left.\left(\hat{\mathbf{x}}^j_{k-1|k-1} - \hat{\mathbf{x}}^j_{k-1|k-1}\right)^\top\right]. \tag{1.57}$$

After the model-matched filtering, the posterior density of model i at instant k is represented by

$$\mathcal{N} \sim \left(\mathbf{x}^i; \hat{\mathbf{x}}^i_{k|k} \mathbf{P}^i_{k|k}\right) \tag{1.58}$$

and using again the Bayes rule, the model probabilities are updated as:

$$w^i_k = \frac{LF^i_k \sum_{j=1}^{s} \pi_{j,i} w^j_{k-1}}{\sum_{i^1}^{s} LF^i_k} \tag{1.59}$$

where $LF^i_k = p(\mathbf{y}_k|\mathbf{Y}_{k-1}, m_k = i)$ is the model likelihood function of (1.47). A general description of the IMM algorithm is depicted in Algorithm 1.

Algorithm 1 Step k of the IMM Algorithm

1: For where $(j, i) \in S$ and $S = \{1, 2, .., s\}$

2: **Reinitialization**:

Calculation of the predicted mode probability, mixing weights, mixing estimates and mixing covariances, respectively,

Predicted mode probability:
$$\eta^{(i)}_{k|k-1} = \sum_j \pi_{ji} \eta^{(j)}_{k-1} \tag{1.60}$$

Mixing weight:
$$\eta^{j|i}_{k-1} = \frac{\pi_{ji} \eta^{(j)}_{k-1}}{\eta^{(i)}_{k|k-1}} \tag{1.61}$$

Mixing Estimate:
$$\bar{\mathbf{x}}^{(i)}_{k-1|k-1} = \sum_j \hat{\mathbf{x}}^{(j)}_{k-1|k-1} \eta^{j|i}_{k-1} \tag{1.62}$$

Mixing Covariance:
$$\begin{aligned} \bar{\mathbf{P}}^{(i)}_{k-1|k-1} = \sum_j [&\mathbf{P}^{(j)}_{k-1|k-1} + \\ &+ (\bar{\mathbf{x}}^{(i)}_{k-1|k-1} - \hat{\mathbf{x}}^{(j)}_{k-1|k-1}) \\ &\times (\bar{\mathbf{x}}^{(i)}_{k-1|k-1} - \hat{\mathbf{x}}^{(j)}_{k-1|k-1})^\top] \eta^{j|i}_{k-1} \end{aligned} \tag{1.63}$$

3: **Model-conditioned EKF**:

Prediction, innovations' covariance matrix, Kalman gain, state estimate and the corresponding error covariance matrix, are given by

Predicted State:
$$\hat{\mathbf{x}}^{(i)}_{k|k-1} = \mathbf{f}(\mathbf{x}^{(i)}_{k-1|k-1}) \tag{1.64}$$

Predicted Covariance:
$$\mathbf{P}^{(i)}_{k|k-1} = \mathbf{F} \bar{\mathbf{P}}^{(i)}_{k-1|k-1} \mathbf{F}^\top + \mathbf{Q}^{(i)}_k \tag{1.65}$$

Residual Covariance:
$$\mathbf{S}^{(i),r}_k = \mathbf{H}^{(i)}_k \mathbf{P}^{(i)}_{k|k-1} (\mathbf{H}^{(i)}_k)^\top + \mathbf{R}^{(i)}_k \tag{1.66}$$

Filter Gain:
$$\mathbf{K}^{(i)}_k = \mathbf{P}^{(i)}_{k|k-1} (\mathbf{H}^{(i)}_k)^\top (\mathbf{S}^{(i)}_k)^{-1} \tag{1.67}$$

Updated State:
$$\hat{\mathbf{x}}^{(i)}_{k|k} = \hat{\mathbf{x}}^{(i)}_{k|k-1} + \mathbf{K}^{(i)}_k (y_k - \mathbf{h}(\hat{\mathbf{x}}^{(i)}_{k|k-1})) \tag{1.68}$$

$$\mathbf{P}^{(i)}_{k|k} = \mathbf{P}^{(i)}_{k|k-1} - \mathbf{K}^{(i)}_k \mathbf{S}^{(i)}_k (\mathbf{K}^{(i)}_k)^\top \tag{1.69}$$

Algorithm 2 Step k of the IMM Algorithm (continued...)

4: **Model probability update**:

The model likelihood function and model probability are respectively

$$\text{Model Likelihood:} \qquad LF_k^{(i)} = \mathcal{N}(y_k - \mathbf{H}_k^{(i)}\hat{\mathbf{x}}_{k|k-1}^{(i)}; 0, \mathbf{S}_k^{(i)}) \tag{1.70}$$

$$\text{Mode probability:} \qquad \eta_k^{(i)} = \frac{\eta_{k|k-1}^{(i)} LF_k^{(i)}}{\sum_j \eta_{k|k-1}^{(j)} LF_k^{(j)}} \tag{1.71}$$

5: **Estimate fusion**:

$$\hat{\mathbf{x}}_{k|k} = \sum_i \hat{\mathbf{x}}_{k|k}^{(i)} \eta_k^{(i)} \tag{1.72}$$

$$\mathbf{P}_{k|k} = \sum_i [\mathbf{P}_{k|k}^{(i)} + (\hat{\mathbf{x}}_{k|k} - \hat{\mathbf{x}}_{k|k}^{(i)})(\hat{\mathbf{x}}_{k|k} - \hat{\mathbf{x}}_{k|k}^{(i)})^\top] \eta_k^{(i)} \tag{1.73}$$

1.4 Inertial Navigation System

Inertial Navigation System (INS) is an electronic device which provides estimates of position, velocity and orientation from an Inertial Measurement Unit (IMU). An INS is a navigation technique in which three orthogonally arranged accelerometers (motion sensors), three gyroscopes (angular rate sensors) and/or a magnetometer (three perpendicular sensors for measuring the strength and/or direction of a magnetic field) supply measurements to track the orientation and position of a mobile target relative to a known initial position, orientation, velocity and acceleration. Recent advances in the construction of Micro Electronimechanical Systems (MEMS) have made it possible to manufacture small and light inertial navigation systems.

The IMU provides the motion information of the object with a high update rate as high as 300 Hz and it can achieve high precision in short time duration. However, without external adding, the system suffers from local anomalies and the error drifts quickly grow with time. The combination of INS and WLAN has been proven to be a reliable solution for indoor navigation [69, 70].

The main argument to use INS for navigation arises from independent operability without external infrastructure, making navigation possible in environments,

where the installation and maintenance of such infrastructure is not affordable.

The INS algorithms uses the acceloremeter and gyroscope readings in the sensor *body* (b) frame of reference ($\mathbf{a}^b_{ib,k}$ and $\boldsymbol{\omega}^b_{ib,k}$ respectively) which are taken at every sample interval Δt at every instant k and usually come from different coordinate systems. Therefore, it is necessary to transform the measured quantities into a common coordinate frame suitable in order to data processing.

A coordinate *arbitrary* frame a-frame in literature is usually denoted with the superscript a and they are briefly described here[71, 72].

Earth- Centered Earth Fixed Frame (ECEF *e*-frame): This frame has its origin at the center of mass of the Earth and rotates with it. The x-axis points to the reference meridian, the z-axis is parallel to the mean spin axis of the Earth and y-axis completes the right-handed orthogonal frame.

Inertial Frame (*i*-frame): In this coordinate frame Newtown's laws of motion are applied and it is not accelerating. It could be chose arbitrary but it is convenient to let its origin coincide with the e-frame.

Body Frame (*b*-frame): This coordinate system has its origin in the center of mass of the Mobile Target and usually the IMU coordinate is aligned with the vehicle *body* b-frame coordinate. The sensor x-axis often points to the forward direction, the y-axis points to the lateral direction, and the z-axis points to the vertical down direction forming a right-handed orthogonal coordinate. Figure 1.7 shows this coordinate frame. This frame is denoted by the superscript b.

1.4.1 Navigation frame mechanization

For low-cost MEMS-based IMU and for short distance navigation applications, simplifications can be made to the *navigation* n-frame mechanization model. This case is when the gyroscope bias error is significantly in excess of the rotation rate of the Earth, and the accelerometer bias error is much larger than the centripetal forces introduced by the earth rotation. Besides, for short distance navigation applications, the effects from the earth rotaton cannot be observed and the gravity (g) is assumed to be a constant [73].

The accelerations are resolved in the instrumental frame of the accelerometers and need to be transformed into platform coordinate *body* b-frame by a fixed rotation matrix. The same applies to the angular rates from the Gyroscope. Gyroscope measures the platform angular rates relative to the *inertial* i-frame of reference and resolves this in the instrumental frame of the gyros. From

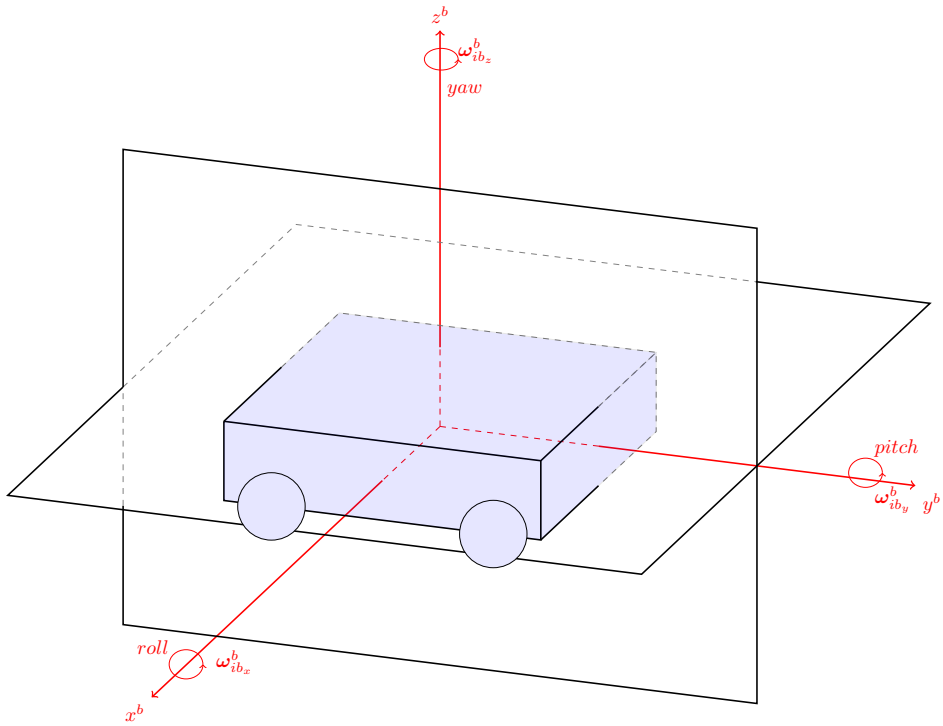


Figure 1.7: Body coordinate frame.

the gyro measurements a rotation matrix is calculated which transform the accelerations in the platform coordinate frame. For the data fusion with other systems, (for example GNSS and/or RSS measurements), the coordinates are transformed into the *navigation* n-frame.

One transformation method from the *body* (b) to the *navigation* (n) frame is based on the Direct Cosine Matrix (DCM) $\mathbf{C}^n_{b,k}$ [71].

The rotation rate of *body* b-frame with respect to *navigation* n-frame can be related with gyroscope angular rate raw measurements as [71]:

$$\boldsymbol{\omega}^n_{nb,k} = \boldsymbol{\omega}^b_{ib,k} - \boldsymbol{\omega}^b_{in,k} \tag{1.74}$$

$$= \boldsymbol{\omega}^b_{ib,k} - \mathbf{C}^n_{b,k}\left(\boldsymbol{\omega}^n_{ie,k} - \boldsymbol{\omega}^n_{en,k}\right) \tag{1.75}$$

where $\boldsymbol{\omega}^b_{ib,k}$ is the angular rate measured by the body-mounted gyroscopes; $\boldsymbol{\omega}^b_{in,k}$ represents the summing rotation rates of the Earth with respect to the inertial i-frame plus the turn rate of navigation n-frame with respect to the Earth.

Ignoring the Earth rotation rate and transport rate (i.e., $\left(\boldsymbol{\omega}^n_{ie,k} - \boldsymbol{\omega}^n_{en,k}\right) = \mathbf{0}$), the relationship between the derivative of Euler angles (attitude) $\dot{\boldsymbol{\Psi}}_k = \begin{bmatrix}\dot{\theta}_k & \dot{\phi}_k & \dot{\varphi}_k\end{bmatrix}^\top$ and the and gyroscope angular rate measurements is given by [71]:

$$\begin{aligned}\dot{\boldsymbol{\Psi}}_k &= \boldsymbol{\Phi}^n_{b,k} \cdot \boldsymbol{\omega}^n_{nb,k} && (1.76)\\ &= \boldsymbol{\Phi}^n_{b,k}\left[\boldsymbol{\omega}^b_{ib,k} - \mathbf{C}^n_{b,k}\left(\boldsymbol{\omega}^n_{ie,k} - \boldsymbol{\omega}^n_{en,k}\right)\right] && (1.77)\\ &= \boldsymbol{\Phi}^n_{b,k} \cdot \boldsymbol{\omega}^b_{ib,k}, && (1.78)\end{aligned}$$

with

$$\boldsymbol{\Phi}^n_{b,k} = \begin{bmatrix}1 & s\theta_k t\phi_k & c\theta_k t\phi_k\\ 0 & c\theta_k & -s\theta_k\\ 0 & \frac{s\theta_k}{c\phi_k} & \frac{c\theta_k}{c\phi_k}\end{bmatrix} \tag{1.79}$$

where $c(\,\cdot\,)$ and $s(\,\cdot\,)$ denote the cosine and sine operator respectively; θ_k, ϕ_k and φ_k represents the roll, pitch and yaw respectively.

Having the derivative of Euler angles and initial attitude information, the attitude estimates (e.g., Euler angles) can be calculated using integral operations. The body b-frame to navigation n-frame DCM can be formulated as shown in Equation 1.80.

The general DCM $\mathbf{C}^n_{b,k}$ as typically three degrees of freedom described with the three Euler angles. Applied to the transformation from the *body* to the *navigation* n-frame, $\mathbf{C}^n_{b,k}$ leads to:

$$\mathbf{C}^n_{b,k} = \begin{bmatrix}c\varphi_k c\theta_k & c\varphi_k s\phi_k s\theta_k - s\varphi_k c\theta_k & c\varphi_k s\phi_k c\theta_k - s\varphi_k s\theta_k\\ s\varphi_k c\theta_k & s\varphi_k s\phi_k s\theta_k + c\varphi_k c\theta_k & s\eta s\phi_k c\theta_k - c\varphi_k s\theta_k\\ -s\phi_k & c\phi_k s\theta_k & c\phi_k c\theta_k\end{bmatrix}. \tag{1.80}$$

Because the DCM is orthonormal, it is important to note the following properties of $\mathbf{C}^n_{b,k}$:

$$\mathbf{C}^n_{b,k} \cdot \mathbf{C}^{n\top}_{b,k} = \mathbf{C}^{n\top}_{b,k} \cdot \mathbf{C}^n_{b,k} = \mathbf{I} \text{ and } \mathbf{C}^n_{b,k} = \left(\mathbf{C}^n_{b,k}\right)^{-1} \tag{1.81}$$

The source error in the data recordings is due to the bias that accelerometer and gyroscope measurements have in smaller or large degree depends on the quality of the sensors, a simple, but effective, model of this effect is [74, 49]:

$$\mathbf{a}^{b}_{ib,k} = \breve{\mathbf{a}}^{b}_{ib,k} - \mathbf{a}^{b,error}_{ib,k} \tag{1.82}$$

$$\boldsymbol{\omega}^{b}_{ib,k} = \breve{\boldsymbol{\omega}}^{b}_{ib,k} + \boldsymbol{\omega}^{b,error}_{ib,k} \tag{1.83}$$

where

$\breve{\mathbf{a}}^{b}_{ib,k}$ is the bias-free true acceleration.

$\breve{\boldsymbol{\omega}}^{b}_{ib,k}$ is the bias-free true angular rate.

$\mathbf{a}^{b,error}_{ib,k}$ is the accelerometer error and

$\boldsymbol{\omega}^{b,error}_{ib,k}$ is the gyroscope error,

where $\mathbf{a}^{b,error}_{ib,k}$ and $\boldsymbol{\omega}^{b,error}_{ib,k}$ are assumed to be normally distributed with a zero mean and covariance $\boldsymbol{\sigma}^2_{acc}$ and $\boldsymbol{\sigma}^2_{gyro}$ [75]:

$$\mathbf{a}^{b,error}_{ib,k} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\sigma}^2_{acc}) \tag{1.84}$$

$$\boldsymbol{\omega}^{b,error}_{ib,k} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\sigma}^2_{gyro}) \tag{1.85}$$

From (1.75) to (1.83), the simplified mechanization model can be expressed as (1.86) and it will be applied as the system propagation model in the integration system.

$$\begin{aligned}
\mathbf{r}_k &= \mathbf{r}_{k-1} + \mathbf{v}_k \cdot \Delta t \\
\mathbf{v}_k &= \mathbf{v}_{k-1} + \left[\mathbf{C}^{n}_{b,k-1} \cdot \left(\breve{\mathbf{a}}^{b}_{ib,k} + \mathbf{a}^{b,error}_{ib,k} + \mathbf{g}\right)\right] \cdot \Delta t \\
\boldsymbol{\Psi}_k &= \boldsymbol{\Psi}_{k-1} + \boldsymbol{\Phi}^{n}_{b,k} \cdot \left(\breve{\boldsymbol{\omega}}^{b}_{ib,k} + \boldsymbol{\omega}^{b,error}_{ib,k}\right) \cdot \Delta t \\
\mathbf{a}^{b,error}_{ib,k} &= \mathbf{a}^{b,error}_{ib,k-1} + \mathbf{w}_{\mathbf{a},k-1} \\
\boldsymbol{\omega}^{b,error}_{ib,k} &= \boldsymbol{\omega}^{b,error}_{ib,k} + \mathbf{w}_{\boldsymbol{\omega},k-1}
\end{aligned} \tag{1.86}$$

In Equation 1.86, $\mathbf{r}_k$, $\mathbf{v}_k$, $\boldsymbol{\Psi}_k$ are position, velocity and attitude vectors of the object in *navigation* n-frame at every k instant, respectively.

The acceleration measurement $\mathbf{a}^{b}_{ib,k}$ obtained from the accelerometer, in its coordinate b-frame, is projected into the *navigation* n-frame using DCM $\mathbf{C}^{n,k}_{b}$. Then,

the acceleration due to gravity is subtracted and the remaining acceleration is:

$$\mathbf{a}_k = \mathbf{C}_{b,k}^{n}\mathbf{a}_{ib,k}^{b} - \begin{bmatrix} 0 & 0 & g \end{bmatrix}^{\top} \tag{1.87}$$

which is the quantity of interest in this thesis, $\begin{bmatrix} \ddot{x}_k & \ddot{y}_k & \ddot{z}_x \end{bmatrix}^{\top}$.

The main blocks of a INS is shown in Figure 1.8. A more detailed diagram of how the acceleration signal is integrated into an algorithm navigation system is shown in Figure 1.9.

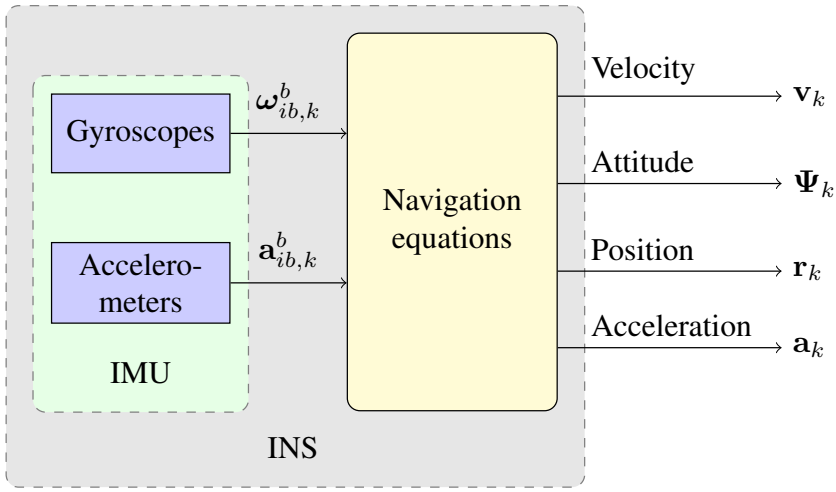


Figure 1.8: Inertial Navigation System diagram.

1.5 Test-Bed Development: Review of Key Technologies

This section provides a general description of the board used as a test device developed for [3]. The overall system consists of the ranging/positioning payload and the database. The ranging/positioning payload is a development board with multiple connections where ranging and positioning algorithms can be easily implemented.

The mobile path is a test-bed used to collect the RSS measurements that includes a *Raspberry Pi* board. The Raspberry Pi Foundation board was

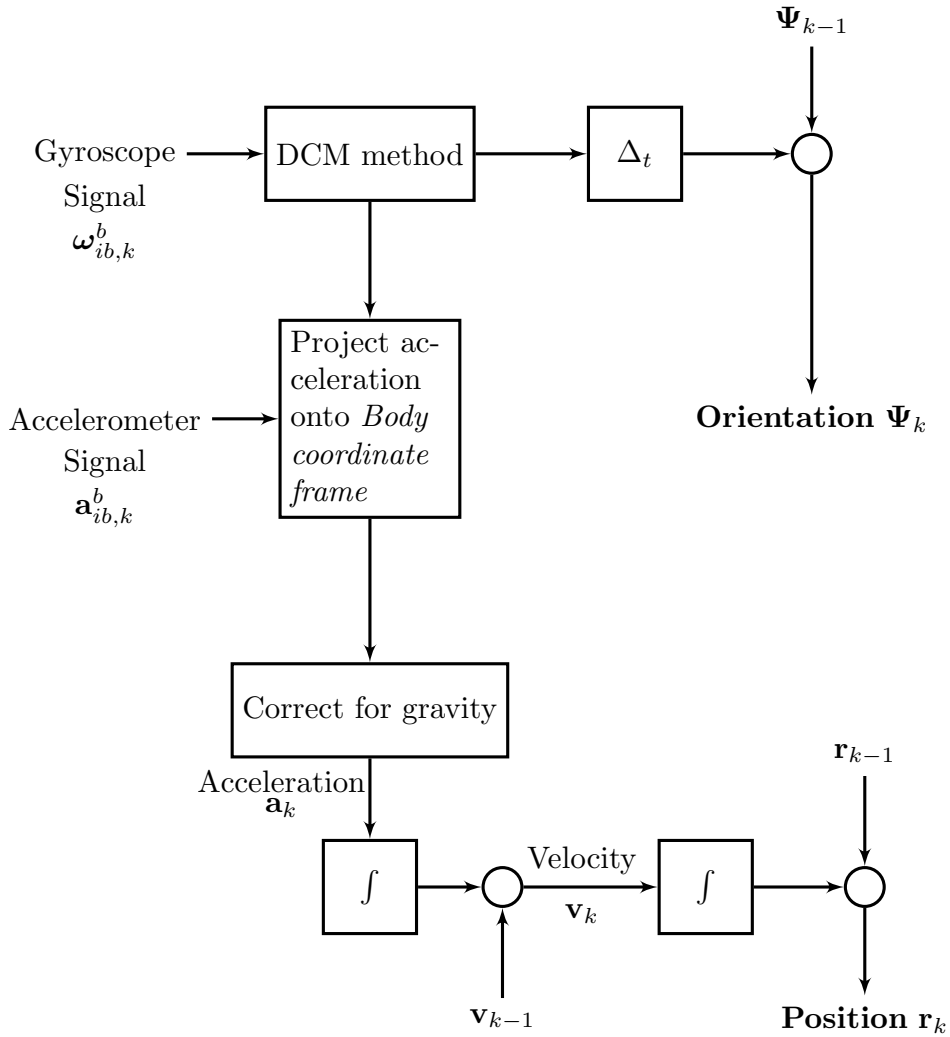


Figure 1.9: Strapdown inertial navigation algorithm.

developed by *Raspberry Pi Foundation*[76] [2].

[2]Raspberry Pi Foundation is a charity founded in 2009 to promote the study of basic computer science in schools, and is responsible for developing a single-board computer called the *Raspberry Pi*.

1.5.0.1 Platform Description

The core of the Test-bed is a *Broadcom BCM2835* system on a chip (SoC), featuring single core 700 MHz ARM1176JZF-S processor and 512 MB of main memory. Older versions were equipped with only 128 or 256 MB of main memory, so we have picked the latest "B" model (with 512 MB) for this project. Officially supported operating system is Raspbian but other alternative systems supported by the community or third part ies are available as well. Since Raspbian is a modified version of Debian GNU/Linux, it benefits from its broad software availability and cross-platform support. Vast majority of Debian packages can be installed and run on Raspbian without any modification in their source code, as ARM architecture is (among others) officially supported by Debian.

There is no hard disk or flash memory built-in on the Pi. The only persistent storage is an external SDHC card, partitioned and used for both firmware and the operating system with custom software and data. External devices can be connected via standard High Speed USB (2.0) interface. There is also wired one Fast Ethernet interface, internally connected to the USB bus. Video output is available through HDMI or RCA/composite ports.

1.5.0.2 Powering

A dedicated micro USB port with no data pins is used for powering the device. Power supply with standard voltage (5V) capable of providing at least 700mA on the output is recommended, but the total power consumption tightly depends on connected peripherals as well. It is generally safer to use a powered USB hub when connecting devices with the total power consumption over 150mA. Because of sensitive circuitry in RPi, susceptible to fluctuations in the input voltage, it is essential to use high-quality power supply for RPi itself and USB hub that does not "backfeed" the power through the upstream port [76]. Using inappropriate accessories can result in malfunction or brick the device entirely.

The table 1.1 lists some of the most important technical details of the Raspberry board.

Chipset	Broadcom BCM2835
CPU	700 MHz ARM
Coprocessors	Digital Processing, Floating Point Unit
GPU	Dual Core Video Core IV® Multimedia Co-Processor
Memory	512 MB SDRAM
Storage	SecureDigital/MMC/SDIO slot
USB	2
Ethernet	1 RJ-45 connector. Ethernet provided through USB
Audio Outputs	1 composite, 1 HDMI
Video Outputs	3.5 mm stereo jack, HDMI
Power Supply	max. 3.5 W
Power Consumption	8.6 cm. × 5.4 cm. × 1.7 cm.
Price	∼ 40€

Table 1.1: Main technical features of Raspberry Pi model B.

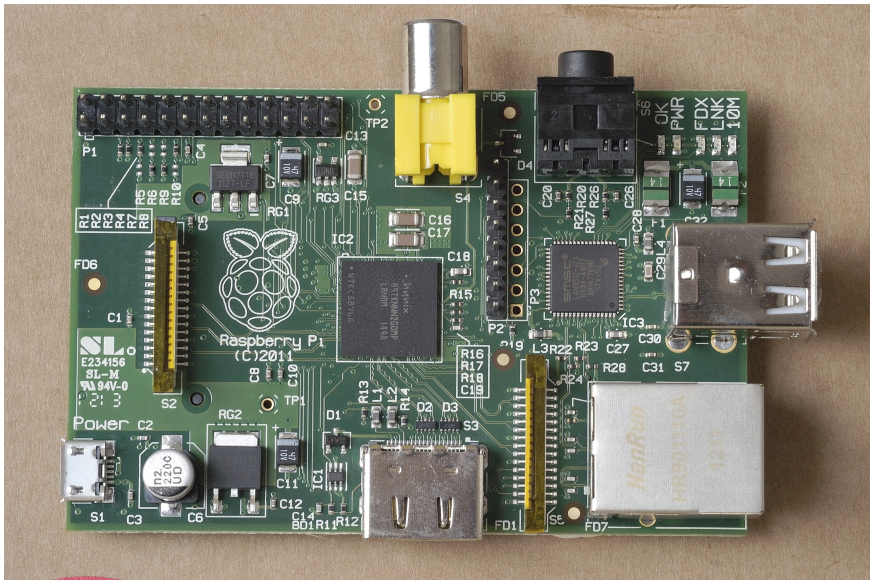


Figure 1.10: Raspberry Pi model B.

1.5.0.3 Accessing WLAN (IEEE 802.11)

This thesis is focused on the signals of IEEE 802.11 wireless networks as the primary source of information. The WiFi adapter chosen to access to these signals is the TL-WN722N WiFi card (from TP-LINK manufacturer).

Standards	IEEE 802.11g, IEEE 802.11b, IEEE 802.11n
Vendor	TP-Link
Chipset	Atheros AR9271
Interface	USB 2.0 Interface
Wireless Signal Rates	Up to 150 Mbps
Frequency Range	2.4-2.4835 GHz
Wireless Transmit Power	20 dBm
Antenna	4 dBI Detachable Omni Directional Antenna with RP-SMA connector
Security	64/128-bit WEP, WPA/WPA2, WPA-PSK/WPA2-PSK (TKIP/AES)
Dimensions	93.5 mm. × 26 mm. × 11 mm.)

Table 1.2: TP-Link TL-WN722N Technical features.

This Wifi-bed was chosen because if its versatility and its compatibility with the *Raspberry Pi Board*. Also, TL-WN722N is also capable to process every received packet, independent of its intended recipient (monitor-mode) and it is capable to send arbitrary packets (packet-injection). Other devices, with similar capabilities, are available, however the chipset is produced by Atheros, a long-established company in the industry [77].

Table 1.2 summarizes the main technical characteristics of the TL-WN722N.

Figure 1.11: The TP-Link TL-WN722N WLAN adapter.

1.5.0.4 IMU chipset

The Pololu AltIMU-10 1.12, is an IMU and altimeter that features *L3GD20* gyro and *LSM303DLHC* accelerometer and magnetometer, and an *LPS331AP* digital barometer. An I^2C interface accesses ten independent pressure, rotation, acceleration, and magnetic measurements that can be used to calculate the sensor's altitude and absolute orientation.

The *LPS331AP*, *L3GD20*, and *LSM303DLHC* have many configurable options, including selectable resolutions for the barometer and dynamically selectable sensitivities for the gyro, accelerometer, and magnetometer. Each sensor also has a choice of output data rates.

The nine independent rotation, acceleration, and magnetic readings provide all the data needed to make an attitude and heading reference system (AHRS). Readings from the absolute pressure sensor can be easily converted to altitudes, giving a total of ten independent measurements. With an appropriate algorithm, a microcontroller or computer, the data are used to calculate the orientation and height of the AltIMU board.

The gyro can be used to very accurately track rotation on a short timescale, while the accelerometer and compass can help compensate for gyro drift over time by providing an absolute frame of reference. The respective axes of the two chips are aligned on the board to facilitate these sensor fusion calculations.

The carrier board includes a low-dropout linear voltage regulator that provides the 3.3 V required by the *LPS331*, *L3GD20*, and *LSM303*, allowing the module to be powered from a single 2.5 V to 5.5 V supply. The board also includes a circuit that shifts the I^2C clock and data pines to the same logic voltage level, thus making it simple to interface the board with 5 V systems. The overall Pololu AltIMU-10 technical description is depicted in Table 1.3.

The *Raspberry Pi Board* is connected to the IMU throught the I^2C bus to the IMU where the Raspberry Pi Board plays the role of master and the IMU as the slave. Figure 1.15 shoes a more detailed connection between these both modules.

Figure 1.16 shows the the output of the accelerometer and gyrosope in static conditions. In this figure, it is notable that the accelerometer z-axis is compensate by the gravitational acceleration.

In general, the test-bed is composed of these tree modules before mentioned (IMU, WLAN Adapter and the *Raspberry Pi Board*). A PC for data analysis

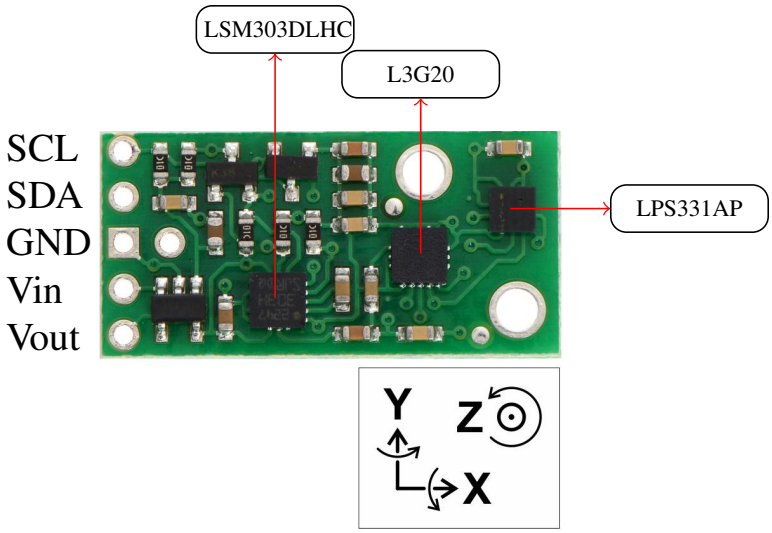


Figure 1.12: The Pololu AltIMU-10.

is connected to the main board via Ethernet for a data analysis of the measurements received and processed from the IMU and the WLAN adapter. The corresponding block diagram is shown in Figure 1.17.

1.5.1 Robot Integration

An integrated navigation information system must continuously know the current position with a good precision thus, a model is needed to measure

Dimensions	25 mm × 13 mm × 3 mm
Weight	0.8 grs
Supply current	6 mA
Output format ^2C	• Gyro: one 16-bit reading per axis • Accelerometer: one 16.bit reading per axis • Manetometer: one 16-bit reading per axis • Barometer: 24-bit pressure reading (4096 LSb/mbar)
Sensitivity Range	• Gyro: ± 245, ± 500, or ± 2000°/s • Accelerometer: ±2, ±4, ±6, ±8, or ±16g • Magnometer: ±2, ±4, ±6, or ±12 gauss • Barometer: 260 mbar to 1260 mbar

Table 1.3: Pololu AltIMU-10 Specifications

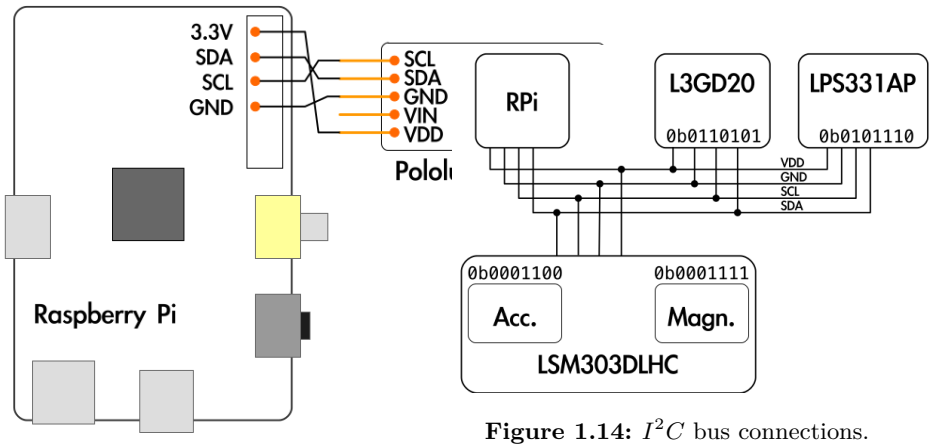


Figure 1.14: I^2C bus connections.

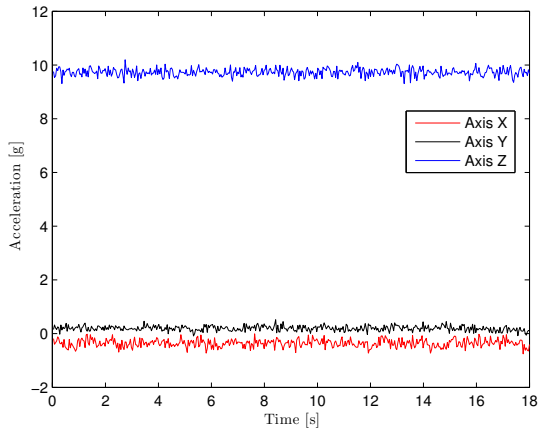
Figure 1.13: Physical connection

Figure 1.15: Diagram of the connections between the *Raspberry Pi Board* (RPi) and Pololu AltIMU-10 connections [3]

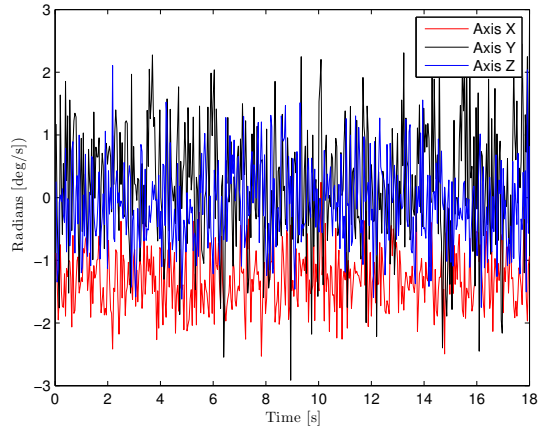
the real position. The chosen model is a four wheel Robot that is capable of performing a programmed trajectory through waypoints. The main board features an Arduino Platform. Arduino is an open-source platform and consists of a physical programmable circuit board (often referred to as a microcontroller) and an IDE (Integrated Development Environment) based in C++ language programming and used for software loading in the board. The initial robot position value for is the first real distance measurement given by the Robot used in the validation with Real data in the developed algorithms in this Thesis.

The rotation, acceleration and magnetic readings are independent providing an attitude and heading reference system. The respective axes of the two chips are aligned on the board to facilitate these sensor fusion calculations.

The test-bed is mounted on the top of the four wheel Robot. The Raspberry Pi microcontroller send the RSS and INS measurements to the data base of the server via Ethernet. In the Figure 1.18,a schematic of the overall system is presented where the ranging/positioning payload read RSS/INS measurements employing a TL-WN722N WiFi card and the Pololu AltIMU-10. Data fusion algorithms and ranging models can be implemented and tested in this platform or logged into a database for offline processing purposes.



(a) Accelerometer



(b) Gyroscope

Figure 1.16: Accelerometer and Gyroscope output signals in static condition.

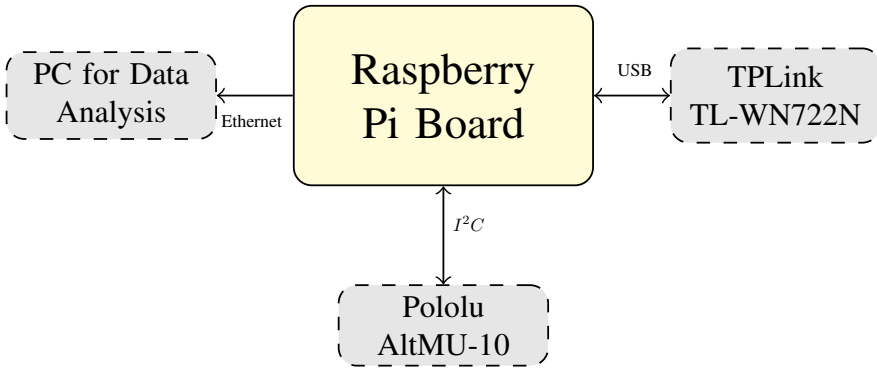


Figure 1.17: Global Connection scheme of the Test-Bed.

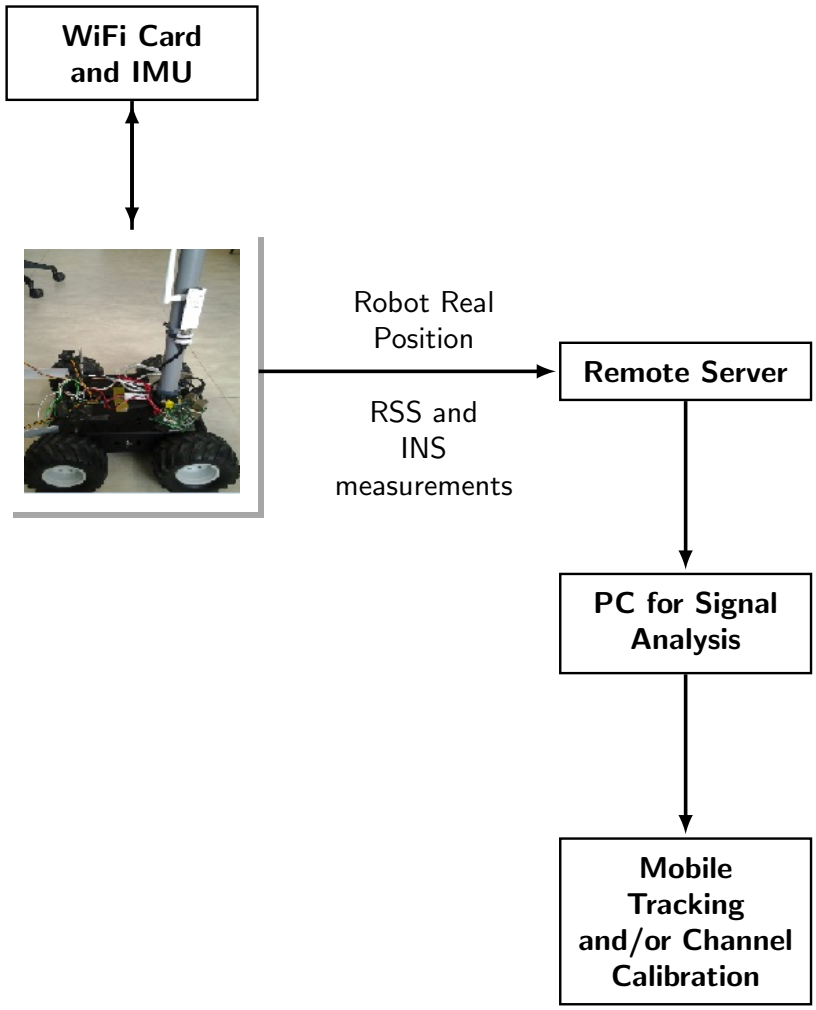


Figure 1.18: Connection diagram of the experimental system.

2

Parameter Estimation Theory

This chapter gives a brief summary of the theory in parameter estimation. We are primarily interested in Bayesian methods. The main feature of Bayesian methods is that instead of giving point estimates, try to quantify the estimation uncertainty in terms of posterior probability distributions which then can be adapted as when new data is available. Most of Bayesian approaches are based in MCMC because the parameter estimation problem can be solved in a statistical manner, and the whole distribution of the parameters can be explored, instead of obtaining point estimates using, e.g., Gaussian approximations.

Contents

2.1 Fundamental Bounds in Parameter Estimation

The accuracy at which the parameters can be determined depends on a variety of factors, such as measurement noise level, sampling rate, number of repeated experiments, etc. Knowing the achievable accuracy before hand is very important. For instance the production of the reagents for such experiments can often be very costly. Therefore, an efficient setup of the experimental conditions is of great importance to avoid unnecessary costs in executing the experiments.

We are interested in estimating a parameter vector $\boldsymbol{\psi}$ given the observations $\mathbf{y}$ where $\boldsymbol{\psi}$ and $\mathbf{y}$ are both random vectors with dimensions n_ψ and n_y, respectively. Let $\hat{\boldsymbol{\psi}}(\mathbf{y})$ be an estimator of $\boldsymbol{\psi}$, which is a function of $\mathbf{y}$. The Cramér Rao inequality [61] shows that the covariance matrix of $\hat{\boldsymbol{\psi}}$ can not go below a bound, which is expressed by:

$$\mathrm{Cov}(\hat{\boldsymbol{\psi}}) \triangleq \mathbb{E}\left\{(\hat{\boldsymbol{\psi}} - \boldsymbol{\psi})(\hat{\boldsymbol{\psi}} - \boldsymbol{\psi})^\top\right\} \geq \boldsymbol{\mathcal{I}}^{-1}(\boldsymbol{\psi}) \tag{2.1}$$

where operator $\mathbb{E}\{\cdot\}$ is the expected value. This lower bound on the variance of an estimated parameter is often referred to as the CRLB and it is defined as:

$$\mathbf{CRLB}(\hat{\boldsymbol{\psi}}) = \boldsymbol{\mathcal{I}}^{-1}(\boldsymbol{\psi}). \tag{2.2}$$

where $\boldsymbol{\mathcal{I}}(\boldsymbol{\psi})$ is defined as Fisher Information Matrix (FIM). The concept of Fisher information was first introduced by Fisher in [78] and then further elaborated on in [79] as he was trying to quantify how accurate parameters of a stochastic variable could be estimated from samples from the distribution.

The Fisher information for multidimensional parameters is sometimes called the FIM to more clearly indicate that it is a matrix and under the mild regularity condition that the *pdf* $p(\mathbf{y}|\boldsymbol{\psi})$ must have twice continuous partial derivatives, the second moment of the score [1] is the FIM and it is defined as [61]:

$$\begin{aligned}\boldsymbol{\mathcal{I}}(\boldsymbol{\psi}) &:= -\mathbb{E}\left\{\Delta_{\mathbf{x}}^{\boldsymbol{\psi}} \log p(\mathbf{y}|\boldsymbol{\psi})\right\} \\ &= \mathbb{E}\left\{\left(\nabla_{\boldsymbol{\psi}} \log p(\mathbf{y}|\boldsymbol{\psi})\right)\left(\nabla_{\boldsymbol{\psi}} \log p(\mathbf{y}|\boldsymbol{\psi})\right)^\top\right\} \qquad (2.3) \\ &= -\mathbb{E}\left\{\frac{\partial^2 \log p(\mathbf{y} \mid \boldsymbol{\psi}))}{\partial \boldsymbol{\psi}_i \partial \boldsymbol{\psi}_j}\right\} \; i,j = 1,...,N \qquad (2.4)\end{aligned}$$

[1] Score is the partial derivative with respect to $\boldsymbol{\psi}$ of the natural logarithm of the likelihood function.

where $\nabla_{\boldsymbol{\psi}}$ and $\Delta^{\boldsymbol{\psi}}_{\boldsymbol{\psi}}$ are the *Jacobian* and the *Hessian*[2] matrix respectively.

Inequality in (2.1) means that $\mathrm{Cov}(\hat{\boldsymbol{\psi}}) - \boldsymbol{\mathcal{I}}^{-1}(\boldsymbol{\psi})$ is a positive semidefinite matrix. Based on (2.1), The following statement can be established: Assuming that CRLB is attainable, any estimation algorithm working optimally (in the sense of the smallest covariance of obtained estimates), must give estimates whose variance is equal to the CRLB. If the CRLB is reachable, then the optimal result can be achieved by a maximum likelihood approach.

The CRLB sets a lower bound on the variance of any unbiased estimator. It is useful as follows:

1. If we find an estimator that achieves the CRLB, then we know that we have found a Minimun Variance Unbiased Estimator.
2. The CRLB can provide a benchmark against which we can compare the performance of any unbiased estimator.
3. The theory behind the CRLB can tell us if an estimator exists that achieves the lower bound.

2.1.1 Posterior Cramér-Rao Lower Bound

The Posterior Cramér-Rao Lower Bound is a natural extension to the classical CRLB for parameter estimation where the sequential Bayesian estimation problem is to find the estimate of the state from the measurements (observations) over time. Basically, treat the state at each time as a parameter with a unknown random variable (more precisely treat each process noise realization as a parameter from which the state can then be found) and derive a bound for how well these parameters can be estimated.

Let us consider a more general nonlinear filtering problem than that defined by (1.23) and (1.22) in Chapter 1.3.3.2. The evolution of the state sequence $\mathbf{x}_k$ is assumed to be an unobserved first order Markov process modeled as

[2]The derivative of a scalar function $f : \mathbb{R}^n \to \mathbb{R}^n$, often called gradient or the *Jacobian*, is: $\nabla_x f = \begin{bmatrix} \frac{\delta_x f_1}{\delta x_1} & \frac{\delta_x f_2}{\delta x_1} & \dots & \frac{\delta_x f_m}{\delta x_1} \\ \frac{\delta_x f_1}{\delta x_2} & \frac{\delta_x f_2}{\delta x_2} & \dots & \frac{\delta_x f_m}{\delta x_2} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\delta_x f_1}{\delta x_n} & \frac{\delta_x f_2}{\delta x_n} & \dots & \frac{\delta_x f_m}{\delta x_n} \end{bmatrix}$. With this definition, the second derivation of $f : \mathbb{R}^n \mapsto \mathbb{R}^n$ becomes $\Delta^y_x f = \nabla_x \nabla_y f = \begin{bmatrix} \frac{\delta^2 f}{\delta x_1 \delta y_1} & \dots & \frac{\delta^2 f}{\delta x_1 \delta y_m} \\ \vdots & \vdots & \vdots \\ \frac{\delta^2 f}{\delta x_n \delta y_1} & \dots & \frac{\delta^2 f}{\delta x_n \delta y_m} \end{bmatrix}$.

$$\mathbf{x}_k = \mathbf{f}_{k-1}(\mathbf{x}_{k-1}, \boldsymbol{\nu}_{k-1}) \tag{2.5}$$

where $\mathbf{f}_{k-1} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_\nu} \to \mathbb{R}^{n_x}$ is, in general, a nonlinear function of state $\mathbf{x}$ and $\boldsymbol{\nu}_{k-1}$ is an independent whine process noise with a dimension of n_u.

The observations about the state are obtained from the measurement equation:

$$\mathbf{y}_k = \mathbf{h}_k(\mathbf{x}_k, \boldsymbol{w}_k) \tag{2.6}$$

where $\mathbf{h}_k : \mathbb{R}^{n_x} \times \mathbb{R}^{n_w} \to \mathbb{R}^{n_y}$ is, in general, a nonlinear function of state $\mathbf{x}$ and $\boldsymbol{w}_{k-1}$ is an independent whine process noise with a dimension of n_w.

If we consider a sequence of target states (refers to as trajectory) $\mathbf{X}_k = \{\mathbf{x}_0, \mathbf{x}_1, ..., \mathbf{x}_k\}$ and its unbiased estimate $\hat{\mathbf{X}}_k$ based on the measurements $\mathbf{Y}_k = \{\mathbf{y}_1, ..., \mathbf{y}_k\}$, the joint *pdf* can be determinated from (2.5) and (2.6) with known prior *pdf* $p(\mathbf{x}_0)$:

$$p(\mathbf{X}, \mathbf{Y}) = p(\mathbf{x}_0) \prod_{i=1}^{k} p(\mathbf{x}_i|\mathbf{x}_{i-1}) \prod_{j=1}^{k} p(\mathbf{x}_j|\mathbf{x}_j) \tag{2.7}$$

If we define $\boldsymbol{\mathcal{J}}_k$ to be the $(k+1)n_x \times (k+1)n_x$ *trajectory* information matrix with dimension of $\mathbf{X}_k$ derived from the *pdf* $p(\mathbf{X}, \mathbf{Y})$, (2.1) becomes:

$$\mathbb{E}\left\{ \left(\hat{\mathbf{X}}_{k|k} - \mathbf{X}_x \right) \left(\hat{\mathbf{X}}_{k|k} - \mathbf{X}_x \right)^\top \right\} \succeq \boldsymbol{\mathcal{J}}_k^{-1}. \tag{2.8}$$

The trajectory information matrix $\boldsymbol{\mathcal{J}}_k$ is defined as [80]:

$$\boldsymbol{\mathcal{J}}_k \triangleq -\mathbb{E}\left(\Delta_{\mathbf{X}_k}^{\mathbf{X}_k} \log p(\mathbf{Y}_k|\mathbf{X}_k) \right) = \mathbb{E}\left(\left(\nabla_{\mathbf{X}_k} \log p(\mathbf{Y}_k|\mathbf{X}_k) \right) \left(\nabla_{\mathbf{X}_k} \log p(\mathbf{Y}_k|\mathbf{X}_k) \right)^\top \right), \tag{2.9}$$

where $\boldsymbol{\mathcal{J}}_k$ has a dimension of $(k+1)n_x \times (k+1)n_x$.

The reelationship between information matrices $\boldsymbol{\mathcal{J}}_k$ and $\boldsymbol{\mathcal{I}}_k$ is the first step in formulating the recursive solution for the computation of the FIM $\boldsymbol{\mathcal{I}}_k$. Remind that the inverse of $\boldsymbol{\mathcal{I}}_k$ is the CRLB. Suppose we decompose $\mathbf{X}_k = \left[\mathbf{X}_{k-1}^\top \mathbf{x}_k^\top \right]^\top$.

The trajetory information matrix $\boldsymbol{\mathcal{J}}_k$ can be accordingly decomposed into blocks:

$$\boldsymbol{\mathcal{J}}_k = \begin{bmatrix} \mathbf{A}_k & \mathbf{B}_k \\ \mathbf{B}_k^\top & \mathbf{C}_k \end{bmatrix} \tag{2.10}$$

where:

$$\mathbf{A}_k = -\mathbb{E}\left\{ \nabla_{\mathbf{x}_{0:k-1}} \left[\nabla_{\mathbf{x}_{k-1}} \log p(\mathbf{X}_k, \mathbf{Y}_k) \right]^\top \right\} \tag{2.11}$$

$$\mathbf{B}_k = -\mathbb{E}\left\{ \nabla_{\mathbf{x}_{0:k-1}} \left[\nabla_{\mathbf{x}_k} \log p(\mathbf{X}_k, \mathbf{Y}_k) \right]^\top \right\} \tag{2.12}$$

$$\mathbf{C}_k = -\mathbb{E}\left\{ \nabla_{\mathbf{x}_k} \left[\nabla_{\mathbf{x}_k} \log p(\mathbf{X}_k, \mathbf{Y}_k) \right]^\top \right\}. \tag{2.13}$$

It can be easily shown in [80] that the covariance of the error in estimating $\mathbf{x}_k$ is lower bounded by the $(n_x \times n_x)$ right-lower block of $\boldsymbol{\mathcal{J}}_k^{-1}$ that can be obtained directly from:

$$\boldsymbol{\mathcal{I}}_k = \boldsymbol{C}_k - \mathbf{B}_k^\top \mathbf{A}_k^{-1} \mathbf{B}_k. \tag{2.14}$$

However, this is not an efficient approach because we have to manipulate large matrices at each time k as depicted in Figure 2.1 and it can become in a notable problem in computing the Posterior Cramér Rao Lower Bound (PCRLB) and a derivation is needed.

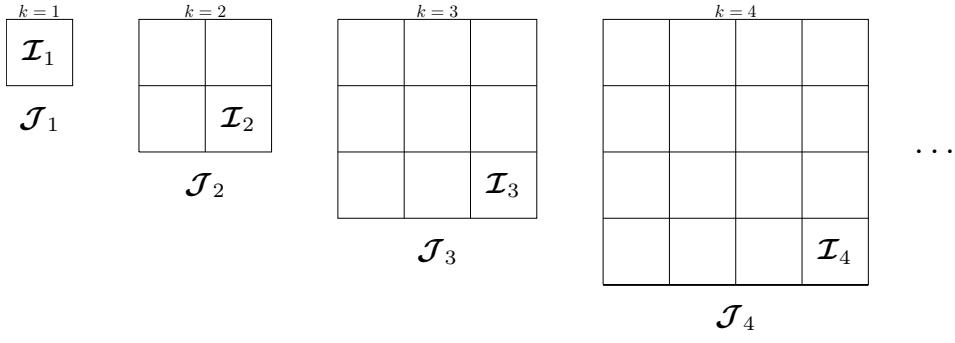


Figure 2.1: Growth of $\boldsymbol{\mathcal{I}}_k$ matrix at every time k [4]

2.1.2 Recursive computation of the PCRLB for Nonlinear Filtering.

Tichavsky et al [80] provided an elegant method to solve this problem by computing the information matrix $\boldsymbol{\mathcal{I}}_k$ recursively:

$$\boldsymbol{\mathcal{I}}_{k+1} = \mathbf{D}_k^{22} - \mathbf{D}_k^{21}(\boldsymbol{\mathcal{I}}_k + \mathbf{D}_k 11)^{-1}\mathbf{D}_k^{12} \tag{2.15}$$

where:

$$\mathbf{D}_k^{11} = -\mathbb{E}\left\{\Delta_{\mathbf{x}_k}^{\mathbf{x}_k} \log p(\mathbf{x}_{k+1}|\mathbf{x}_k)\right\}, \tag{2.16}$$
$$\mathbf{D}_k^{12} = -\mathbb{E}\left\{\Delta_{\mathbf{x}_k}^{\mathbf{x}_{k+1}} \log p(\mathbf{x}_{k+1}|\mathbf{x}_k)\right\}, \tag{2.17}$$
$$\mathbf{D}_k^{21} = -\mathbb{E}\left\{\Delta_{\mathbf{x}_{k+1}}^{\mathbf{x}_k} \log p(\mathbf{x}_{k+1}|\mathbf{x}_k)\right\} = \left[\mathbf{D}_k^{12}\right]^\top, \tag{2.18}$$
$$\mathbf{D}_k^{22} = -\mathbb{E}\left\{\Delta_{\mathbf{x}_{k+1}}^{\mathbf{x}_{k+1}} \log p(\mathbf{x}_{k+1}|\mathbf{x}_k)\right\} - \mathbb{E}\left\{\Delta_{\mathbf{x}_{k+1}}^{\mathbf{x}_{k+1}} \log p(\mathbf{y}_{k+1}|\mathbf{x}_{k+1})\right\}, \tag{2.19}$$

and the initialization is done considering the prior density of the states $p(\mathbf{x}_0)$ as follows:

$$\boldsymbol{\mathcal{I}}_0 = \mathbb{E}\left\{-\Delta_{\mathbf{x}_0}^{\mathbf{x}_o} \log p(\mathbf{x}_0)\right\} \tag{2.20}$$

Proof. The proof requires to establish the following property of joint densities and measurements:

$$\begin{aligned} p(\mathbf{X}_{k+1}, Y_{k+1}) &= p(\mathbf{x}_{k+1}, \mathbf{X}_k, \mathbf{y}_{k+1}, \mathbf{Y}_{k+1}) \\ &= p(\mathbf{y}_{k+1}|\mathbf{x}_{k+1}, \mathbf{X}_k, \mathbf{Y}_k)p(x_{k+1}|\mathbf{X}_k, \mathbf{Y}_k)p(\mathbf{X}_k, \mathbf{Y}_k) \\ &= p(\mathbf{y}_{k+1}|\mathbf{x}_{k+1}p(\mathbf{x}_{k+1}|\mathbf{x}_k)p(\mathbf{Y}_k|\mathbf{Z}_k). \end{aligned} \tag{2.21}$$

Descomposing $\mathbf{X}_{k+1}$ as $\mathbf{X}_{k+1} = \begin{bmatrix}\mathbf{X}_{k+1}^\top & \mathbf{x}_k^\top & \mathbf{x}_{k+1}^\top\end{bmatrix}$. Then the information matrix $\boldsymbol{\mathcal{I}}_{k+1}$ can be descomposed as:

$$\boldsymbol{\mathcal{J}}_{k+1} = \begin{bmatrix} \mathbf{A}_{k+1} & \mathbf{A}_{k+1} & \mathbf{A}_{k+1} \\ \mathbf{B}_{k+1} & \mathbf{A}_{k+1} & \mathbf{A}_{k+1} \\ \mathbf{A}_{k+1} & \mathbf{A}_{k+1} & \mathbf{A}_{k+1} \end{bmatrix} \tag{2.22}$$

where:

$$\begin{aligned}
\mathbf{A}_{k+1} &= -\mathbb{E}\left\{\Delta_{\mathbf{x}_{k-1}}^{\mathbf{x}_{k-1}} \log p(\mathbf{X}_{k+1}, \mathbf{Y}_{k+1})\right\} \\
&= -\mathbb{E}\left\{\Delta_{\mathbf{x}_{k-1}}^{\mathbf{x}_{k-1}} \left(\log p(\mathbf{X}_k, \mathbf{Y}_k) + \log p(\mathbf{x}_{k+1}|\mathbf{x}_k) + \log p(\mathbf{y}_{k+1}|\mathbf{x}_{k+1})\right)\right\} \\
&= -\mathbb{E}\left\{\Delta_{\mathbf{x}_{k-1}}^{\mathbf{x}_{k-1}} \log p(\mathbf{X}_k, \mathbf{Y}_k)\right\} + \mathbf{0} + \mathbf{0} \\
&= \mathbf{A}_k, \qquad (2.23) \\
\mathbf{C}_{k+1} &= -\mathbb{E}\left\{\Delta_{\mathbf{x}_k}^{\mathbf{x}_k} \log p(\mathbf{X}_k, \mathbf{Y}_k)\right\} \\
&= -\mathbb{E}\left\{\Delta_{\mathbf{x}_k}^{\mathbf{x}_k} \left(\log p(\mathbf{X}_k, \mathbf{Y}_k) + \log p(\mathbf{x}_{k+1}|\mathbf{x}_k) + \log p(\mathbf{y}_{k+1}|\mathbf{x}_{k+1})\right)\right\} \\
&= -\mathbb{E}\left\{\Delta_{\mathbf{x}_k}^{\mathbf{x}_k} \log p(\mathbf{X}_k, \mathbf{Y}_k)\right\} - \mathbb{E}\left\{\Delta_{\mathbf{x}_k}^{\mathbf{x}_k} \log p(\mathbf{x}_{k+1}|\mathbf{x}_k)\right\} + \mathbf{0} \\
&= \mathbf{C}_k + \mathbf{D}_k^{11}. \qquad (2.24)
\end{aligned}$$

In the same form, the remaining terms in (2.22) are:

$$\begin{aligned}
\mathbf{B}_{k+1} &= \mathbf{B}_k \qquad & \mathbf{E}_{k+1} &= \mathbf{D}_k^{12} \\
\mathbf{L}_{k+1} &= 0 \qquad & \mathbf{F}_{k+1} &= \mathbf{D}_k^{22}
\end{aligned} \qquad (2.25)$$

and now (2.22) can be rewritten as:

$$\boldsymbol{\mathcal{J}}_{k+1} = \begin{bmatrix} \mathbf{A}_k & \mathbf{B}_k & \mathbf{0} \\ \mathbf{B}_k^\top & \mathbf{C}_k + \mathbf{D}_k^{11} & \mathbf{D}_k^{12} \\ \mathbf{0} & \mathbf{D}_k^{21} & \mathbf{D}_k^{22} \end{bmatrix} \qquad (2.26)$$

Information matrix $\boldsymbol{\mathcal{I}}_{k+1}$ is computed as the inverse of the right-lower $(n_x \times n_x)$ submatrix of $\boldsymbol{\mathcal{J}}_{k+1}^{-1}$. Using the same approach as in (2.14), we have:

$$\begin{aligned}
\boldsymbol{\mathcal{I}}_{k+1} &= \mathbf{D}_k^{22} - \begin{bmatrix} \mathbf{0} & \mathbf{D}_k^{21} \end{bmatrix} \begin{bmatrix} \mathbf{A}_k & \mathbf{B}_k \\ \mathbf{B}_l^\top & \mathbf{C}_k + \mathbf{D}_k^{11} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{0} \\ \mathbf{D}_k^{12} \end{bmatrix} \qquad (2.27) \\
&= \mathbf{D}_k^{22} - \mathbf{D}_k^{21} \left[\mathbf{C}_k + \mathbf{D}_k^{11} - \mathbf{B}_k^\top \mathbf{A}_k^{-1} \mathbf{B}_k\right] \mathbf{D}_k^{12} \qquad (2.28)
\end{aligned}$$

Using the definition of $\boldsymbol{\mathcal{I}}_k$ of (2.14), the desired recursive formula (2.15) is proved. Note that the recursion of (2.15) involves computation with matrices of dimension $(n_x \times n_x)$ only, Thus, the computational complexity of the PCRLB is independent of the index time k

Conventional PCRLB considers the measurements as random vectors, and at any particular time k the bound is calculated by taking the average of both the measurements and the states up to time k. The recursive method for computation of the PCRLB is applicable to the most nonlinear filtering

problems defined by (2.5) and (2.6). Some special cases of these filtering problems are presented in the next section

2.1.2.1 System Model with Additive Gaussian Noise

Consider the system model:

$$\mathbf{x}_k = \mathbf{f}_{k-1}(\mathbf{x}_k) + \boldsymbol{\nu}_k \tag{2.29}$$
$$\mathbf{y}_k = \mathbf{h}_k(\mathbf{x}_k) + \boldsymbol{w}_k \tag{2.30}$$

where $\boldsymbol{\nu}_k$ and $\boldsymbol{w}_k$ are mutually independent, zero-mean white Gaussian noise with covariances $\mathbf{Q}_k$ and $\mathbf{R}_k$ respectively. An additional condition on covariances is that they have to be nonsingular (invertible).

So, according to these assumptions, we have:

$$\nabla_{\mathbf{x}_k} \log p(\mathbf{x}_k|\mathbf{x}_{k-1}) = \nabla_{\mathbf{x}_k} \left[-\frac{1}{2} \left[\mathbf{x}_k - \mathbf{f}_{k-1}(\mathbf{x}_{k-1})\right]^\top \mathbf{Q}_k^{-1} \left[\mathbf{x}_k - \mathbf{f}_{k-1}(\mathbf{x}_{k-1})\right] \right] \tag{2.31}$$
$$= \left[\nabla_{\mathbf{x}_k} \mathbf{f}_{k-1}^\top(\mathbf{x}_{k-1}) \mathbf{Q}_k^{-1} \left[\mathbf{x}_{k+1} - \mathbf{f}_{k-1}(\mathbf{x}_{k-1})\right]\right] \tag{2.32}$$

and similar with:

$$\nabla_{\mathbf{x}_k} = \log p(\mathbf{y}_k|\mathbf{x}_k) = \left[\nabla_{\mathbf{x}_k} \mathbf{h}_k^\top(\mathbf{x}_k) \mathbf{R}_k^{-1} \left[\mathbf{y}_k - \mathbf{h}_k(\mathbf{x}_k)\right]\right] \tag{2.33}$$

So following Equation (2.19), the matrices $\mathbf{D}_k^{11}$, $\mathbf{D}_k^{12}$, $\mathbf{D}_k^{22}$ are simplified as:

$$\mathbf{D}_{11} = \mathbb{E}\left\{\mathbf{F}_{k-1}^\top \mathbf{Q}_k^{-1} \mathbf{F}_{k-1}\right\} \tag{2.34}$$
$$\mathbf{D}_{12} = -\mathbb{E}\left\{\mathbf{F}_k^\top\right\} \mathbf{Q}_k^{-1} \tag{2.35}$$
$$\mathbf{D}_k^{22} = \mathbf{Q}_k^{-1} - \mathbb{E}\left\{\mathbf{H}_k^\top \mathbf{R}_k^{-1} \mathbf{H}_k\right\} \tag{2.36}$$

where $\mathbf{H}_k$ and $\mathbf{F}_{k-1}$ are the Jacobian matrices evaluates at the true value of $\mathbf{x}_k$ and $\mathbf{x}_{k-1}$ respectively. Because the model, matrices $\mathbf{D}_k^{11}$, $\mathbf{D}_k^{12}$ are deterministic and they can be easily to obtain. But the non-linearity of the measurements,

this feature cannot be applied to $\mathbf{D}_k^{22}$ so a Monte Carlo approximation has to be implemented.

2.1.2.2 System Model in a Linear Case

The linear Gaussian case is a special case of (2.30) where:

$$\mathbf{f}_{k-1}(\mathbf{x}_k) = \mathbf{F}_{k-1}\mathbf{x}_{k-1} \tag{2.37}$$
$$\mathbf{h}_k(\mathbf{x}_k) = \mathbf{H}_k\mathbf{x}_k \tag{2.38}$$

In this case, $\mathbf{D}_k^{11}$, $\mathbf{D}_k^{12}$, $\mathbf{D}_k^{22}$ are deterministic and the expectation operator in $\boldsymbol{D}_k^{22}$ can be dropped out yielding to

$$\mathbf{D}_{11} = \mathbf{F}_k^\top \mathbf{Q}_k^{-1}\mathbf{F}_k \tag{2.39}$$
$$\mathbf{D}_{12} = -\mathbf{F}_k^\top \mathbf{Q}_k^{-1} \tag{2.40}$$
$$\mathbf{D}_k^{22} = \mathbf{Q}_k^{-1} - \mathbf{H}_k^\top \mathbf{R}_k^{-1}\mathbf{H}_k \tag{2.41}$$

Substituting (2.41) into recursive Equation 2.15, we obtain:

$$\boldsymbol{\mathcal{I}}_{k+1} = \mathbf{Q}_k^{-1} + \mathbf{H}_k^\top \mathbf{R}_H^{-1}\mathbf{H}_k - \boldsymbol{\mathcal{I}}_k + \mathbf{F}_{k-1}^\top \mathbf{Q}_k^{-1}\mathbf{F}_{k-1} \tag{2.42}$$

Let us denote the information matrix $\boldsymbol{\mathcal{I}}_k$ by $\boldsymbol{P}_k|k^{-1}$, the inverse of a filter error covariance matrix. The application of the matrix inversion lemma[3] yields to:

$$\mathbf{P}_{k+1|k+1}^{-1} = \left(\mathbf{Q}_k + \mathbf{F}_{k-1}\mathbf{P}_{k|k}\mathbf{F}_k^\top\right)^{-1} + \mathbf{H}_k^\top \mathbf{R}_k^{-1}\mathbf{H}_{k+1} \tag{2.43}$$

Using matrix inversion lemma, it is seen that (2.43) is algebraically equivalent to (1.33). This shows is that the PCRLB for a linear Gaussian filtering problem is equivalent to the matrix of the Kalman filter.

[3]Matrix inversion Lemma: $\left(\mathbf{A} + \mathbf{B}\mathbf{C}\boldsymbol{B}^\top\right) = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{B}\left(\mathbf{B}^\top\mathbf{A}^{-1}\mathbf{B} + \mathbf{C}^{-1}\right)^{-1}\mathbf{B}^\top\mathbf{A}^{-1}$

2.2 Maximum Likelihood Estimation

The method of MLE is, one of the most popular technique for deriving estimators. Recall that the $\mathbf{y}_1, \mathbf{y}_2, .., \mathbf{y}_N$ are the N observed data with probability function $p(\mathbf{y}; \boldsymbol{\psi}_1, \boldsymbol{\psi}_2, .., \boldsymbol{\psi}_p)$, the likelihood function is defined by:

$$\mathcal{L}(\boldsymbol{\psi}; \mathbf{y}) = L(\boldsymbol{\psi}_1, \boldsymbol{\psi}_2, .., \boldsymbol{\psi}_p; \mathbf{y}_1, \mathbf{y}_2, .., \mathbf{y}_N) = \prod_{i=1}^{N} p(\mathbf{y}_i; \boldsymbol{\psi}_1, \boldsymbol{\psi}2, .., \boldsymbol{\psi}_p) \quad (2.44)$$

For a given $\mathbf{y}$, let $\hat{\boldsymbol{\psi}}(\mathbf{y})$ be a parameter value at which $\mathcal{L}(\boldsymbol{\psi}; \mathbf{y})$ attains its maximum with $\mathbf{y}$ held fixed.

Notice that, by this construction, the range of the MLE coincides with the range of the parameter. Intuitively, the MLE is a reasonable choice for an estimator. The MLE is the parameter value for which the observed sample is most likely. In general, the MLE is a good point estimator, possessing some of the optimality properties: *consistency*, *efficiency*, and *asymptotic normality* [34] .

If the likelihood function is differentiable (in $\boldsymbol{\psi}_i$), possible candidates for the MLE are the values of $(\boldsymbol{\psi}_1, \boldsymbol{\psi}_2.., \boldsymbol{\psi}_p)$ that solve

$$\frac{\delta}{\delta \boldsymbol{\psi}_i} \mathcal{L}(\boldsymbol{\psi}; \mathbf{y}) = 0, \quad i = 1, ..., p \quad (2.45)$$

Notice that the solution of (2.45) are only possible candidates for the MLE since the first derivative being 0 is only a necessary condition for a maximum, not a sufficient condition. Furthermore, the zeros of the first derivative locate only extreme points in the interior of the domain of a function. If the extreme occur on the boundary the first derivative may not be 0. Thus the boundary must be checked separately for extreme. In many cases, estimation is performed using a set of independent identically distributed measurements. The MLE estimator exhibits several characteristics which can be interpreted to mean that it is asymptotically optimal. These characteristics include:

- The MLE is asymptotically unbiased, i.e., its bias tends to zero as the number of samples increases to infinity.

- The MLE is asymptotically efficient, i.e., it achieves the Cramér-Rao lower bound when the number of samples tends to infinity.

- The MLE is asymptotically normal. As a number of samples increases, the distribution of the MLE tends to the Gaussian distribution with covariance matrix equal to the inverse of the FIM. In addition, this property makes possible to calculate, assuming some kind of Gaussianity, confidence ranges where the true value of the parameter is confined with a given probability.

2.3 Bayesian parameter estimation

This section addresses the design of an estimation method for parameter vector $\boldsymbol{\psi}$ following the Bayesian probability. A statistical model interprets the observations as random variables and assigns a probability distribution $p(\mathbf{y}|\boldsymbol{\psi})$ to them, where $\mathbf{y}$ is the observed data and $\boldsymbol{\psi}$ is an unknown parameter vector. This makes the model accessible to statistical methods.

One special case in statistical modeling is Bayesian modeling. It aims at maximizing the posterior distribution $p(\boldsymbol{\psi}|\mathbf{y})$ by including a prior knowledge, about probable values of unobserved $\boldsymbol{\psi}$ before observed data $\mathbf{y}$ known as distribution prior $\pi(\boldsymbol{\psi})$ So, defining a model in the spirit of Bayesian statistics includes defining a likelihood and a prior distribution.

The Bayesian viewpoint to parameter estimation considers the parameters as random variables which depend on the given data and which may vary according to our prior beliefs. In other words, starting with a predefined prior parameter distribution, the data is used to update our beliefs, and to arrive at a posterior parameter distribution which contains all information on the parameters that can be learned from the data.

In Bayesian statistics the posterior distribution $p(\boldsymbol{\psi}|\mathbf{y})$ contains all relevant information combining prior knowledge of the unknown parameters $\boldsymbol{\psi}$ given the observed data $\mathbf{y}$. The posterior distribution can be derived from the likelihood and the prior using the Bayes theorem:

$$p(\boldsymbol{\psi}|\mathbf{y}) = \frac{p(\mathbf{y}|\boldsymbol{\psi}) \cdot \pi(\boldsymbol{\psi})}{p(\mathbf{y})} = \frac{p(\mathbf{y}|\boldsymbol{\psi}) \cdot \pi(\boldsymbol{\psi})}{\int p(\mathbf{y}|\boldsymbol{\psi}) \cdot \pi(\boldsymbol{\psi}) d\boldsymbol{\psi}} \tag{2.46}$$

Due to the fact that the integral can be difficult to evaluate and that, in general, it is not necessary to know the exact posterior, in some cases we use that [81]:

$$p(\boldsymbol{\psi}|\mathbf{y}) \propto p(\mathbf{y}|\boldsymbol{\psi}) \cdot \pi(\boldsymbol{\psi}) \tag{2.47}$$

In Bayesian parameter estimation, there are three key quantities that we are often interested and they are defined via probability statements on the unknown variable of interest. These quantities are:

1. **Prior Predictive** $p(\mathbf{y})$

   The normalizing constant in Bayes theorem, $p(\mathbf{y})$ is a very important quantity,

   $$p(\mathbf{y}) = \int p(\mathbf{y}, \boldsymbol{\psi}) d\boldsymbol{\psi} = \int p(\mathbf{y}|\boldsymbol{\psi}) \cdot \pi(\boldsymbol{\psi}) d\boldsymbol{\psi} \tag{2.48}$$

   It represents the "evidence" for a particular model, and can be thought as the probability of observing the data that was observed before it was observed also known as the evidence or marginal likelihood.

2. **Marginal distribution** of a subset of parameters in a multivariate model

   Let $\boldsymbol{\psi} = [\boldsymbol{\psi}_1, \boldsymbol{\psi}_2, ..., \boldsymbol{\psi}_p]$ denote a p dimensional model and suppose we are interested in $\{p(\boldsymbol{\psi}_p|\mathbf{y}) : \boldsymbol{\psi}_p \in \boldsymbol{\psi}\}$. Then,

   $$\begin{aligned} p(\boldsymbol{\psi}_p|\mathbf{y}) &= \int p(\boldsymbol{\psi}_p, \boldsymbol{\psi}_{-p}|\mathbf{y}) d\boldsymbol{\psi}_{-p} \\ &= \int p(\boldsymbol{\psi}_p|\boldsymbol{\psi}_{p-1}, \mathbf{y}) p(\boldsymbol{\psi}_{-p}|\mathbf{y}) d\boldsymbol{\psi}_{-p} \end{aligned} \tag{2.49}$$

   where $\boldsymbol{\psi}_{-p}$ denotes the vector $\boldsymbol{\psi}$ with $\boldsymbol{\psi}_p$ removed

3. **Posterior predictions** The prediction for future unobserved response of the system $\tilde{\mathbf{y}}$ is based on the posterior predictive distribution, Then, the posterior predictive $p(\tilde{\mathbf{y}}|\mathbf{y})$ follows,

   $$\begin{aligned} p(\tilde{\mathbf{y}}|\mathbf{y}) &= \int p(\tilde{\mathbf{y}}|\boldsymbol{\psi}, \mathbf{y}) p(\boldsymbol{\psi}|\mathbf{y}) d\boldsymbol{\psi} \\ &= \int p(\tilde{\mathbf{y}}|\boldsymbol{\psi}) p(\boldsymbol{\psi}|\mathbf{y}) d\boldsymbol{\psi} \end{aligned} \tag{2.50}$$

   Note that $\tilde{\mathbf{y}}$,$\mathbf{y}$ are conditionally independent given $\boldsymbol{\psi}$; though clearly $p(\tilde{\mathbf{y}}, \mathbf{y})$ are dependent.

In summary, all statistical inference can be deduced from the posterior distribution by reporting appropriate summaries. This typically takes the form of evaluating integrals with the form of

$$\boldsymbol{J} = \int f(\boldsymbol{\psi}) p(\boldsymbol{\psi}|\mathbf{y}) d\boldsymbol{\psi} = \mathbb{E}_{p(\boldsymbol{\psi}|\mathbf{y})} \left[f(\boldsymbol{\psi}) \right] \tag{2.51}$$

of some logical indicator function $f(\boldsymbol{\psi})$ with respect to the posterior distribution.

As Bayesian models of cognitive phenomena become more sophisticated, the need for efficient inference methods becomes more urgent. The goal of Bayesian Inference is to maintain a full posterior probability distribution over a set of random variables. However, the major limitation in the implementation of Bayesian approaches is that obtaining the posterior distribution often requires the integration of high-dimensional functions and a numerical analysis is often infeasible. This can be computationally very difficult. This long-standing problem was solved by MCMC sampling, one of the major breakthroughs in 20th century statistics. MCMC outputs a sequence of parameter sets (Markov chain) whose empirical distribution, for long sequences, approximates (converges to) the posterior distribution.

Several standard approaches to generate such Markov chains exist, including Gibbs sampling, Metropolis - Hastings and reversible jump. Using these algorithms it is possible to implement posterior simulation in essentially any problem which allow pointwise evaluation of the prior distribution and likelihood function.

2.3.1 Introduction to Markov Chains Monte Carlo

MCMC sampling provides an elegant way to assess the parameters of a model, even if the corresponding posterior distribution is not accessible analytically. It outputs a sample of parameters whose empirical distribution, for long sequences, converges to the true posterior.

MCMC methods use computer simulation of Markov chains in the parameter space. The Markov chains are defined in such a way that the posterior distribution in the given statistical inference problem is the asymptotic distribution. This allows to use ergodic averages to approximate the desired posterior expectations. That is, any statistic of a posterior distribution can be computed according to M simulated samples.

The original Monte Carlo approach was a method developed by physicists to use random number generation to compute integrals and the simultaneously solves inference of $\{p(\boldsymbol{\psi}|\mathbf{y}), p(\boldsymbol{\psi}_p|\mathbf{y}), p(\tilde{\mathbf{y}}, \mathbf{y})\}$ (Equations 2.48 - 2.50). The problem is that these integrals with the form of (2.51) are usually impossible to evaluate analytically and then the parameter is multidimensional, even numerical methods may fail. So, the statistical sampling technique is directly relevant to solving difficult integrals in statistical inference.

MCMC methods construct a Markov chain on the state space $\boldsymbol{\psi} \in \boldsymbol{\Psi}$, whose steady state distribution is the posterior density of interest $p(\boldsymbol{\psi}|\mathbf{y})$ using a bag of samples drawn from the density.Then, MCMC procedures return a collection of M samples, $\left\{\boldsymbol{\psi}^{(1)}, \boldsymbol{\psi}^{(2)}, ..., \boldsymbol{\psi}^{(M)}\right\}$ where each sample can be assumed to e drawn from $p(\boldsymbol{\psi}|\mathbf{y})$.

The underlying logic of MCMC is to set up a Markov chain in $\boldsymbol{\psi}$ with ergodic distribution $p(\boldsymbol{\psi}|\mathbf{y})$. Starting with some initial state $\boldsymbol{\psi}^{(0)}$, we simulate M transitions under this Markov chain and record the simulates states $\boldsymbol{\psi}^{(m)}$, $m = 1, 2, .., M$. The ergodic sample average

$$\mathbb{E}_{p(\boldsymbol{\psi})}\left[f(\boldsymbol{\psi})\right] \simeq \frac{1}{M}\sum_{m=1}^{M} f(\boldsymbol{\psi}^{(m)}) \tag{2.52}$$

converges to the desired integral $\boldsymbol{J}$. Equation 2.52 is referred as *Monte Carlo integration*. Monte Carlo integration can be used to approximate posterior (or marginal posterior) distributions required for a Bayesian analysis.

The art of MCMC is to set up a suitable Markov chain with the desired posterior as stationary distribution and to judge when to stop simulation diagnosing when the chain has practically converged. The MCMC algorithms combines the *prior* distribution with the likelihood to obtain the *posterior* distribution. MCMC algorithms are typically run for a large number of iterations (to achieve the convergence to the target posterior) starting at an arbitrary $\boldsymbol{\psi}$.

As the name suggests, MCMC works by simulating a descrete-time Markov chain. That is, it produces a dependent sequence (a chain) of random variables $\left\{\boldsymbol{\psi}^{(m)}\right\}_{m=1}^{M}$, with approximate distribution

$$p(\boldsymbol{\psi}^{(m)}|\mathbf{y}) \approx p(\boldsymbol{\psi}|\mathbf{y}), \tag{2.53}$$

and the chain is initialized with a user defined starting value, $\boldsymbol{\psi}^{(0)}$.

A Markov chain is a stochastic process which yields a sequence of states where one state depends only on the directly preceding one (the so-called Markov property):

$$p(\boldsymbol{\psi}^{(m)}|\boldsymbol{\psi}^{(m-1)},...,\boldsymbol{\psi}^{(1)}) = p(\boldsymbol{\psi}^{(m)}|\boldsymbol{\psi}^{(m-1)}). \tag{2.54}$$

Possible transitions between the states are specified by a transition matrix $\mathbf{T}$

$$\mathbf{T}(\boldsymbol{\psi}^{(m)}|\boldsymbol{\psi}^{(m-1)}), \quad \text{with} \tag{2.55}$$

$$\sum_{\boldsymbol{\psi}^{(m)}} \mathbf{T}(\boldsymbol{\psi}^{(m)}|\boldsymbol{\psi}^{(m-1)}) = 1 \quad \forall \quad m \tag{2.56}$$

A fundamental role in MCMC simulation is that for any starting point, the chain will convergence to the invariant distribution $p(\boldsymbol{\psi})$, as long as the homogeneous transition matrix M is a stochastic transition matrix that obeys the following properties:

1. *Irreducibility.* For any state of the Markov chain, there is a positive probability of visiting all other states. That is, the matrix $\mathbf{T}$ cannot be reduced to separate smaller matrices, which is also the same as stating that the transition graph is connected.

2. *Aperiodicity.* The chain should not get trapped in cycles

If $\mathbf{T} \triangleq \mathbf{T}(\boldsymbol{\psi}^{(m)}|\boldsymbol{\psi}^{(m-1)})$ remains invariant for all m the Markov chain is called homogeneous. A distribution $p(\boldsymbol{\psi})$ is called invariant if the transition matrix is constructed such that after several steps and for any starting point the chain converges to this distribution. This is exactly the behavior which is desired if MCMC sampling is used to approximate a posterior distribution that can not be assessed otherwise. The detailed balance condition (or reversibility) is a sufficient but not necessary condition for the invariance of a target distribution $\boldsymbol{\psi}$:

$$p(\boldsymbol{\psi}^{(m)})\mathbf{T}(\boldsymbol{\psi}^{(m)}|\boldsymbol{\psi}^{(m-1)}) = p(\boldsymbol{\psi}^{(m-1)})\mathbf{T}(\boldsymbol{\psi}^{(m)}|\boldsymbol{\psi}^{(m-1)}) \tag{2.57}$$

and summing both sides over $\boldsymbol{\psi}^{(m-1)}$, gives:

$$p(\boldsymbol{\psi}^{(m)}) = \sum_{\boldsymbol{\psi}^{i-1}} p(\boldsymbol{\psi}^{(m-1)})\mathbf{T}(\boldsymbol{\psi}^{(m)}|\boldsymbol{\psi}^{(m-1)}) \tag{2.58}$$

Thus, by ensuring detailed balance, it is possible to ensure that a target distribution $p(\boldsymbol{\psi}^{(m)})$ is invariant and converges to the target density, $p(\boldsymbol{\psi}|\mathbf{y})$ as m gets "large". Broadly speaking, when m is small then $p(\boldsymbol{\psi}^{(m)})$ can often be "far" from $p(\boldsymbol{\psi}|\mathbf{y})$ (given an arbitrary starting value $\boldsymbol{\psi}^0$). In this case, we will want to discard the initial set of R samples, $\left\{\boldsymbol{\psi}^{(0)}, ..., \boldsymbol{\psi}^{(R-1)}\right\}$, as being unrepresentative of the steady state of the chain, $p(\boldsymbol{\psi}|\mathbf{y})$. The time (discarted iterations) R is known as *burn-in* period [82]. After this large number of iterations, the distribution of the observations generated from the simulation is approximately the target distribution.

MCMC samplers are irreducible and aperiodic Markov chains that have the target distribution as the invariant distribution. One way to design these samplers is to ensure that detailed balance is satisfied. However, it is also important to design samplers that converge quickly. However, several standard MCMC approaches to simulate samples, nonstandard, multivariate distributions. The most general procedures for MCMC simulation from a target distribution are the Metropolis - Hastings algorithm and the Gibbs sampler. Using one of these algorithms it is possible to implement posterior simulation in essentially any parameters inference problem which allow pointwise evaluation of the prior distribution and the likelihood function [83].

2.3.2 Implementation of the Gibbs sampler

Gibbs Sampler is named after the physicist J. W. Gibbs, in reference to the analogy between the sampling algorithm and statistical physics. The algorithm was devised by Geman and Geman [84].

The Gibbs sampling algorithm is one of the simplest MCMC algorithms and it is a special case of the Metropolis - Hastings Algorithms except that instead drawing from a proposal distribution for each dimension, then accepting or rejecting the proposed sample, we simply draw a value for that dimension according to the variable's corresponding conditional distribution [85]. The sampler can be efficient when the parameters are not highly dependent on each other and the full conditional distributions are easy to sample from.

It was named and introduced by [84] in the context of image processing demonstrating that this algorithm converges remarkably quickly for a wide

range of problems also. Having said this, the circumstance of using this algorithm for our propose essentially resides in its universality and not on any claim that is the most efficient method for any problem.

We consider a multi-variable case that $\boldsymbol{\psi} = (\boldsymbol{\psi}_1, \boldsymbol{\psi}_2, ..., \boldsymbol{\psi}_P)$ denotes the parameter vector. The general form of the Gibbs Sampler proceeds by iteratively for $p = \{1, 2, .., P\}$, generating from the conditional posterior distributions:

$$\boldsymbol{\psi}_p^{(m+1)} \sim p(\boldsymbol{\psi}_p | \boldsymbol{\psi}_p^{(m+1)}, ..., \boldsymbol{\psi}_{p-1}^{(m+1)}, \boldsymbol{\psi}_{p+1}^{(m)}, ..., \boldsymbol{\psi}_P^{(M)}, \mathbf{y}) \tag{2.59}$$

These distributions are called the full conditionals. The components can either be updated in random order or in a systematic order. The general algorithm for a multi-variable case is shown in algorithm 3.

Algorithm 3 Gibbs Sampler Algorithm for $\boldsymbol{\psi}$ inference in a multi variable case

Given the joint distribution $f(\boldsymbol{\psi}_1, \boldsymbol{\psi}_2, ..., \boldsymbol{\psi}_P)$
Specify an initial value for $\boldsymbol{\psi}_0^{(0)}; \boldsymbol{\psi}_2^{(0)}; ...; \boldsymbol{\psi}_P^{(0)}$
for iteration $p = 1, .., P$ **do**
  for iteration $m = 0, 1, 2, ..., M$ **do**
    Generate:
    $\boldsymbol{\psi}_p^{(m+1)} \sim p(\boldsymbol{\psi}_p | \boldsymbol{\psi}_p^{(m+1)}, ..., \boldsymbol{\psi}_{p-1}^{(m+1)}, \boldsymbol{\psi}_{p+1}^{(m)}, ..., \boldsymbol{\psi}_p^{(m)}, \mathbf{y})$
  end for
end for

2.3.3 Metropolis - Hastings Algorithm

MCMC sampling combines the Monte Carlo principle of approximating a distribution by drawing random samples with the principle of Markov chains, which offers a mathematical framework to ensure that the derived sample has the desired properties. In this setting, the unknown parameters are the states of the Markov chain, and a proposal function that suggests a new set of parameters based on the current one replaces the transition matrix T. The main challenge is to ensure that the Markov chain and the proposal function fulfill the required properties such that the desired posterior distribution is the invariant distribution of the chain. Various methods exists where Gibbs Sampler is one of them. The other method is the Metropolis - Hastings algorithm.

The success and popolarity of the Gibbs samples owes to the fact that in many statistical mdels, the complete conditional posterior distribution $p(\psi_p | \psi_l, p \neq$

$l, \mathbf{y}$) take the form of some well-known distributions allowing efficient random variate generation. But there remian many imporant applications where this is not the case, requiring alternative MCMC schemes. Possibly the most generic such scheme is the Metropolis - Hastingsalgorithm that has first been suggested in 1953 and further extended in 1970 [86, 87].

Consider generating from a posterior distribution $p(\boldsymbol{\psi}|\mathbf{y})$ where $\boldsymbol{\psi}$ denotes the current state of the Markov chain. A proposal $\tilde{\boldsymbol{\psi}}$ is generated from some proposal generating distribution $q(\tilde{\boldsymbol{\psi}}|\boldsymbol{\psi})$ that accepts a probability $p_a(\tilde{\boldsymbol{\psi}}|\boldsymbol{\psi})$ and restricting attention to transition operators satisfying detailed balance,

$$p_a(\tilde{\boldsymbol{\psi}}|\boldsymbol{\psi})q(\tilde{\boldsymbol{\psi}}|\boldsymbol{\psi})p(\boldsymbol{\psi}|\mathbf{y}) = p_a(\boldsymbol{\psi}|\tilde{\boldsymbol{\psi}})q(\boldsymbol{\psi}|\tilde{\boldsymbol{\psi}})p(\tilde{\boldsymbol{\psi}}|\mathbf{y}) \tag{2.60}$$

gives an equality constraint. The discussion of how to choose the proposal distribution $q(\,\cdot\,)$ is discussed later.

We also have inequalities that must hold for probabilities:

$$\begin{aligned} 0 \leq p_a(\tilde{\boldsymbol{\psi}}|\boldsymbol{\psi}) \leq 1 \quad &\text{and} \\ 0 \leq p_a(\boldsymbol{\psi}|\tilde{\boldsymbol{\psi}}) \leq 1, \quad & \end{aligned} \tag{2.61}$$

and optimizing the average acceptance probability $p(\boldsymbol{\psi}|\mathbf{y})p_a(\tilde{\boldsymbol{\psi}}|\boldsymbol{\psi}) + p(\tilde{\boldsymbol{\psi}}|\mathbf{y})p_a(\boldsymbol{\psi}|\tilde{\boldsymbol{\psi}})$ with respect to $q(\tilde{\boldsymbol{\psi}}|\boldsymbol{\psi})$ and $q(\boldsymbol{\psi}|\tilde{\boldsymbol{\psi}})$ must saturate one or more of the inequalities (2.61) (property of *linear programming problems*). Therefore, either $p_a(\tilde{\boldsymbol{\psi}}|\boldsymbol{\psi}) = 1$ or $p_a(\boldsymbol{\psi}|\tilde{\boldsymbol{\psi}}) = 1$ and $p_a(\tilde{\boldsymbol{\psi}}|\boldsymbol{\psi})$ is given by Equation 2.60, this gives the Metropolis - Hastings acceptance rule:

$$p_a(\tilde{\boldsymbol{\psi}}|\boldsymbol{\psi}) = \min\left\{1, \quad \frac{p(\tilde{\boldsymbol{\psi}}|\mathbf{y})}{p(\boldsymbol{\psi}|\mathbf{y})} \cdot \frac{q(\tilde{\boldsymbol{\psi}}|\boldsymbol{\psi})}{q(\boldsymbol{\psi}|\tilde{\boldsymbol{\psi}})}\right\} \tag{2.62}$$

According to the constraint of Equation 2.60, a pair of valid acceptance probabilities must have the same ratio $\frac{p_a(\tilde{\boldsymbol{\psi}}|\boldsymbol{\psi})}{p_a(\boldsymbol{\psi}|\tilde{\boldsymbol{\psi}})}$. Therefore they can be obtained by multiplying the Metropolis - Hastings probabilities by a constant less than one. This corresponds to adjusting the proposal distribution $q(\boldsymbol{\psi}|\tilde{\boldsymbol{\psi}})$ to suggest staying still more often harms the asymptotic variance of the chain: in this sense using Equation 2.62 is optimal.

Given this result it is unsurprising that Metropolis - Hastings has become almost synonymous with MCMC. In fact, a rich variety of more generic MCMC-based

algorithms exist and continue to be developed which many of these satisfy the detailed balance (Equation 2.60),

The choice of the proposal distribution $q(\boldsymbol{\psi}|\tilde{\boldsymbol{\psi}})$ is essentially arbitrary, subject only to some technical constraints. Using a symmetric proposal distribution with $q(\boldsymbol{\psi}|\tilde{\boldsymbol{\psi}}) = q(\boldsymbol{\psi}|\boldsymbol{\psi})$, for example a normal centered at $\boldsymbol{\psi}$, has the practical advantage that the ratio of proposal distributions $\frac{q(\tilde{\boldsymbol{\psi}}|\boldsymbol{\psi})}{q(\boldsymbol{\psi}|\tilde{\boldsymbol{\psi}})}$ cancels out of the expression for $p_a(\tilde{\boldsymbol{\psi}}|\boldsymbol{\psi})$. Often *Metropolis chain* is used to refer to this special case only. Another practically interesting variation is the use of an independent probing distribution $q(\tilde{\boldsymbol{\psi}})$, i.e., the proposal is independent of the current state. In general, it is possible to use suboptimal inference and learning algorithms to generate data-driven proposal distributions.

In comparative with other MCMC methods, the Gibbs sampler is a special case of the single-component Metropolis - Hastings algorithm where the proposals q are the full conditionals and the acceptance probability is always one. Algorithm 4 gives a procedure for simulating a Metropolis - Hastings Algorithm with stationary distribution $p(\boldsymbol{\psi}|\mathbf{y})$.

Algorithm 4 Metropolis - Hastings Algorithm

Specify an initial value for $\boldsymbol{\psi}^{(\mathbf{0})}$
for iteration $m = 0, 1, 2, ..., M$ **do**
  Sample $u \sim \mathcal{U}_{[0,1]}$
  Propose $\tilde{\boldsymbol{\psi}} \sim q(\tilde{\boldsymbol{\psi}}|\boldsymbol{\psi})$
  if $u < p_a(\tilde{\boldsymbol{\psi}}|\boldsymbol{\psi}^m) = \min\left\{1, \quad \frac{p(\tilde{\boldsymbol{\psi}}|\mathbf{y})}{p(\boldsymbol{\psi}^m|\mathbf{y})} \cdot \frac{q(\tilde{\boldsymbol{\psi}}|\boldsymbol{\psi})}{q(\boldsymbol{\psi}^m|\tilde{\boldsymbol{\psi}})}\right\}$ **then**
    $\boldsymbol{\psi}^{m+1} = \tilde{\boldsymbol{\psi}}$
  else
    $\boldsymbol{\psi}^{m+1} = \boldsymbol{\psi}^m$
  end if
end for

2.4 Conclusions

In this Chapter, an overview and references are given for parameter estimation techniques. MCMC methods based en Metropolis-Hasting Algorithms and Gibbs Sampling algorithm provide an efficient implementation where the proposal distribution adapts to the shape of the posterior and the methods become less dependent on the initial distribution. This methods provide new information (valuable information to the modeler) that includes correlations between parameters, uncertainties related to parameters, prediction distributions

and variance of the parameters demonstrating that MCMC sampling methods could provide good accuracy when experimental data is used.

The methodologies studied in this Chapter are used in the next Chapters. MCMC sampling method is proposed in Chapter 3 as a methodology for the channel model parameters estimation along the respective CRLB derivation to validate the computational results. The PCRLB, ML, EKF and IMM techniques explained also in this Chapter, are then widely used to build a Robust indoor localization algorithm in Chapter 4.

3

Two-Slope Path Loss Model

This chapter addresses the problem of determining the Cramér-Rao lower bound (CRLB) for the parameters and breakpoint distance in a Path-Loss Channel model for Received Signal Strength (RSS) measurements. The path loss model is usually assumed for corrupted RSS measurements due to the shadow fading channel feature. In this paper the two-slope path loss model is considered, in which RSS measurements are modeled differently for close and far distances. Closed-form expressions for the CRLB parameters are derived for unknown breakpoint distance. For unknown parameters and breakpoint distance value, a Bayesian estimation method is proposed. The CRLB is then compared with the performance of the herein proposed method.

Contents

3.1 Indoor Radio Propagation Model

Indoor RSS-based localization has become a popular solution, but standard techniques still consider a time invariant signal model with a priori known constant parameters. This standard RSS-based localization problem with known APs positions, a simple single slope path loss model and known model parameters, has already been solved in the literature using standard/fusion solutions [31, 28, 88, 52]. While some contributions considered the RSS-based localization problem using a single path loss model with unknown parameter [54, 89, 90, 40, 91, 92, 93], the general solution considering unknown path loss parameters and a generalized distance dependent measurement model is an important missing point.

Regarding the system specification, the most common is to use a classical channel model calibration [94, 95, 33, 96, 97, 98] for RSS observations and parameterized by a path loss exponent and a shadow fading but this is no useful in real-life applications where the indoor channel model are distance dependent and unknown to a certain extent. The path loss depends specifically on the distance between the transmitter (AP) and the receiver (Mobile Target) [99]. This effect is due to path reflections from the environment (specially reflections from walls and ceilings).

In [6, 23], a two-slope path loss model with known parameters for reducing the complexity of calibration process is considered. However, it has been observed in experimental campaigns that these parameters fluctuate and are indeed distance dependent [100, 101, 32] . As a conclusion, the parameters employed in the traditional path loss model are highly site-specific [41, 102, 16].

In this work, an extension of the classical path loss model accounting for two regions of propagation, referred to as the two-slope model [6, 23, 21] is considered to overcome the practical limitations of the standard case. The two-slope model was obtained by measurement campaigns using radio signal ray tracing methods, premeasured RSS contours centered at the receiver or multiple measurements at several base stations [22, 31, 42].

In practice, the system parameters are not perfectly known, what leads to poor localization performances when using the previous solution in real-life applications. The robust or adaptive localization problem implies the sequential position determination from a set of RSS measurements and the simultaneous model calibration or system model parameters estimation.

Few works have considered the localization problem using RSS measures together with the adaptive estimation of the propagation model parameters [54, 89, 90, 40, 91, 92], and these contributions only take into account simple propagation models.

In this contribution, a generalized two-slope path loss model to overcome the practical limitations of the conventional approach is used. This can be seen as an improvement of the standard localization case. To solve this problem a more sophisticated solution is needed to cope with the different distance-dependent RSS measurement models. No contributions in the literature face this generalized localization problem, thus the solution provided in this thesis can be simplified to optimally cope with it when model parameters are known also.

3.1.1 Measurement Model

The indoor radio propagation channel is highly unpredictable and time-varying, due to severe transmission impairments. The most significant transmission impairments for LOS transmission are [22]:

1. *Attenuation.* The strength of a signal decays with distance over any transmission environment. This reduction in strength or attenuation often is logarithmic.

2. *Free space loss.* In any wireless communication, the signal disperses with distance. A receiving antenna will receive less signal power the farther it is from the transmitting antenna. The transmitted signal attenuates over distance because the signal is being spread over a large area. This form of attenuation is known as free space loss.

3. *Fading.* Fading refers to the time variation of received signal power caused by changes in the transmission medium or path. Fading is the most challenging technical problem in designing a communication system. In a fixed environment, fading is affected by changes in atmospheric conditions. Whereas in a mobile environment where either the receiving or transmitting antenna is in motion relative to the other, the relative location of various obstacles changes with time, causing complex transmission effects.

4. Multipath: Multipath is defined as a propagation phenomenon that results in radio signals reaching the receiving antenna by two or more paths. The direct and reflected signals are often opposite in phase, which can result in a significant signal loss due to mutual cancelation in some

circumstances. Depending on the differences in the path lengths of direct and reflected waves, the composite signal can be either larger or smaller than the direct signal. Multipath is caused by the following propagation mechanisms:

- *Reflection.* Reflection occurs when a propagating electromagnetic wave impinges upon an object which has very large dimensions when compared to the wavelength of the propagating wave. The reflected waves may interfere constructively or destructively at the receiver.
- *Diffraction.* Diffraction occurs when the radio path between the transmitter and receiver is obstructed by a surface that is large compared to the wavelength of the radio wave.
- *Scattering.* Scattering occurs when the medium through which the wave travels consists of objects with dimensions that are small compared to the wavelength, and where the number of obstacles per unit volume is large. Scattered waves are produced by rough surfaces, small objects, or by other irregularities in the channel.

Propagation model can built the radio map and also account any changes in the environment. In order to evaluate the effectiveness of a given channel coding and processing technique before construction, a model of the channel that adequately describes the environment must be developed. Such analysis reduces the cost of developing a complex system by reducing the amount of hardware that has to be developed for performance evaluation.

There are several models in literature to characterize an indoor propagation model [19, 20]. A propagation model is a set of mathematical expressions,diagrams, and algorithms used to represent the radio characteristics of a given environment. The prediction models can be either empirical (also called statistical) or theoretical (also called deterministic), or a combination of these two. While the empirical models are based on measurements, the theoretical models deal with the fundamental principles of radio wave propagation phenomena.

In the empirical models, all environmental influences are implicitly taken into account regardless of whether they can be separately recognized. This is the main advantage of these models. Because deterministic models are based on the principles of physics they may be applied to different environments without affecting the accuracy. In practice, their implementation usually requires a huge database of environmental characteristics, which is sometimes either impractical or impossible to obtain. Both theoretical and measurement based propagation models indicate that average received signal power decreases logarithmically with distance. Empirical models help in reducing computational complexity as

well as increasing the accuracy of the predictions [22]. The empirical model used in this study is the two-slope path loss model. Some works consider a single path loss model with an inaccurate channel parameters assumption [103, 104].

The widely used model for RSS observatios is the path loss model, which is a simple yet realistic model for such measurements. It is parametetrizes by the path loss exponent (related to the power decat with respect to distance) and the shadowinf (that is, random propagation effects). However, it has been oberved in experimental campaigns that these parameters fluctuate and are indeed distance dependent. As a conclusion, the parameters employed in the traditional path loss model are highly site-specific [102, 41]. Therefore, in many situations more sophisticated model should be accounted.

In this tehsis, an extension of the classical path loss model accounting for two regions of propagation, referred to as the two-slope model [6], is considered to overcome the practical limitations of the standard case. The path loss depends specifically on the distance between the AP and the Mobile Target [99]. Indeed, it has been observed that for far distances ($5 \leq d \leq 30$ meters), exists a steeper overall drop in the RSS at the receiver known as breakpoint distance (d_{bp}). This effect is due to path reflections from the environment (specifically reflections from walls and ceilings)

The classical two-slope model assumes a linear dependence between the path loss expressed in Decibel (dB) and the logarithm of the distance d between the transmitter and the receiver as [6]:

$$RSS(d) = \begin{cases} h^{(1)}(d) & \text{if } d \leq d_{bp} \\ h^{(2)}(d_{bp}) + 10\alpha_2 \log_{10}(d/d_{bp}) & \text{if } d > d_{bp} \end{cases} \tag{3.1}$$

where the first equation gives the path loss (in decibels) for close distances (distances less than d_{bp}) and the second equation gives the path loss beyond d_{bp}. The α_1 and α_2 values are the slopes before and after d_{bp}, respectively. Also known as the path loss exponents. The function $h^{(1)}(d)$ applies to distances less than d_{bp} and has a slope of α_1 and corresponds to the one-slope model explained in Section 1.3.1.

Following the notation of $h(d) \triangleq L$ in (1.13) of Section 1.3.1 and assuming isotropic antennas, (1.13) becomes for $d \leq d_{bp}$,

$$h^{(1)}(d) = L_o + 10\alpha_1 \log_{10}(d) \ , \tag{3.2}$$

and, for $d > d_{bp}$,

$$h^{(2)}(d) = h^{(1)}(d_{bp}) + 10\alpha_2 \log_{10}(d/d_{bp}). \tag{3.3}$$

For any given transmitter/receiver configuration, the regions surrounding these stations can differ, resulting in the received signal with strength differing from the nominal (3.1). This variation (known as shadow fading or log-normal shadowing) can be modeled by an additive zero-mean Gaussian random variable. The notation, $\chi_{\sigma^2} \sim \mathcal{N}(0, \sigma^2)$ is used.

Then, including shadow fading to (3.1), the model is

$$RSS(d) = \begin{cases} h^{(1)}(d) + \chi_{\sigma_1^2} & \text{if } d \leq d_{bp} \\ h^{(2)}(d) + \chi_{\sigma_2^2} & \text{if } d > d_{bp} \end{cases} \tag{3.4}$$

As happens for the path loss exponents, the variance values differ before and after the breakpoint distance. Typically the values depend on the scenario but in all cases it is observed that $\sigma_1^2 < \sigma_2^2$ and $\alpha_1 < \alpha_2$. This model can be used in computer simulation to provide received power levels for random locations in communication system design and analysis

The standard deviation of the received power before and after breakpoint distance, σ_1 and σ_2, is expressed in units of dB and is assumed relatively constant with distance. Figure 3.10 represents a simulation of real measurements taken from the two-slope path loss model at different relative distances between the AP and the sensing device.

The multipath and shadowing fading present in indoor environments when the base-portable distance increases, the number of intervening obstacles and other reasons, make that path loss model do not always hold. Because this particular reason, the parameters employed in this model is site-specific [41] [102]. To sum up, the two-slope RSS measrumente model is parameterized and fully determined by the following set:

$$\boldsymbol{\psi} = \begin{bmatrix} \alpha_1 & \alpha_2 & \sigma_1^2 & \sigma_2^2 & d_{bp} \end{bmatrix}^\top \tag{3.5}$$

There are several models in litertarute to characterize the radio propagation channel model [19, 20]. This study is focused on IEEE802.11x channel model becase does not require an accurate floor plan of the building and can be implemented without using a third party software and the already dense deployment of APs in indoor environments [21].

IEEE802.11x is a family of specifications for WLANs developed by the Institute of Electrical and Electronics Engineers (IEEE) originally based on the HiperLAN/2 channel models [93]. They were then updated to provide a better representation of indoor environments [41] such as small offices and residences providing six parameterizations of the model (termed A to F) operating in the

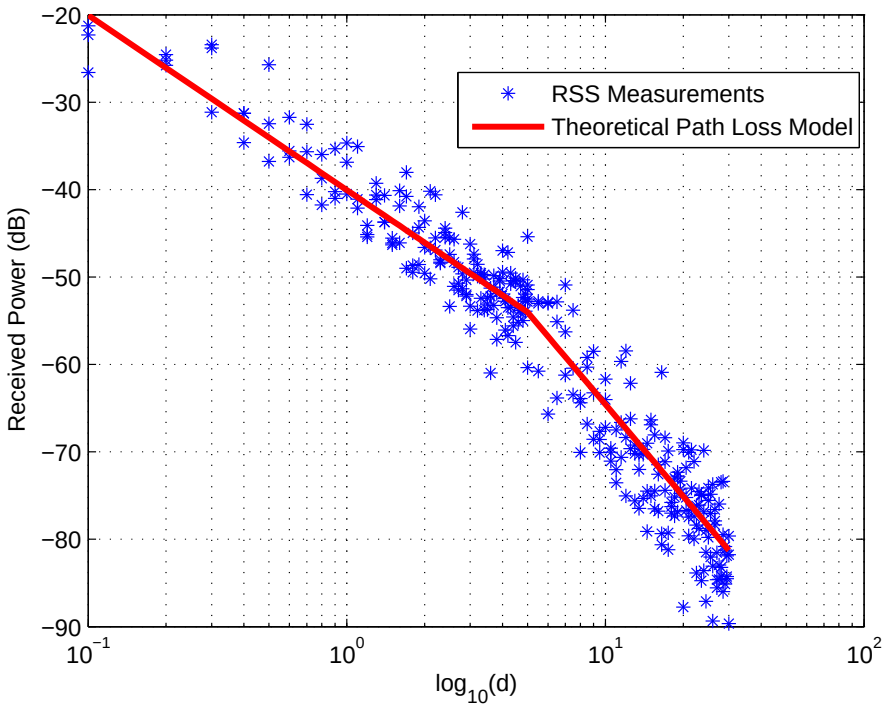


Figure 3.1: Simulation of a two-slope path loss model with $\sigma_1 = 3$ dB, $\sigma_2 = 5$ dB, $\alpha_1 = 2$, $\alpha_2 = 3.5$, and $d_{bp} = 5$ meters. Four RSS measurements per distance were recorded.

ultra high frequency band, specifically between 2.4 and 2.5 GHz, and having throughput rates up to 2 Mbps. A brief description of path loss parameters obtained for IEEE 802.11x is given in Table 3.1.

3.2 CRLB for two-slope path loss model

In this section, we develop the CRLB (introduced in Section 2.1) for the unknown parameters $\boldsymbol{\psi}$ and a brief resume is repeated here for convenience. It is important to note that the CRLB derived in this section considers the case of unknown d_{bp}.

It is considered the typical case where N independent observations are recorded at known distances to the AP under calibration, $\{d_n\}_{n=1}^{N}$ and

Model	Breakpoint distance d_{bp}	Path Loss Exponent before d_{bp} α_1	Path Loss Exponent after d_{bp} α_2	Shadow Fading before d_{bp} σ_1^2	Shadow Fading after d_{bp} σ_2^2
A	5	2	3.5	3	4
B	5	2	3.5	3	4
C	5	2	3.5	3	5
D	10	2	3.5	3	5
E	20	2	3.5	3	6
F	30	2	3.5	3	6

Table 3.1: Typical Pat Loss Parameters for IEEE802.11x channel models [6].

$\boldsymbol{y} = \{y_n \triangleq RSS(d_n)\}_{n=1}^{N}$. The lower bounds are useful as a benchmark to judge estimation methods as well as to evaluate their consistency. If $\log p(\boldsymbol{y}|\boldsymbol{\psi})$ is twice differentiable with respect to $\boldsymbol{\psi}_k$, the FIM (Equation 2.4) can be written as:

$$\boldsymbol{\mathcal{I}}(\boldsymbol{\psi}) = \mathbb{E}\left\{ \left[\nabla_{\boldsymbol{\psi}} \log p(\boldsymbol{y}|\boldsymbol{\psi})\right] \left[\nabla_{\boldsymbol{\psi}} \log p(\boldsymbol{y}|\boldsymbol{\psi}\right]^{\top} \right\} \tag{3.6}$$

The inverse of the FIM gives a lower bound, the CRLB, on the covariance matrix of any unbiased estimator $\hat{\boldsymbol{\psi}}$ [105, 106, 107]. The CRLB is used as a benchmark to evaluate the performance of an estimation algorithm and can provide guidance to improve the experimental design. [107] provide a method of deriving the CRLB describing the use of the FIM. More precisely

$$\boldsymbol{CRLB}(\hat{\boldsymbol{\psi}}) \succeq \boldsymbol{\mathcal{I}}^{-1}(\boldsymbol{\psi}) \tag{3.7}$$

Following the derivation of the CRLB for $\boldsymbol{\psi}$ in Appendix A, we have the final form of the lower bound obtained as the inverse of (3.7). This results in:

$$\mathbf{CRLB}(\boldsymbol{\psi}) = \begin{pmatrix} M_{11} & 0 & M_{13} & 0 & M_{15} \\ 0 & \frac{1}{I_{22}} & 0 & 0 & 0 \\ M_{31} & 0 & M_{33} & 0 & M_{35} \\ 0 & 0 & 0 & \frac{1}{I_{44}} & 0 \\ M_{51} & 0 & M_{52} & 0 & M_{55} \end{pmatrix} \tag{3.8}$$

where,

$$M_{11} = \frac{I_{11}I_{33}M_{CRLB} + [I_{15}(I_{11}I_{33} - I_{13}^2) - I_{13}(I_{11}I_{35} - I_{13}I_{15})]^2}{I_{11}(I_{11}I_{33} - I_{13}^2)M_{CRLB}} \tag{3.9}$$

$$M_{33} = \frac{I_{11}(I_{11}I_{55} - I_{15}^2)}{M_{CRLB}}, \tag{3.10}$$

$$M_{55} = \frac{I_{11}(I_{11}I_{33} - I_{13}^2)}{M_{CRLB}}, \tag{3.11}$$

and

$$M_{CRLB} = (I_{11}I_{55} - I_{15}^2)(I_{11}I_{33} - I_{13}^2) - (I_{11}I_{35} - I_{13}I_{15})^2 \tag{3.12}$$

The diagonal contains the values of the variance bounds of the parameter $\boldsymbol{\psi}$. The crossed values for α_1 and α_2 and with respect to d_{bp} (elements M_{13}, M_{15} and M_{35}), appear because each part of the two-slope path loss model depends on α_1.

3.3 Inference of Model Parameters

This section addresses the design of an estimation method of $\boldsymbol{\psi}$. The statistical problem is to detect the change point in the means and variances of the RSS measurements, as well as estimating the other channel model parameters.

One special case in statistical modeling is the Bayesian modeling. It aims ay maximizing the posterior distribution $p(\boldsymbol{\psi}|\boldsymbol{y})$ by including prior knowledge on $\boldsymbol{\psi}$ given as probability distribution prior $\pi(\boldsymbol{\psi})$. So, defining a model in the spirit of Bayesian statistics includes defining a likelihood and a prior distribution.

3.3.1 Bayesian Parameter Estimation. Problem Formulation

The measurements follow the Gaussian distribution discussed in Chapter 2.3. Recall that the particularity is that mean and variance may change at one specific distance. Since the Bayesian approach is followed, an *a priori* distribution for $\boldsymbol{\psi}$ has to be set. When considering Bayesian identification, $\boldsymbol{\psi}$ is modelled as an unobserved random variable, we place a prior $\pi(\boldsymbol{\psi})$ on $\boldsymbol{\psi}$ factorized as:

$$\boldsymbol{\psi} \sim \pi(\boldsymbol{\psi}) = \pi(\alpha_1)\pi(\alpha_2)\pi(\sigma_1^2)\pi(\sigma_2^2)\pi(d_{bp}) \tag{3.13}$$

An uniform *prior* over the full range of possible distances (defined as $d_{\max}$) is assumed for the d_{bp}, and conjugated priors are given to the path losses and the variances:

$$\begin{aligned} \alpha_i &\sim \pi(\alpha_i) = \mathcal{N}(0, \mathcal{V}_{\alpha_i}^2) \\ \sigma_i^2 &\sim \pi(\sigma_i^2) = \Gamma^{-1}(a_i, b_i) \\ d_{bp} &\sim \pi(d_{bp}) = \mathcal{U}(0, d_{\max}) \ , \end{aligned} \tag{3.14}$$

where $i = \{1, 2\}$. $\mathcal{V}_{\alpha_i}^2$, a_i and b_i control the initial uncertainty on the parameters of the model. This section is concerned in finding the unknown parameters $\boldsymbol{\psi}$ based on a batch of N measurements.

The problem of inferring the posterior distribution over the latent variables α_1, σ_1^2, α_2 and σ_2^2 could be solved analytically via Bayes theorem [81].

Since little knowledge is assumed, $\mathcal{V}_{\alpha_i}^2$=0.0001, a_i=0.1 and b_i=0.0001 values are used in the results section.

The statistical model was implemented with Gibbs Sampling [83, 108, 85, 109], which provides the joint posterior distribution of interest. The motivation of Gibbs sampler is that given a multivariate distribution, it is simpler sample from conditional distribution than to integrate over a joint distribution.

To estimate the posterior moments of the type of the Equation 2.51 we define a Markov chain in $\boldsymbol{\psi}$ (defined in 2.52). Further, to implement the Gibbs sampler, the posterior conditional for each parameter in the model and then sample from them is required. They are obtained as:

$$\alpha_1 \sim p(\alpha_1|\sigma_1^2, \alpha_2, \sigma_2^2, d_{bp}, \boldsymbol{y}) = \mathcal{N}\left(\alpha_{1,o}, \sigma_{1,0}^2\right), \tag{3.15}$$

$$\sigma_1^2 \sim p(\sigma_1^2|\alpha_1, \alpha_2, \sigma_2^2, d_{bp}, \boldsymbol{y}) = \Gamma^{-1}\left(a_{1,o}, b_{1,o}\right), \tag{3.16}$$

$$\alpha_2 \sim p(\alpha_1|\sigma_1^2, \alpha_2, \sigma_2^2, d_{bp}, \boldsymbol{y}) = \mathcal{N}\left(\alpha_{2.o}, \sigma_{2,0}^2\right), \tag{3.17}$$

$$\sigma_2^2 \sim p(\sigma_1^2|\alpha_1, \alpha_2, \sigma_2^2, d_{bp}, \boldsymbol{y}) = \Gamma^{-1}\left(a_{2,o}, b_{2,o}\right), \tag{3.18}$$

$$d_{bp} \sim p(d_{bp}|\alpha_1, \sigma_1^2, \alpha_2, \sigma_2^2, \boldsymbol{y}) \propto p(\boldsymbol{y}|\boldsymbol{\psi}) \cdot \pi(d_{bp}) \ , \tag{3.19}$$

where,

$$\alpha_{1,o} = 20\frac{\sigma_{1,0}^2}{\sigma_1^2}\left(\sum_{n=1}^{N_{dbp}} y_n \log_{10} d_n - L_o \sum_{n=1}^{N_{dbp}} \log_{10} d_n)\right), \tag{3.20}$$

$$\sigma_{1,0}^2 = \frac{\sigma_1^2 \mathcal{V}_{\alpha_1}^2}{100\mathcal{V}_{\alpha_1}^2 \sum_{n=1}^{N_{dbp}} \log_{10}^2 d_n + \sigma_1^2}, \tag{3.21}$$

$$\begin{aligned}\alpha_{2,o} =& 20\frac{\sigma_{2,o}^2}{\sigma_2^2}\left[\alpha 2 \sum_{n=N_{dbp}+1}^{N} y_n \log_{10}\frac{d_n}{d_{bp}} - \right. \\ & \left. - (L_o + 10\alpha_1 \log_{10} d_{bp}) \sum_{n=N_{dbp}+1}^{N} \log_{10}\frac{d_n}{d_{bp}}\right],\end{aligned} \tag{3.22}$$

$$\sigma_{2,0}^2 = 100\frac{\sigma_2^2 \mathcal{V}_{\alpha_2^2}^2}{\mathcal{V}_{\alpha_2}^2 \sum_{n=N_{dbp}+1}^{N} \log_{10} \frac{d_n}{d_{bp}}}, \tag{3.23}$$

and thus the Bayesian solution based on Gibbs Sampling can be easily implemented where N refers to the total number of RSS measurements and N_{dbp} the number of measurements before the breakpoint distance. The mathematical expressions for $\alpha_{1,o}$,$\sigma_{1,o}$,$\alpha_{2,o}$ and $\sigma_{2,o}$ are derived in Appendix B.

Now, given the nature of a MCMC, all we need to define is the transition probability. Given a current value for $\boldsymbol{\psi}^{(m)}$ after m transitions, we need to generate a new value $\boldsymbol{\psi}^{(m+1)}$. We do so by sampling from the complete posterior conditional distributions for $\alpha_1, \alpha_2, \sigma_1^2, \sigma_2^2$ and d_{bp} (Equation 3.15-3.19). This process continues until convergence to $p(\alpha_1, \sigma_1^2, \alpha_2, \sigma_2^2, d_{bp}|\boldsymbol{y})$ as desired. For this reason, Gibbs Sampling algorithms is tipically run for a large number of iterations. Ergodic values of the type of Equation 2.52 provide numerical evaluations of any desired posterior distribution.

The MCMC simulation is a special case of a Gibss Sampling. In other words, we assume that $\boldsymbol{\psi} = (\psi_1, \psi_2, \psi_3, \psi_4, \psi_5)$ denotes the parameter vector. Algorithm 5 gives the generic details of the Gibbs sampler algorithm implemented in this work.

Algorithm 5 Gibbs Sampler Algorithm for $\boldsymbol{\psi}$ inference

Given the joint distribution $p(\psi_1, \psi_2, \psi_3, \psi_4, \psi_5)$
Initialize $\psi_1^{(0)} = \alpha_1^{(0)}, \psi_2^{(0)} = \alpha_2^{(0)}, \psi_3^{(0)} = \sigma_1^{2,(0)}, \psi_4^{(0)} = \sigma_2^{2,(0)}$ and $\psi_5^{(0)} = d_{bp}^{(0)}$
for iteration $m = 0, 1, 2, ..., M$ **do**
Generate:
$\alpha_1^{(m+1)} \sim p(\alpha_1 | \sigma_1^{2,(m)}, \alpha_2^{(m)}, \sigma_2^{2,(m)}, d_{bp}^{(m)}, \boldsymbol{y})$
$\sigma_1^{2,(m+1)} \sim p(\sigma_1^2 | \alpha_1^{(m+1)}, \alpha_2^{(m)}, \sigma_2^{2,(m)}, d_{bp}^{(m)}, \boldsymbol{y})$
$\alpha_2^{(m+1)} \sim p(\alpha_1 | \sigma_1^{2,(m+1)}, \alpha_2^{(m+1)}, \sigma_2^{2,(m)}, d_{bp}^{(m)}, \boldsymbol{y})$
$\sigma_2^{2,(m+1)} \sim p(\sigma_1^2 | \alpha_1^{(m+1)}, \alpha_2^{(m+1)}, \sigma_2^{2,(m+1)}, d_{bp}^{(m)}, \boldsymbol{y})$
$d_{bp}^{(m+1)} \sim p(d_{bp} | \alpha_1^{(m+1)}, \sigma_1^{2,(m+1)}, \alpha_2^{(m+1)}, \sigma_2^{2,(m+1)}, \boldsymbol{y})$
end for

3.4 Evaluation and Results

3.4.1 Simulation Results

In this work the model IEEE 802.11c is used with the parameters set as in Table 3.1. These values represent a typical office environment. The Gibbs sampler for the breakpoint distance model was implemented using WinBugs [110]. Two MCMC chains were simulated, each with 10^4 Monte Carlo runs and 300 *burn-in* samples. The performance of the proposed algorithm was evaluated with 250 iterations to obtain the RMSE of each parameter in $\boldsymbol{\psi}$. These simulations illustrate the convergence shape of the RMSE according to its corresponding CRLB plot.

Two experiments were performed:

1 The experiment consisting in evaluating the effect of the number of RSS measurements (N) on the $\hat{\boldsymbol{\psi}}$ inference. In this simulation, the distance $d_{\max} = 20$ meters was fixed and divided from 10 to 100 to obtain several interval distances. Per every interval distance, one RSS sample was taken and in this way, the granularity of RSS measurements in distance was adjusted.

2 In the second experiment, $d_{\max}$ was divided in 10 interval (distance interval fixed in 20 centimeters) and the amount of RSS measurements was modified by increasing from 1 to 15 samples per each interval distance.

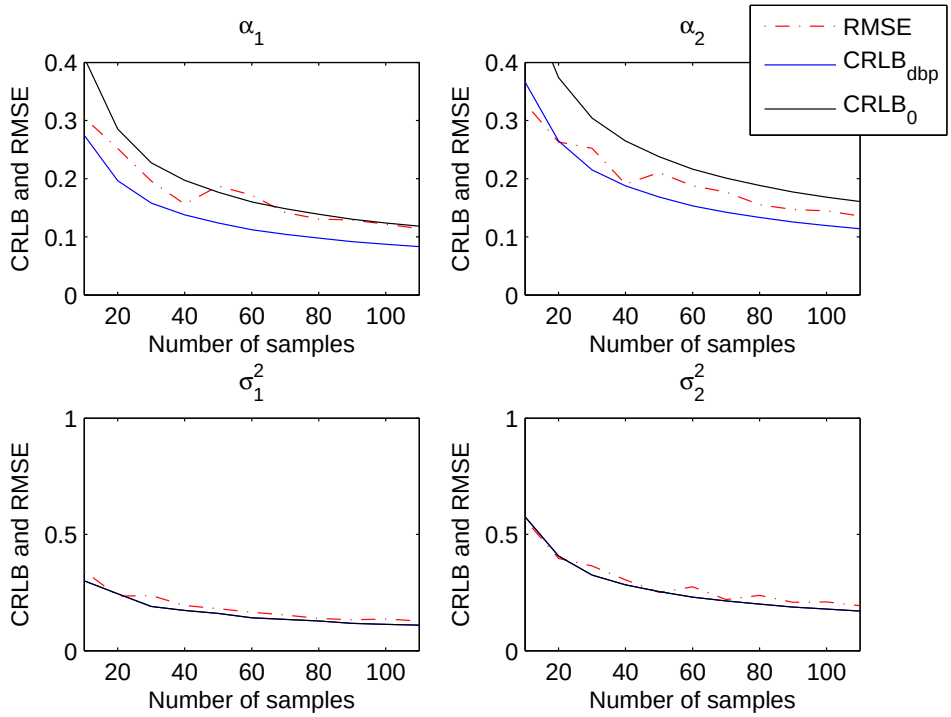


Figure 3.2: Simulation results corresponding to Experiment 1. Lower bounds of α_1, σ_1^2, α_2, σ_2^2, computed when the interval distance between each RSS measurements decreases in every iteration.

The results of the first experiment are shown in Figure 3.2. The RMSE for every latent variable in comparative with their respective CRLB is plotted. The label $CRLB_0$ refers to the case when the parameters were estimated taking each part of the model separate and independently and a known d_{bp} value (theoretical according to Table 3.1). $CRLB_{dbp}$ is the lower bound derived in this work.

It can be observed that the RMSE of the proposed algorithm attains the derived CRLB, regardless the assumptions of the bound are more optimistic when d_{bp} is known. Figure 3.3 is for the second experiment with the same comparative as well as the first one.

According to every before established experiment, Figure 3.4 and 3.5 show the RMSE values for d_{bp} and its corresponding lower bound.

These results show that the inference of $\boldsymbol{\psi}$ parameters achieve an asymptotically efficient performance. When RSS measurements samples are taken for fixed N,

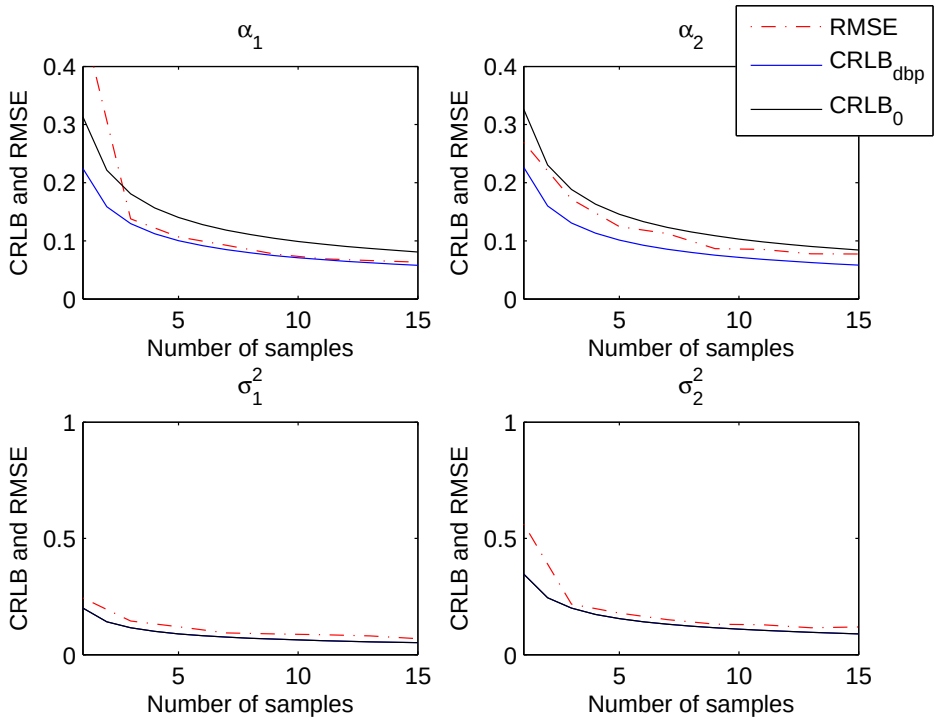


Figure 3.3: Simulation results corresponding to Experiment 2. Lower bounds of α_1, σ_1^2, α_2, σ_2^2, computed when the samples are increased per interval distance.

and number of samples per every interval distance are greater than one, the results are promising.

The output of the Gibbs sampler implemented for the path loss parameters estimation is shown in Figures 3.6 and 3.7. The marginal distributions for channel parameters and d_{bp} estimation are plotted in Figure 3.8. The simplest procedure to judge when sufficiently many M transitions have been simulated to obtain ergodic values $\hat{\boldsymbol{J}}$ close to the desired expectation is to plot the trajectories $\boldsymbol{\psi}_j^{(t)}$ against iteration number t and judge convergence. This shows how the Gibbs sampler sequentially samples the value of each variable separately, in a component-wise fashion.

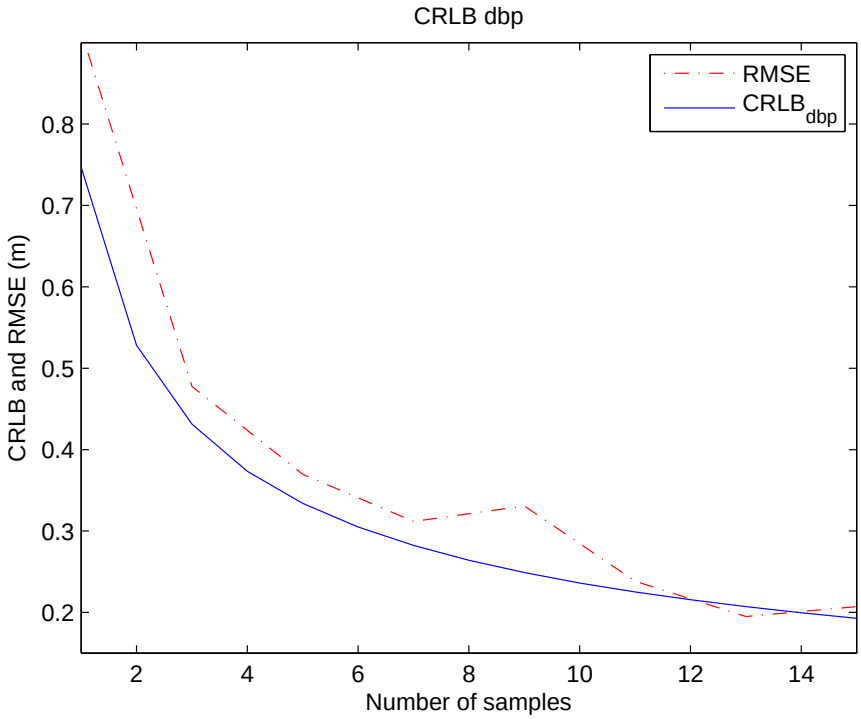


Figure 3.4: Lower bound and RMSE of d_{bp} when the samples are increased in every interval distance.

3.4.2 Validation with Real Data

Thus verification of the simulation requires actual RSS channel measurements. The measurements were obtained in a real office environment LOS scenario. The architectural plan is the second floor of a typical multi-story office building with drywall and Wood Wall panelings reinforced with aluminum bars. These RSS measurements were used for the model calibration. The experiment with real data consisted in taking RSS measurements from the same AP in a LOS condition. The test-bed used to collect the RSS measurements includes a *RaspberryPi* board. The test-bed used for validation with Real data was described in Section 1.5.

The two experiments described in Section 3.4.1 were performed with real data to validate our proposed channel calibration algorithm. In order to mitigate the antennas orientation problem, the antenna's test bed was always faced up

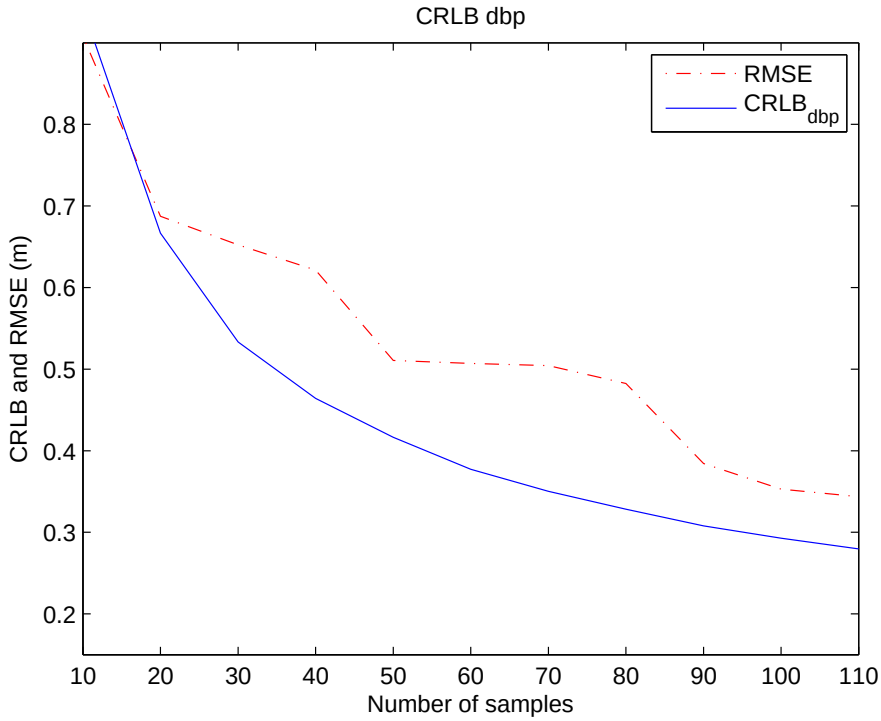


Figure 3.5: Lower bound and RMSE of d_{bp} when RSS measurements are taken when N is decreased in every iteration.

and oriented parallel to the AP. The RSS measurements were collected during evening and on weekends to ensure that the channel is mostly static during the campaign.

Figure 3.9 shows the RSS measurements taken until 13.9 meters in comparative with the path loss model obtained from our Bayesian inference algorithm proposed for a classical model. Figure 3.9 corresponds to the RSS measurementes taken with the second experiment.

For the first experiment, we made a comparative between the classical one-slope model and the two-slope model with the channel model calibration results. The differences between both cases in a LOS condition can be seen in Figure 3.10. As well, the confidence intervals for the classical model and for the two-slope model are plotted also. From the Figures 3.9 and 3.10, it is notable that for large distances, the model calibration algorithm for a two-slope model has a best channel parameters estimation in comparative with the classical model and the estimated parameter values are detailed also in Table 3.2.

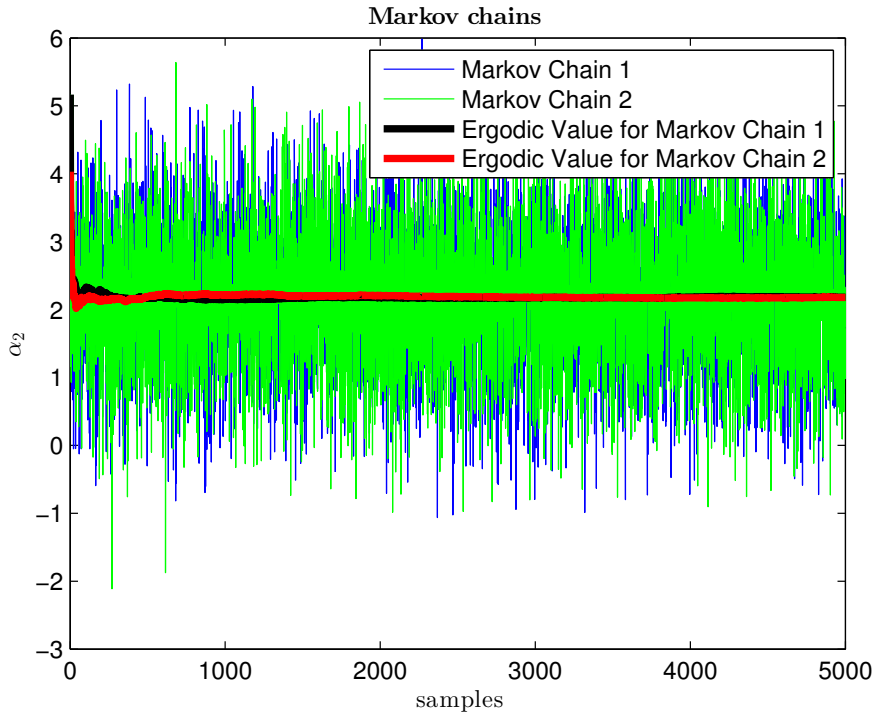


Figure 3.6: Markov chains and samples for bivariate Normal target distributions for α_1.

3.5 Conclusions

In this thesis, a Bayesian approach for estimating the breakpoint distance, and the two-slope path loss model parameters in corrupted RSS measurements is

	Two-Slope	One Slope
$\hat{\alpha}_1$	1.9172	2.6755
$\hat{\alpha}_2$	4.8506	-
$\hat{\sigma}_1$	2.3328	2.364
$\hat{\sigma}_2$	4.1013	-
$\hat{d}_{bp}$	6.2036	-

Table 3.2: Channel parameters values.

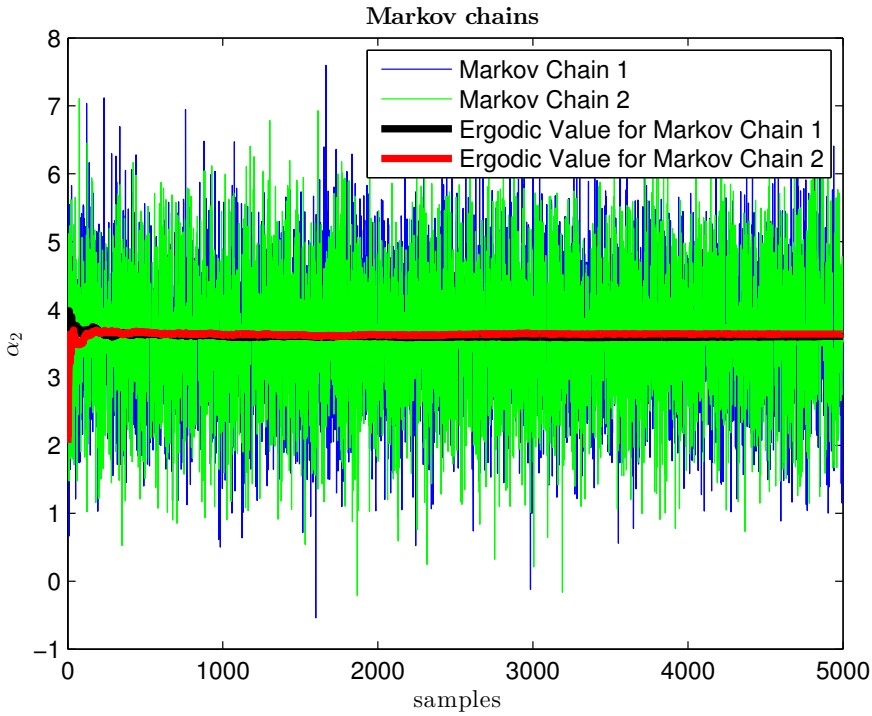


Figure 3.7: Markov chains and samples for bivariate Normal target distributions for α_2.

presented. Additionally, this work proposes an implementation based on Gibbs sampling for which all the necessary posterior conditional distributions are shown. The CRLB for the case of unknown d_{bp} is derived. The bound was compared with the evaluated RMSE of the proposed method, which estimates d_{bp}. The comparative was made with CRLB and not with the Bayesian CRLB because the prior is not informative. The results are tight to the bound, thus indicates that the Bayesian method is estimating the five parameters in a consistent manner. Also, the joint estimation of the two-slope model parameters provides better results than individual fitting considering measurements from each of the models. With computational results, it is remarkable that the channel model calibration algorithm for a two-slope model is more suitable in comparative with the classical model. Real data was used to validate our simulation results. Two sets of RSS measurements were taken in a real environment showing that our Bayesian channel calibration algorithm has good accuracy in both cases leading to a future work as a robust RSS indoor localization algorithm.

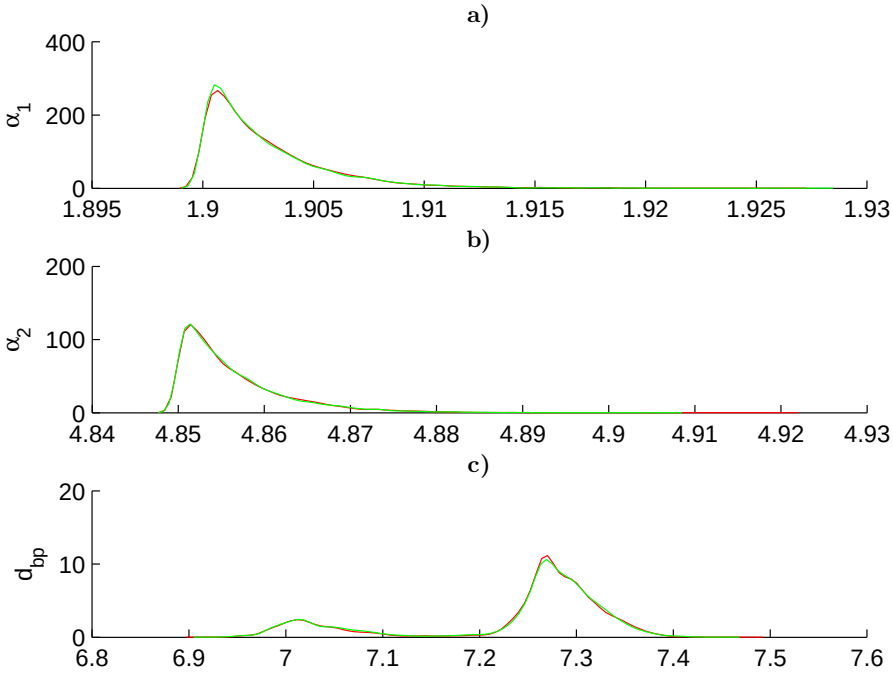


Figure 3.8: Marginal Distributions for channel parameters by Bayesian inference for RSS measurementes in a LOS condition.

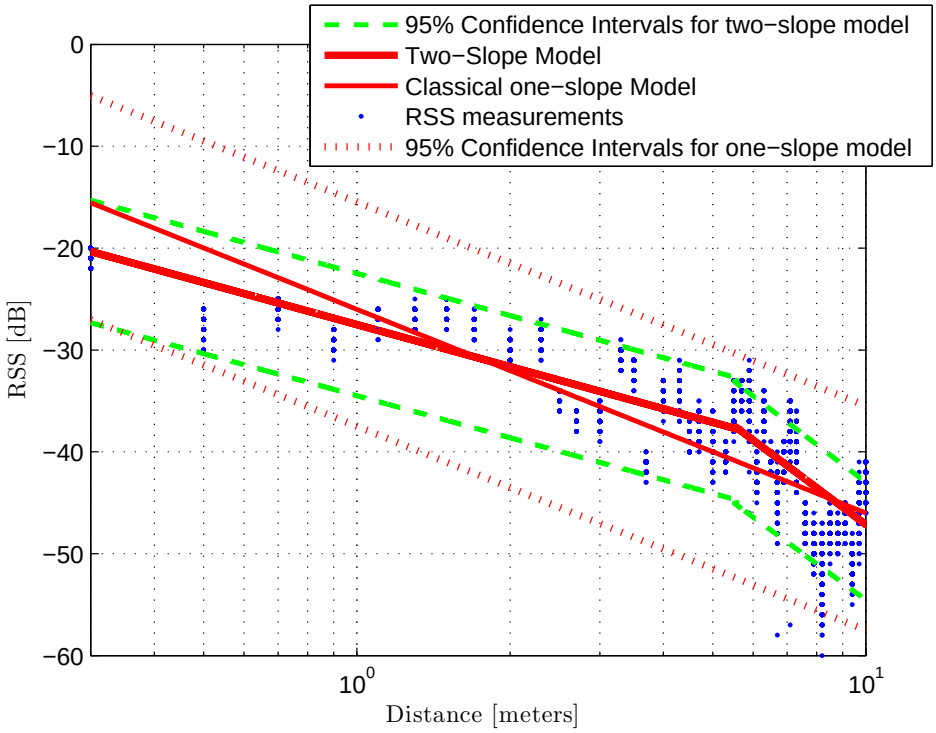


Figure 3.9: Two-slope model with real RSS measurements in a LOS scenario corresponding to the second experiment of Section 3.4.1.

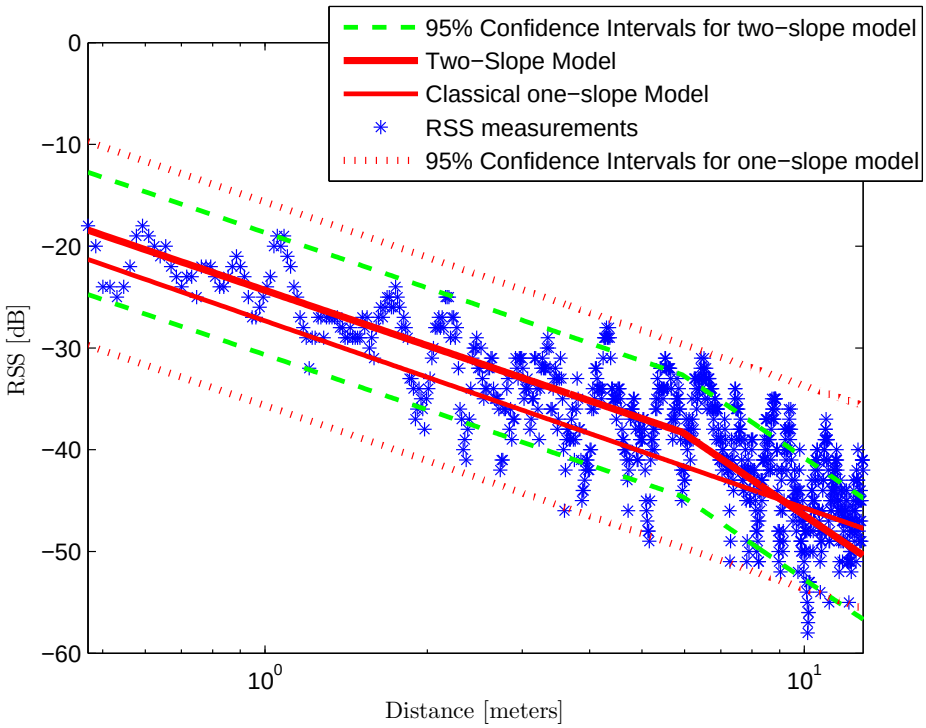


Figure 3.10: Two-slope model with real RSS measurements in a LOS scenario corresponding to the first experiment of Section 3.4.1.

4

Adaptive IMM-based Robust Indoor Localization Algorithm

The main concern of this chapter is to provide a clear answer and the mathematical derivation and solution to the Indoor Localization problem. The first goal is to justify the reasoning behind the use of an IMM-based approach to solve the distance estimation problem from a batch of RSS measurements. Then, to detail the proposed solution to obtain the final position from these intermediate distance estimates. Together with the main architectural core, the model calibration strategies and the corresponding maximum likelihood (ML) estimators for the two-slope model parameters are derived.

A general solution to the robust RSS-based indoor localization problem considering a generalized realistic propagation model is an important missing point in the literature, because the position determination assuming known system parameters and/or simplified system models is not of practical interest. This is the main problem of interest in this Chapter.

Contents

4.1 Overall Robust IMM-based Architecture

Few works have considered the localization problem using RSS measures together with the adaptive estimation of the propagation model parameters [54, 89, 90, 40, 91, 92], and these contributions only take into account simple propagation models.

In this thesis, a generalized two-slope path loss model to overcome the practical limitations of the conventional approach is used. This can be seen as an improvement of the standard localization case. To solve this problem a more sophisticated solution is needed to cope with the different distance-dependent RSS measurement models. No contributions in the literature face this generalized localization problem, thus the solution provided in this thesis can be simplified to optimally cope with it when model parameters are known also.

There are several dynamic systems in many engineering applications that are characterized by a few possibles modes of operation. In this work, there are two dynamics models for the signal model in $h(1)$ and $h(2)$ referred in 3.1. These types of problems are often modeled as a jump Markov jump or hybrid-state estimation problems [111]. This subsection summarizes the overall system and IMM architecture descripted in section 1.3.3.3. A diagram of the interconnection of elements can be consulted in Fig. 4.1.

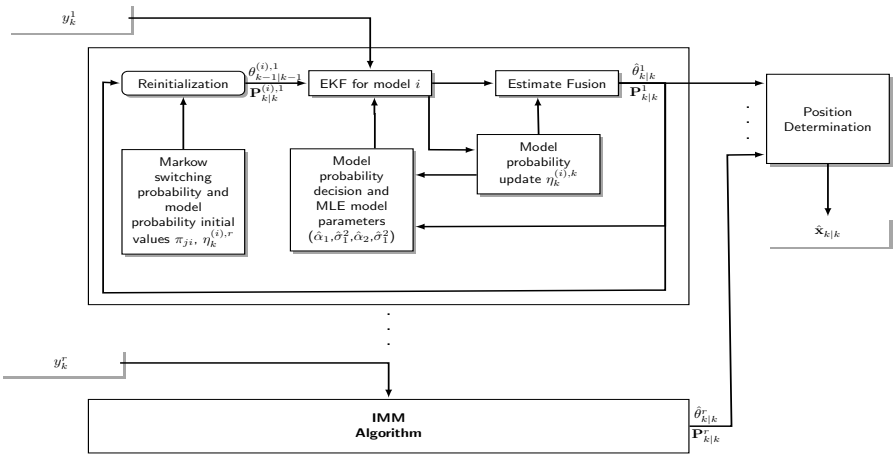


Figure 4.1: Overall system architecture of the proposed IMM algorithm with parameters estimation.

The main goal of the proposed method is to use the proposed EKF-IMM algorithm to sequentially estimate both the two-slope path loss model parameters

and the mobile target position. Notice that in practice, these parameters are typically found after a scene analysis and the posterior linear regression on a semilogarithmic scale [102, 41, 19, 15, 16, 112]. To avoid this off-line site-dependent procedure, the proposed solution performs the on-line parameters estimation within the IMM architecture.

In general, the IMM is capable to deal with model transition, which are modeled as a two-state Markov jump process, being used to filter the distance measurements. The state $\boldsymbol{\theta}_k^r$ is simultaneously estimated by the two EKFs in the IMM according to the corresponding model, which are fused using the model probabilities to obtain the final estimate. The model probabilities resolve wheter the RSS measurement aquired at instant k obeys to (3.2) or (3.3). These probabilities are used to construct the two measurement subsets to obtain the ML path loss parameters estimation, which in turn are feed back to the corresponding EKF. In the proposed architecture, an EKF-IMM is run for every AP. The set of distance estiamtes are used to form the blended positioning solution. The EKF used to obtain the final mobile position was described in 4.3.

4.1.1 Parallel IMM-based solution for distance estimation

The first approach that comes to mind to solve the distance determination problem is the use of a traditional filtering solution, such as the EKF, where the observation accounts for the full set of RSS measurements $\mathbf{z}_k = [z_k^1, ..., z_k^N]^\top$ and the global state evolution takes into account the N individual states, $\boldsymbol{\theta}_k = [\boldsymbol{\theta}_k^1; ...; \boldsymbol{\theta}_k^N]$. But it is straightforward to see from Equation 3.1 that this is not a valid approach, because the measurement model is distance dependent. The two-slope model can be used to effectively model the distance between the AP and the mobile node but both implicit models must be treated separately. The natural solution to overcome this model-switching problem is to use an IMM-based approach.

The key idea behind the IMM is to use a bank of L KFs, each one designed to cope with a specific model (or model set), and to obtain the state estimation as a clever combination of the individual estimates. If the full set of N independent observations y_k is considered, the question that arises is how many KFs should be considered into the IMM as each independent observation may obey $\mathcal{M}_1$ or $\mathcal{M}_2$, the answer is 2^N filters (i.e., all the possible combinations of $\mathcal{M}_1$ and $\mathcal{M}_2$ for the N observations). It is clear that this is not a practical solution for an arbitrary number of APs, therefore, again a divide-and-conquer strategy treating independent measurements separately is the best solution. In this

contribution a parallel IMM-based approach is adopted, considering N IMMs each one involving 2 KFs according to the two path loss models. The block diagram of a single IMM is sketched in Figure 4.2.

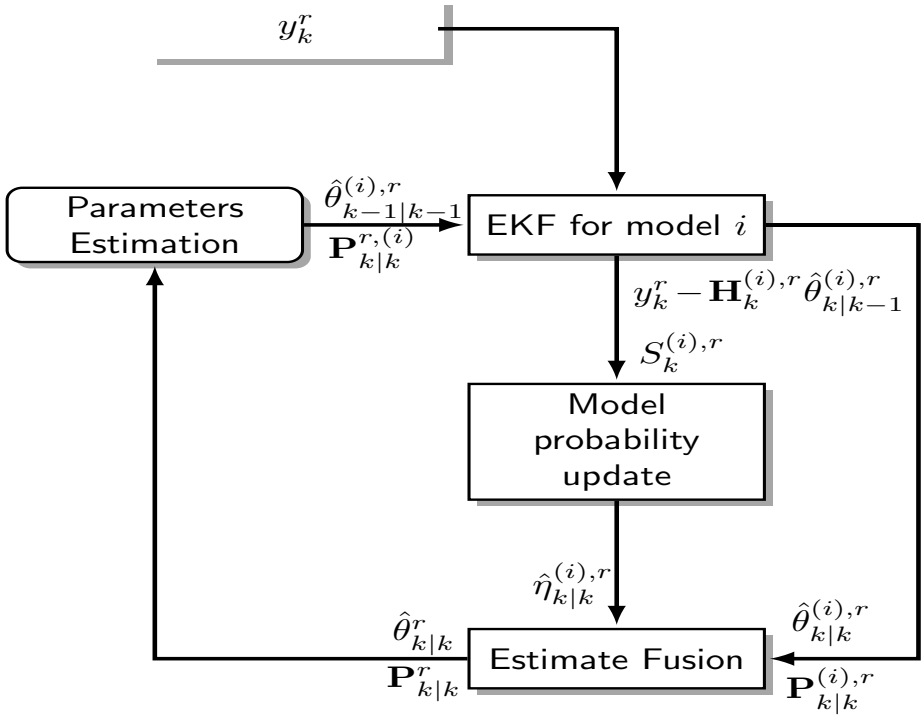


Figure 4.2: IMM Architecture Algorithm at instant k.

At each discrete-time instant k, the IMM algorithm follows a clear three step architecture :

i *Reinitialization*, which represents the interaction between filters;

ii *KF*, being the individual filtering solutions;

iii *Model probability*, which computes the model likelihood to decide if the input measurement comes from $\mathcal{M}_1$ or $\mathcal{M}_2$.

The final estimates are obtained as a weighted combination of the individual KF outputs using the corresponding model likelihoods. Mathematically, one cycle of the standard IMM associated to the r^{th} RSS measurement is sketched in Algorithm 7, where π_{ji} (for $i, j = 1, 2$) is a two-state Markov model transition probability matrix to describe the switching system. The Markov chain is shown in Figure 4.3.

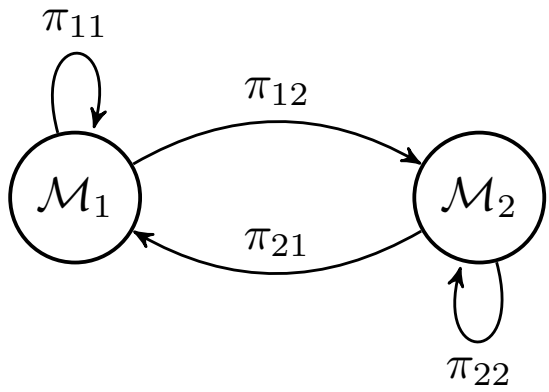


Figure 4.3: Two-state Markov switching model.

Notice that in its standard form, both the two-slope model parameters and the process noise variance, gathered in vector $\boldsymbol{\psi}^{(1)}$, must be specified in the IMM. These parameters must be set to the true ones for an optimal solution. Moreover, the initialization of both EKFs and each AP, $\left\{\hat{\boldsymbol{\theta}}_{0|0}^{(i),r}, \mathbf{P}_{0|0}^{(i),r} \text{ for } i = 1, 2\right\}$, must be set according to the problem at hand. In the sequel an example of a possible parameterization (i.e., these values are the ones used later in the simulations).

- The adaptive estimation of the two-slope model parameters to obtain a robust filtering solution is provided in Sections 4.2.1, together with the discussion on how to set σ_d^2 and d_{bp}.

- The error covariance matrix has an initial value assigned as $\mathbf{P}_{0|0}^{(i),r} = \mathbf{Q}_k$ for each AP. The initial value state vector for the filter is $\hat{\boldsymbol{\theta}}_{0|0}^{(i),r} = \boldsymbol{\theta}_0^r + \boldsymbol{\omega}$. where $\boldsymbol{\omega} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$.

- The two-state Markov transition probability matrix of model switching in the proposed algorithm is for each AP [5],

$$\pi_{ji} = \begin{bmatrix} 0.995 & 0.005 \\ 0.005 & 0.995 \end{bmatrix} \tag{4.1}$$

  .

- The initial model probabilities are set to $\eta_1^{(1),r} = \eta_1^{(2),r} = 0.5$ for every AP.

4.1.2 State-space formulation: RSS → distance

As previously stated, the proposed strategy to solve the localization problem uses a two-step approach. In the first step (i.e., distance estimation) and for the r^{th} AP, the observations correspond to the RSS measurements and the unknown states to be sequentially inferred are

$$\boldsymbol{\theta}_k^r = \begin{bmatrix} d_k^r & \dot{d}_k^r \end{bmatrix}^\top, \tag{4.2}$$

where d_k^r is the distance between the mobile and the r_{th} AP and $\dot{d}_k^r$ is the rate of change of this distance. The linear evolution of states is

$$\boldsymbol{\theta}_k^r = \mathbf{A}_\theta \boldsymbol{\theta}_{k-1}^r + v_k^r \tag{4.3}$$

where v_k^r is the process noise accounting for possible modeling mismatches, such as a possible acceleration of the mobile. In other words, this noise term gathers different forces that could affect target's dynamics and which are not explicitly modeled. The process noise is normally distributed with zero mean and covariance matrix [88, 54]:

$$\mathbf{Q}_k = \sigma_d^2 \mathbf{B}_\theta \mathbf{B}_\theta^\top \tag{4.4}$$

where σ_d^2 models the uncertainty on the mobile dynamics.The state equation includes these matrices:

$$\mathbf{A}_\theta = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \quad ; \quad \mathbf{B}_\theta = \begin{bmatrix} \frac{\Delta t^2}{2} \\ \Delta t \end{bmatrix}, \tag{4.5}$$

where Δt is the sampling period.

To complete the state-space representation, the observation vector is defined. In this case, the RSS measurements per AP are precisely the observations used to infer $\boldsymbol{\theta}_k^r$, and thus

$$y_k^r \triangleq \mathrm{RSS}^r(d_k) = h(d_k) + n_k \tag{4.6}$$

where we recall that the model for $\mathrm{RSS}^r(d)$ depends on the breakpoint distance. Therefore, $h(\,\cdot\,)$ has to be selected according to Equation 3.1 and variance of the measurement noise n_k is σ_1^2 or σ_2^2. To conclude, the state-space formulation for the pair $\{RSS, d\}$ and one single AP is given by $\boldsymbol{\theta}_k^r$ and y_k^r.

If the state estimation or filtering problem taking into account this state-space formulation is to be solved using an Extended Kalman filter (EKF) solution, the following Jacobian matrices of the measurement functions are needed, because $h^{(1)}$ and $h^{(2)}$ are nonlinear. Where the corresponding 2×1 Jacobian matrices are

$$\mathbf{H}_k^{(1)} = \begin{bmatrix} \frac{\alpha_1}{\log 10} \frac{10}{d} & 0 \end{bmatrix}; \tag{4.7}$$

$$\mathbf{H}_k^{(2)} = \begin{bmatrix} \frac{\alpha_2}{\log 10} \frac{10}{d} & 0 \end{bmatrix}. \tag{4.8}$$

Notice that the dependence on the breakpoint distance disappeared from the second measurement equation.

Algorithm 6 Step k of the IMM for the r^{th} AP

1: For $i = \{1, 2\}$ and $j = \{1, 2\}$

2: **Reinitialization**:

Calculation of the predicted mode probability, mixing weights, mixing estimates and mixing covariances, respectively,

Predicted mode probability:
$$\eta_{k|k-1}^{(i),r} = \sum_j \pi_{ji} \eta_{k-1}^{(j),r} \tag{4.9}$$

Mixing weight:
$$\eta_{k-1}^{j|i,r} = \frac{\pi_{ji} \eta_{k-1}^{(j),r}}{\eta_{k|k-1}^{(i),r}} \tag{4.10}$$

Mixing Estimate:
$$\bar{\boldsymbol{\theta}}_{k-1|k-1}^{(i),r} = \sum_j \hat{\boldsymbol{\theta}}_{k-1|k-1}^{(j),r} \eta_{k-1}^{j|i,r} \tag{4.11}$$

Mixing Covariance:
$$\bar{\mathbf{P}}_{k-1|k-1}^{(i),r} = \sum_j [\mathbf{P}_{k-1|k-1}^{(j),r} + (\bar{\boldsymbol{\theta}}_{k-1|k-1}^{(i),r} - \hat{\boldsymbol{\theta}}_{k-1|k-1}^{(j),r}) \times (\bar{\boldsymbol{\theta}}_{k-1|k-1}^{(i),r} - \hat{\boldsymbol{\theta}}_{k-1|k-1}^{(j),r})^\top] \eta_{k-1}^{j|i,r} \tag{4.12}$$

3: **Model-conditioned EKF**:

Prediction, innovations' covariance matrix, Kalman gain, state estimate and the corresponding error covariance matrix, are given by

Predicted State:
$$\hat{\boldsymbol{\theta}}_{k|k-1}^{(i),r} = \mathbf{A}_\theta \bar{\boldsymbol{\theta}}_{k-1|k-1}^{(i),r} \tag{4.13}$$

Predicted Covariance:
$$\mathbf{P}_{k|k-1}^{(i)} = \mathbf{A}_\theta \bar{\mathbf{P}}_{k-1|k-1}^{(i),r} \mathbf{A}_\theta^\top + \mathbf{Q}_k^{(i)} \tag{4.14}$$

Residual Covariance:
$$S_k^{(i),r} = \mathbf{H}_k^{(i),r} \mathbf{P}_{k|k-1}^{(i),r} (\mathbf{H}_k^{(i),r})^\top + \sigma_i^2 \tag{4.15}$$

Filter Gain:
$$\mathbf{K}_k^{(i),r} = \mathbf{P}_{k|k-1}^{(i),r} (\mathbf{H}_k^{(i),r})^\top (S_k^{(i),r})^{-1} \tag{4.16}$$

Updated State:
$$\hat{\boldsymbol{\theta}}_{k|k}^{(i),r} = \hat{\boldsymbol{\theta}}_{k|k-1}^{(i),r} + \mathbf{K}_k^{(i),r} (y_k^r - \mathbf{H}_k^{(i),r} \hat{\boldsymbol{\theta}}_{k|k-1}^{(i),r}) \tag{4.17}$$

$$\mathbf{P}_{k|k}^{(i),r} = \mathbf{P}_{k|k-1}^{(i),r} - \mathbf{K}_k^{(i),r} S_k^{(i),r} (\mathbf{K}_k^{(i),r})^\top \tag{4.18}$$

Algorithm 7 Step k of the IMM for the r^{th} AP continued...

4: **Model probability update**:

The model likelihood function and model probability are respectively

$$\text{Model Likelihood:} \quad LF_k^{(i),r} = \mathcal{N}(y_k^r - \mathbf{H}_k^{(i),r}\hat{\boldsymbol{\theta}}_{k|k-1}^{(i),r}; 0, S_k^{(i),r}) \tag{4.19}$$

$$\text{Mode probability:} \quad \eta_k^{(i),r} = \frac{\eta_{k|k-1}^{(i),r} LF_k^{(i),r}}{\sum_j \eta_{k|k-1}^{(j),r} LF_k^{(j),r}} \tag{4.20}$$

5: **Estimate fusion**:

$$\hat{\boldsymbol{\theta}}_{k|k}^r = \sum_i \hat{\boldsymbol{\theta}}_{k|k}^{(i),r} \eta_k^{(i),r} \tag{4.21}$$

$$\mathbf{P}_{k|k}^r = \sum_i [\mathbf{P}_{k|k}^{(i),r} + (\hat{\boldsymbol{\theta}}_{k|k}^r - \hat{\boldsymbol{\theta}}_{k|k}^{(i),r})(\hat{\boldsymbol{\theta}}_{k|k}^r - \hat{\boldsymbol{\theta}}_{k|k}^{(i),r})^\top] \eta_k^{(i),r} \tag{4.22}$$

4.2 Online Model Calibration

In the previous section, the overall system model has been fully described, providing the mathematical formulation needed to come up with a new powerful and robust indoor localization solution. Notice that the first state-space model is parameterized and completely determined by $\boldsymbol{\psi}^{(1)} = [\sigma_d^2, \boldsymbol{\psi}_d]^\top = [\sigma_d^2, \alpha_1, \alpha_2, \sigma_1^2, \sigma_2^2, d_{bp}]^\top$, and the second one, by $\boldsymbol{\psi}^{(2)} = [\sigma_p^2, \mathbf{R}_{p,k}]^\top$, where the positions of the N APs are assumed to be known. To understand the problem at hand and to clearly place the new contribution of this work with respect to state-of-the-art RSS-based localization solutions, the main estimation problems within the indoor localization framework are summarized in the sequel.

- *Generalized localization problem*: in this contribution, the term "generalized" refers to the use of a generalized two-slope path loss model (still considering perfectly known model parameters) to overcome the practical limitations of the conventional approach. This can be seen as an improvement of the standard localization case. To solve this problem a more sophisticated solution is needed to cope with the different distance-dependent RSS measurement models. From the best of our knowledge, no contributions in the literature face this generalized localization problem, thus the solution provided in this work can be simplified to optimally

cope with it when model parameters are known.

- *Robust localization problem*: in practice, the system parameters are not perfectly known, what leads to poor localization performances when using the previous solution in real-life applications. The robust or adaptive localization problem implies the sequential position determination from a set of RSS measurements and the simultaneous model calibration or system model parameters estimation. Two cases can be considered:
  - *Case I: standard model*. Few works have considered the localization problem using RSS measures together with the adaptive estimation of the propagation model parameters [54, 89, 90, 40, 91, 92], and these contributions only take into account simple propagation models.
  - *Case II: generalized two-slope model*. A general solution to the robust RSS-based indoor localization problem considering a generalized realistic propagation model is an important missing point in the literature, because the position determination assuming known system parameters and/or simplified system models is not of practical interest. This is the main problem of interest in this contribution.

This paper proposes a solution to the generalized robust indoor localization problem, considering a two-slope RSS propagation model and the estimation of the path loss exponents and shadowing of the model. Mathematically, this is expressed as the sequential estimation of $\mathbf{x}_k = [x_k, y_k, \dot{x}_k, \dot{y}_k]^\top$ using an intermediate estimate of $\boldsymbol{\theta}_k^r = [d_k^r \ \dot{d}_k^r]^\top$ for r from 1 to N, and the simultaneous estimation of $[\alpha_1, \alpha_2, \sigma_1^2, \sigma_2^2]^\top$ as a model calibration strategy. Why the other parameters, $[\sigma_d^2, \sigma_p^2, \mathbf{R}_{p,k}, d_{bp}]^\top$, are not estimated and how they are determined will be discussed in the following sections.

Taking into account the robust filtering problem at hand, the general system model formulation and the two-step estimation approach, the following points are sequentially treated in the sequel: *i*) distance estimation from a set of RSS measures (Section 4.1.1), *ii*) position estimation from the intermediate distance estimates (Section 4.3) and *iii*) model calibration and parameter estimation strategies (Section 4.2.1). To close the loop, the overall discussion on the complete robust localization solution is given in Section 4.1.

4.2.1 Maximum Likelihood Two-Slope Model Calibration

In the previous paragraphs, the proposed two-step state estimation solution has been derived, first providing an intermediate distance estimate using a bank of N IMMs to cope with the N APs and the two-slope path model; and

then, an EKF-based solution to obtain the final position from these estimates. Both state-space models are determined by a set of parameters, $\boldsymbol{\psi}^{(1)}$ and $\boldsymbol{\psi}^{(2)}$, respectively, which in general may be unknown to a certain extent. It has already been justified that the model parameters must be somehow adjusted or estimated to obtain a robust and flexible solution. The second state-space model parameters' specification, $\boldsymbol{\psi}^{(2)}$, has already been detailed in Section 4.3, and thus hereafter the focus is on the sequential estimation of $\boldsymbol{\psi}^{(1)}$.

This section presents the estimation strategies for the two-slope model calibration together with the theoretical derivation of the ML parameters estimators for $[\alpha_1, \alpha_2, \sigma_1^2, \sigma_2^2]^\top$. Recall that the first state-space model has two extra parameters, namely, the process noise variance σ_d^2 and the breakpoint distance d_{bp}, which are not estimated but rather set as detailed in the sequel:

- The process noise variance for the distance estimation problem, σ_d^2, is determined assuming that the mobile target has an average velocity of 1 m/s in the simulations presented in this work, so a small initial value for σ_d^2 of 0.7 m/s^2 is chosen. The same reasoning applies for σ_p^2.
- The breakpoint distance indicates the change of models in the path loss model. An off-line Bayesian approach for the d_{bp} estimation was considered in [112], but its applicability to the on-line configuration of interest here is still under investigation. In this work, the IEEE standard [41] channel parameters are considered to set the value of $d_{bp} = 5$ meters. For the sake of completeness, in the computer simulations the impact of under- and over-estimation of this value is presented.

In the two-slope model (Equation 3.1), the RSS measurements may come from the first equation modeling the propagation for close distances (called $\mathcal{M}_1$), or alternatively, they may obey the second equation modeling the propagation for distances beyond the breakpoint distance (called $\mathcal{M}_2$). Therefore, at the input of the system, there exists a model uncertainty which must be resolved.

In the proposed methodology, each IMM inherently treats this model uncertainty by computing the model likelihood from the innovations of each KF. For each AP r and model i, the model probability is given by $\eta_k^{(i),r}$. These probabilities are used into the filter to weight the outputs of the individual KFs but can also be reused for the model calibration. At each time step and using these model probabilities, two subsets of RSS measurements are constructed: if $\eta_k^{(1),r} > \eta_k^{(2),r}$, the RSS measurement y_k^r is associated to $y_{1,k}^r$ (i.e., which represents the RSS measurements subset obeying $\mathcal{M}_1$), $\mathcal{Y}_{1,k}^r = \{y_k^r : \eta_k^{(1),r} > \eta_k^{(2),r}\}$, otherwise, it is associated to $\mathcal{Y}_{2,k}^r$ (i.e., which concatenates the RSS measurements obeying $\mathcal{M}_2$). Therefore, we take a hard decision to associate RSS measurements to each of the models and construct the maximum likelihood estimator (MLE) accordingly. Note that the cardinality of these sets is upper

bounded by the present time instant, $U_i \triangleq |\mathcal{Y}^r_{i,k}| < k,\ i = \{1, 2\}$, and that their sum is precisely $|\mathcal{Y}^r_{1,k}| + |\mathcal{Y}^r_{2,k}| = k$. For the sake of clarity in the forthcoming derivations, let us define the elements in the sets as $\mathcal{Y}^r_{1,k} = \{y^r_{1,1}, \ldots, y^r_{1,U_1}\}$ and $\mathcal{Y}^r_{2,k} = \{y^r_{2,1}, \ldots, y^r_{2,U_2}\}$.

4.2.1.1 ML estimators for $\mathcal{M}_1$

In the following, the ML estimators for the close distance ($d \leq d_{bp}$) model parameters are provided. Notice that the parameters to be estimated in this case are $\left\{\alpha_1, \sigma_1^2\right\}$ at each AP. As the parameter α_1 appears in both models, $\hat{\alpha}_{1,1}$ is used for the close distances and $\hat{\alpha}_{1,2}$ for distances beyond d_{bp}, both being estimators of α_1.

The expressions for the estimators are herein provided, the reader is referred to Appendix C for the complete derivation. For the r^{th} AP, the ℓ^{th} sample of the first model subset $\mathcal{Y}^r_{1,k}$ at time k is Gaussian distributed, $y^r_{1,\ell} \sim \mathcal{N}(\bar{y}^r_{1,\ell}, \sigma_1^{2,r})$, with $\bar{y}^r_{1,\ell} = L_0 + 10\alpha^r_{1,1} \log_{10} d^r_\ell$.

Using this subset and assuming a known distance to the r^{th} AP, d^r_ℓ, at instant k the ML estimators are given by

$$\hat{\sigma}_1^{2,r} = \frac{1}{U_1} \sum_{\ell=1}^{U_1} \left(y^r_{1,\ell} - \bar{y}^r_{1,\ell}\right)^2, \tag{4.23}$$

$$\hat{\alpha}^r_{1,1} = \frac{\sum_{\ell=1}^{U_1} \left[(y^r_{1,\ell} - L_0)(\log d^r_\ell)\right]}{10 \sum_{\ell=1}^{U_1} (\log d^r_\ell)^2}. \tag{4.24}$$

4.2.1.2 ML estimators for $\mathcal{M}_2$

Following the same procedure but using the second model subset of RSS measurements, the ML estimators of $[\alpha_{1,2}, \alpha_2, \sigma_2^2]$ are given in the sequel for the r^{th} AP. Refer to Appendix D for the complete derivation.

For the r^{th} AP, the ℓ^{th} sample of the second model subset $\mathcal{Y}^r_{2,k}$ at time k is Gaussian distributed as well, $y^r_{2,\ell} \sim \mathcal{N}(\bar{y}^r_{2,\ell}, \sigma_2^{2,r})$, with $\bar{y}^r_{2,\ell} = L_0 + 10\alpha^r_{1,2} \log_{10} d_{bp} + 10\alpha^r_2 \log_{10}\left(\frac{d^r_\ell}{d_{bp}}\right)$.

Using this subset and assuming a known distance to the r^{th} AP, d_ℓ^r, then at time instant k the ML estimator of σ_2^2 is given by

$$\hat{\sigma}_2^2 = \frac{1}{U_2} \sum_{\ell=1}^{U_2} \left(y_{2,\ell}^r - \bar{y}_{2,\ell}^r \right)^2, \tag{4.25}$$

and the ML estimators $\hat{\alpha}_2^r$ and $\hat{\alpha}_{1,2}^r$ are the first and second elements of vector $\hat{\boldsymbol{\alpha}}$ in (D.2).

For the MLEs of both models there are two issues to account for. Firstly, notice that in practice the true distance to the r^{th} AP is unknown, and thus an estimate $\hat{d}_\ell^r$ must be used instead, which is available at the output of the corresponding IMM.

Secondly, two ML estimates exist for α_1^r, but the algorithm needs to take a decision and provide a unique solution $\hat{\alpha}_1^r$. In this work, the following simple rule is considered. The idea is to use the estimator that has gathered more observations at time k, also accounting for the actual model probabilities $\eta_k^{(1),r}$ and $\eta_k^{(2),r}$. More precisely, when $\eta_k^{(1),r} > \eta_k^{(2),r}$ the method is in favor of using $\hat{\alpha}_1^r = \hat{\alpha}_{1,1}^r$ if $U_1 > U_2$ and $\hat{\alpha}_1^r = \hat{\alpha}_{1,2}^r$ otherwise. Similarly, if $\mathcal{M}_2$ is more probable, $\eta_k^{(1),r} < \eta_k^{(2),r}$, then the procedure will select $\hat{\alpha}_1^r = \hat{\alpha}_{1,2}^r$ if $U_1 < U_2$ or keep the last estimated value otherwise. With this procedure we avoid large transients in cases where a target is in $\mathcal{M}_2$ and suddenly enters into $\mathcal{M}_1$, in which case the estimators are indeed starting to operate for this model.

4.3 Mobile Target's position determination

The main goal of this work is to obtain the mobile target position. This issue is solved with an additional EKF using as observations the N distance estimates obtained from the bank of IMMs. Two evaluations were made. In Section 4.3.1 is described the validation with only using RSS measurements. Section 4.3.2 correspond the methodology with using INS measurements.

4.3.1 Position determination with RSS measurements

4.3.1.1 Location calculation: distance → position

The final goal is to sequentially obtain the mobile position. The location calculation is performed using the N distance measures related to the N visible APs. In this case, the state vector gathers the mobile position and velocity, $\mathbf{x}_k = [x_k, y_k, \dot{x}_k, \dot{y}_k]^\top$, and the observation vector is $\mathbf{z}_k = [\hat{d}_k^1, \ldots, \hat{d}_k^N]^\top$, in this case $\hat{d}_k^r$ referring to the noisy distance measurements between the target and the r^{th} AP at time k. The state-space mathematical representation of the problem is given in the sequel.

Process equation the state evolution is given by $\mathbf{x}_k = \mathbf{A}_x\mathbf{x}_{k-1} + w_k$, where the resulting Gaussian process noise w_k has a covariance matrix $\mathbf{Q}_{\boldsymbol{x}_k} = \sigma_p^2\mathbf{B}_x\mathbf{B}_x^\top$, σ_p^2 is the variance related to the mobile acceleration, and

$$\mathbf{A}_x = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \; ; \; \mathbf{B}_x = \begin{bmatrix} \frac{\Delta t}{2} & 0 \\ 0 & \frac{\Delta t}{2} \\ \Delta t & 0 \\ 0 & \Delta t \end{bmatrix}. \tag{4.26}$$

Observation equation the relation between distance and position is defined as $\mathbf{z}_k = \mathbf{g}_k(\mathbf{x}_k) + \boldsymbol{\nu}_k$, where $\mathbf{z}_k \in \mathbb{R}^N$ is a vector gathering the estimated distances to the N APs, the observation error $\boldsymbol{\nu}_k$ is modeled as an uncorrelated white Gaussian noise with covariance $\mathbf{R}_{p,k}$, and the nonlinear observation function, $\mathbf{g}_k(\mathbf{x}_k)$, is defined as the distance of the mobile to every anchor point,

$$\mathbf{g}_k(\mathbf{x}_k) = \begin{bmatrix} \sqrt{(x_k - x_p^1)^2 + (y_k - y_p^1)^2} \\ \vdots \\ \sqrt{(x_k - x_p^N)^2 + (y_k - y_p^N)^2} \end{bmatrix} = \begin{bmatrix} d_k^1 \\ \vdots \\ d_k^N \end{bmatrix}, \tag{4.27}$$

where $\{x_p^r, y_p^r\}$ is the position of the r^{th} AP, d_k^r is the true distance between the mobile and the r^{th} AP, and the corresponding Jacobian used to solve the

nonlinear filtering problem into an EKF solution is given by

$$\mathbf{G}_k(\mathbf{x}_k) = \begin{bmatrix} \frac{x_k - x_p^1}{d_k^1} & \frac{y_k - y_p^1}{d_k^1} & 0 & 0 \\ \vdots & & & \vdots \\ \frac{x_k - x_p^N}{d_k^N} & \frac{y_k - y_p^N}{d_k^N} & 0 & 0 \end{bmatrix}. \tag{4.28}$$

4.3.1.2 Position Estimation Using Time-Varying Kalman Filter

At each time step k, the output of the r^{th} IMM filter is an estimate of the distance (and distance rate) to the r^{th} AP, $\hat{\boldsymbol{\theta}}_{k|k}^r$. Therefore, the output of the whole block of parallel IMMs is an estimated distance vector, $\mathbf{z}_k = [\hat{d}_k^1, \ldots, \hat{d}_k^N]^\top$, where $\hat{d}_k^r$ is the first element in $\hat{\boldsymbol{\theta}}_{k|k}^r$. Using the state-space formulation given by $\mathbf{x}_k$ and $\mathbf{z}_k$, and taking the estimated distance vector as input measurements, the position determination can be solved straightforwardly with an EKF.

The EKF-based position estimation is sketched in Algorithm 8. Notice that the measurement Jacobian matrix in (4.36) is evaluated at the predicted state to obtain a linear formulation, thus, $\mathbf{G}_k = \mathbf{G}_k(\mathbf{x}_k)|_{\mathbf{x}_k = \hat{\mathbf{x}}_{k|k-1}}$. Regarding the implementation of such solution, both the initialization, $\left\{\hat{\mathbf{x}}_{0|0}, \mathbf{P}_{x,0|0}\right\}$, and the noise statistics parameters, $\left\{\sigma_p^2, \mathbf{R}_{p,k}\right\}$, must be specified:

- The measurement noise covariance can be set according to the output of the individual IMM filters. The measurement noise accounts for possible errors or observation noise, but the set of measurements used in the position determination is the set of estimated distances. The error on the estimation of the distance within the IMM is given by the estimation error covariance matrix, which for each AP is $\mathbf{P}_{k|k}^r$. Therefore, at each time step, the measurement error covariance is set according to $\mathbf{R}_{p,k} = \text{diag}(\mathbf{P}_{k|k}^1(1,1), ..., \mathbf{P}_{k|k}^N(1,1))$.

- The error covariance matrix has an initial value assigned as $\mathbf{P}_{x,0|0} = 4\sigma_p^2 \mathbf{B}_x \mathbf{B}_x^\top$ for each AP. The initial value of σ_p^2 is indicated in Section 4.2.1.

- The initial value state vector for the filter is $\hat{\boldsymbol{x}}_{0|0} = \boldsymbol{x}_0 + \boldsymbol{\omega}$, with $\boldsymbol{\omega} \sim \mathcal{N}(\mathbf{0}, 0.8\boldsymbol{I})$.

Algorithm 8 EKF formulation for position determination

Require: $\hat{\mathbf{x}}_{0|0}$, $\mathbf{P}_{x,0|0}$, σ_p^2 and $\mathbf{R}_{p,k}$, $\forall\ k$

1: Set $k \Leftarrow 1$

Time update (prediction)

2: Estimate the predicted state:

3: $\hat{\mathbf{x}}_{k|k-1} = \mathbf{A}_x \hat{\mathbf{x}}_{k-1|k-1}$.

4: Estimate the predicted error covariance:

5: $\mathbf{P}_{x,k|k-1} = \mathbf{A}_x \mathbf{P}_{x,k-1|k-1} \mathbf{A}_x^\top + \sigma_p^2 \mathbf{B}_x \mathbf{B}_x^\top$.

Measurement update (estimation)

6: Estimate the predicted measurement:

7: $\hat{\mathbf{z}}_{k|k-1} = \mathbf{g}_k\left(\hat{\mathbf{x}}_{k|k-1}\right)$.

8: Estimate the innovation covariance matrix:

9: $\mathbf{P}_{y,k|k-1} = \mathbf{G}_k \mathbf{P}_{x,k|k-1} \mathbf{G}_k^\top + \mathbf{R}_{p,k}$.

10: Estimate the Kalman gain

11: $\tilde{\mathbf{K}}_k = \mathbf{P}_{x,k|k-1} \mathbf{G}_k^\top \mathbf{P}_{y,k|k-1}^{-1}$.

12: Estimate the updated state

13: $\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \tilde{\mathbf{K}}_k\left(\mathbf{z}_k - \hat{\mathbf{z}}_{k|k-1}\right)$.

14: Estimate the corresponding error covariance:

15: $\mathbf{P}_{x,k|k} = \mathbf{P}_{x,k|k-1} - \tilde{\mathbf{K}}_k \mathbf{G}_k \mathbf{P}_{x,k|k-1}$.

16: Set $k \Leftarrow k+1$ and go to step 2.

4.3.2 Position determination with RSS/INS measurements

The state vector gathers the mobile target position and velocity, as well as the accelerometer bias vector,

$$\mathbf{x}_{ins} \triangleq \begin{bmatrix} x_k & y_k & z_k & \dot{x}_k & \dot{y}_k & \dot{z}_k & \delta a_{x,k} & \delta a_{y,k} & \delta a_{z,k} \end{bmatrix}^\top . \tag{4.29}$$

The observation vector is defined as the output of each IMM:

$$\mathbf{z}_k = \begin{bmatrix} \hat{d}_k^1 & \cdots & \hat{d}_k^N \end{bmatrix}^\top , \tag{4.30}$$

and the observation equation is:

$$\mathbf{z}_k = \mathbf{g}_k(\mathbf{x}_k) + \nu_k \tag{4.31}$$

where $\hat{d}_k^r$ is the first element in $\hat{\boldsymbol{\theta}}_{k|k}^r$ and $\mathbf{g}_k(\mathbf{x}_k)$ is a function that computes the Euclidean distance between the target at $[x_k, y_k, z_k]^T$ and each AP. The state equation is:

$$\mathbf{x}_{ins} = \mathbf{A}_{ins} \mathbf{x}_{k-1} + \mathbf{B}_{ins} \mathbf{a}_{k-1} + \mathbf{D}_x \omega_x. \tag{4.32}$$

ω_x is a white, zero mean Gaussian distribution with $\mathbf{Q}_{ins} = \mathbb{E}\left(\omega_x \omega_x^\top\right)$ and

$$\mathbf{A}_{ins} = \begin{bmatrix} \mathbf{I}_{3\times3} & \Delta t\mathbf{I}_{3\times3} & \frac{\Delta t^2}{2}\mathbf{I}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} & \Delta t\mathbf{I}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} \end{bmatrix} \tag{4.33}$$

$$\mathbf{B}_{ins} = \begin{bmatrix} \frac{\Delta t^2}{2}\mathbf{I}_{3\times3} \\ \Delta t\mathbf{I}_{3\times3} \\ \mathbf{0}_{3\times3} \end{bmatrix}; \tag{4.34}$$

$$\mathbf{D}_{ins} = \begin{bmatrix} \frac{\Delta t^3}{6}\mathbf{I}_{3\times3} \\ \frac{\Delta t^2}{2}\mathbf{I}_{3\times3} \\ \Delta t\mathbf{I}_{3\times3} \\ \mathbf{0}_{3\times3} \end{bmatrix}. \tag{4.35}$$

The observation error ν_k is modeled as an uncorrelated white Gaussian noise with covariance $\mathbf{R}_{x,k} = \text{diag}\left(\mathbf{P}^1_{k|k}(1,1), ..., \mathbf{P}^N_{k|k}(1,1)\right)$, being each element the corresponding entry in the filtering covariance matrix for d^r_k.

The EKF-based position estimation is shown in Figure 4.4, where

$$\mathbf{G}_{ins}(\mathbf{x}_k) = \begin{bmatrix} \frac{x_k - x_p^1}{d_k^1} & \frac{y_k - y_p^1}{d_k^1} & \frac{z_k - z_p^1}{d_k^1} & \mathbf{0}_{1\times6} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{x_k - x_p^N}{d_k^N} & \frac{y_k - y_p^N}{d_k^N} & \frac{y_k - z_p^N}{d_k^N} & \mathbf{0}_{1\times6} \end{bmatrix}. \tag{4.36}$$

4.4 Evaluation and Results with RSS measurements

This section evaluates the proposed algorithms (calibration and distance filtering) with numerial and real RSS measurements. In Section4.4.0.1, the results of the IMM-EKF algorithm were obtained with synthetic signal and in Section 4.4.0.2 with Real data.

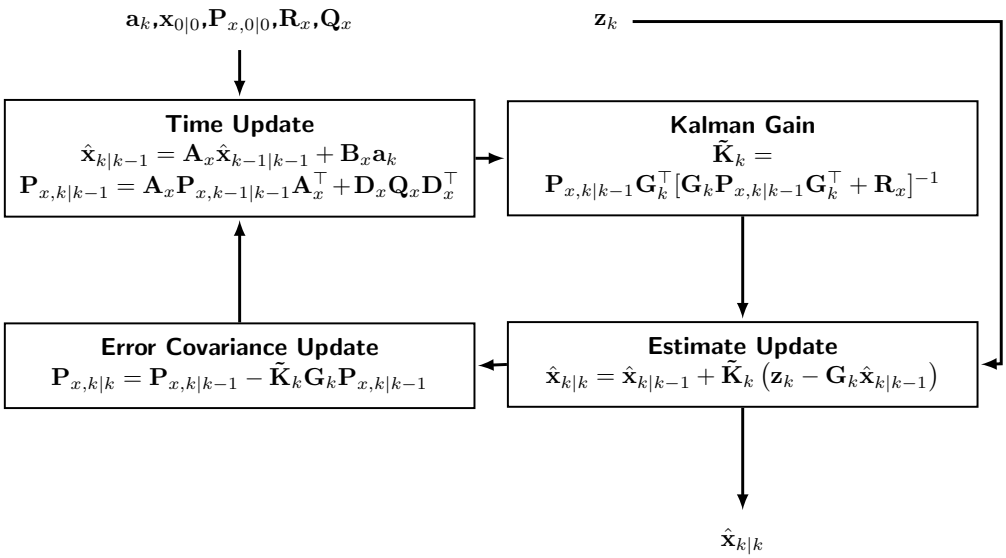


Figure 4.4: EKF algorithm for position determination where $\mathbf{G}_k$ corresponds to $\mathbf{G}_k(\hat{\mathbf{x}}_{k|k-1})$.

4.4.0.1 Simulation results

The method proposed in this work was validated by computer simulations in an scenario depicted in Figure 4.5 that could be considered as a realistic scenario and where the number of APs ($N = 6$) were deployed in a 30×30 m^2 area at known locations. A mobile node was moving in the area, with initial coordinates being at origin, $(0, 0)$, and a steady velocity of 1 m/s corresponding to a pedestrian movement. The duration of the trajectory consisted in 18 seconds with a sample period Δt of 100 ms, emulating realistic WLAN measurements. The trajectory was kept the same (see Fig. 4.5) throughout the simulations in order to see particular cases such as node approaching an AP (e.g., AP 2), leaving the proximity of an AP (e.g., AP 5), and combinations (e.g. AP 6). These 3 APs (e.g. 2, 5, and 6) are those where model switching could take place. The rest of them have a relative distance to the mobile node larger than $d_{bp} = 5$ meters along the track, and thus they are likely to provide observations for $\mathcal{M}_2$. The true model parameters were $\alpha_1 = 2$, $\sigma_1 = 3$, $\alpha_2 = 3.5$, and $\sigma_2 = 5$.

Firstly, a batch of simulations for a single realization of the method are shown which allows us to comment details, as well as to provide some insights and intuition regarding the operation of the proposed method. Secondly, Monte

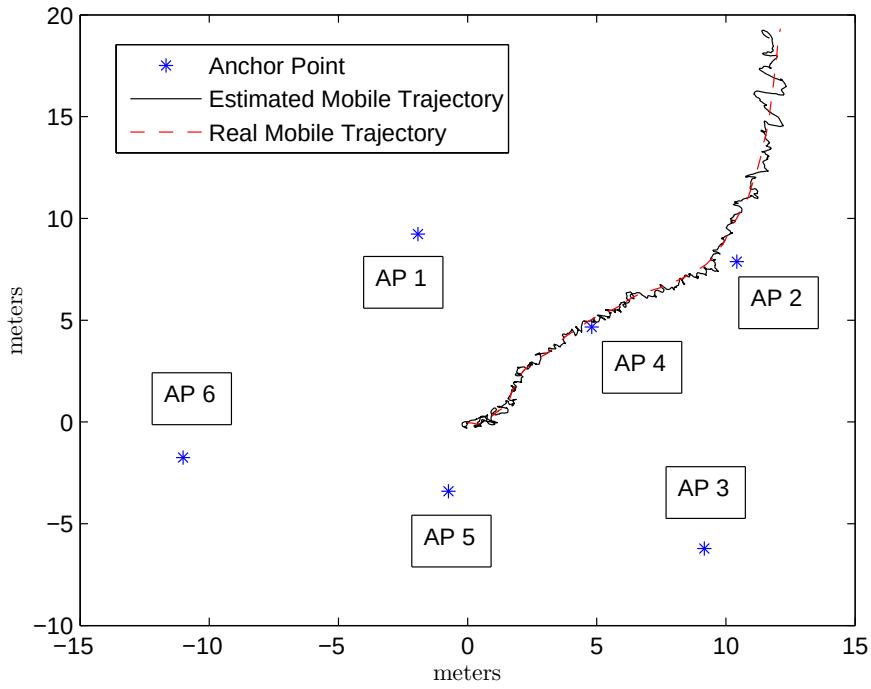


Figure 4.5: Plot of scenario with real and estimated path of the mobile target, for one realization.

Carlo simulations were performed to evaluate the root mean square error (RMSE) performance of our method, compared to other comparative solutions. Namely, we compared our method (termed in the legends as IMM-MLE) with the same IMM architecture able to track the node under the two switching models but with known model parameters, in which case the MLE are not required since the true values were set. Also, we compared the solution to a standard EKF-based algorithm considering that all observations obey $\mathcal{M}_1$, and thus not accounting for the possible changes in the path loss exponent and shadowing variance for relative distances larger than d_{bp}. The latter method assumed perfect knowledge of $\mathcal{M}_1$ parameters.

For a single realization, the distances estimated compared with the real ones with respect to every AP along the simulation duration are shown in Fig. 4.6. For better understanding of these results, the d_{bp} value is plotted also. Notice that in some time instants, the distance between mobile target and some APs changes from values above d_{bp} to lower values and vice versa. This indicates that a switch model has occurred. The IMM structure is able to track the

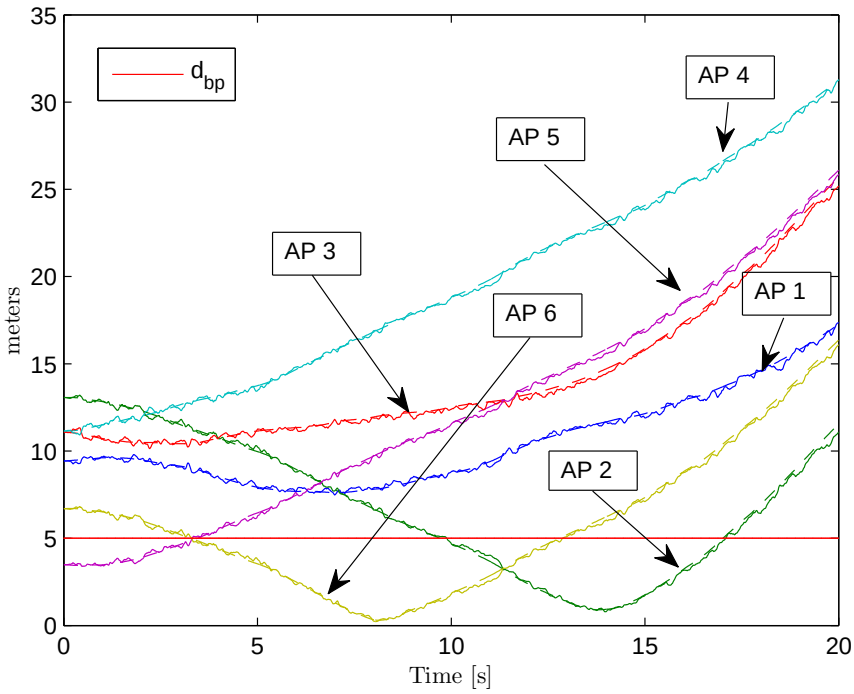


Figure 4.6: Real and estimated distance of the mobile target to every AP.

distances even under model switching.

In the situation were the mobile node is close to the breakpoint distance (that is the border for the two-slope model), the model probabilities $\eta_k^{(1),r}$ and $\eta_k^{(2),r}$ are likely to exhibit nervous behaviors. To illustrate this fact, the estimated distance referred to AP 5 along the simulation is shown in Fig. 4.7. The bottom plot therein shows the performance of the decision process in $\mathcal{M}_1$–$\mathcal{M}_2$ switching. The top plot present the estimated distance.

Figures 4.8 and 4.9 show the MLE of the model parameters for AP 5, featuring their convergence to the true values. For the sake of completeness, the bottom plot corresponds to the computed model probabilities. The decision process between updating the estimator $\hat{\alpha}_1$ (either according to $\hat{\alpha}_{1,1}$ or $\hat{\alpha}_{1,2}$) is visible in the elapsed time when the model target remains after the model switching.

The RMSE performance was evaluated, and compared to the benchmark methods detailed earlier. Fig. 4.10 shows the average RMSE of distance estimation over all 6 APs. Figure 4.11 shows the corresponding RMSE and PCRLB of position estimation, the latter computed recursively as in [80] and

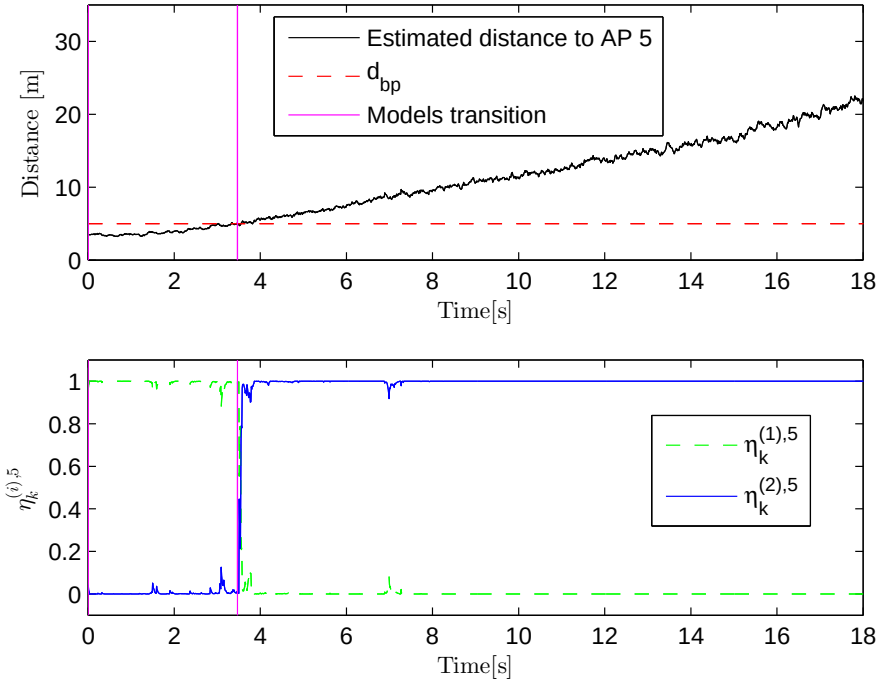


Figure 4.7: Estimated distance according to probability performance η_k^1 to AP 5 for one realization.

derived in Appendix E. From these figures, it is highlighted that our robust indoor localization method has good accuracy when compared to a method that has full knowledge of the model parameters. Clearly, the method where a standard EKF designed to optimally operate on $\mathcal{M}_1$ cannot cope with the complexity of the received RSS measures. This is mostly because there is a large probability of being within $\mathcal{M}_2$ than in $\mathcal{M}_1$. Focusing on AP 5 as our illustrative link, the corresponding RMSE of the four parameters estimation is shown in Fig. 4.12, where we observe the convergence of the ML-based method after some instants.

Finally, we performed some sensitivity tests of the proposed method to deviations from the assumed model parameters. Namely, the designed robust algorithm does not estimate neither d_{bp} nor σ_d^2. Refer to an explanation in Section 4.2.1. However, this instance is taken as a sensitivity analysis of our algorithm to these values. Both cases were taken separately. The first analysis is when $d_{bp} = 3$ m (underestimation) and $d_{bp} = 10$ m (overestimated), recall that $d_{bp} = 5$ m in the simulated data. Figure 4.13 shows the RMSE

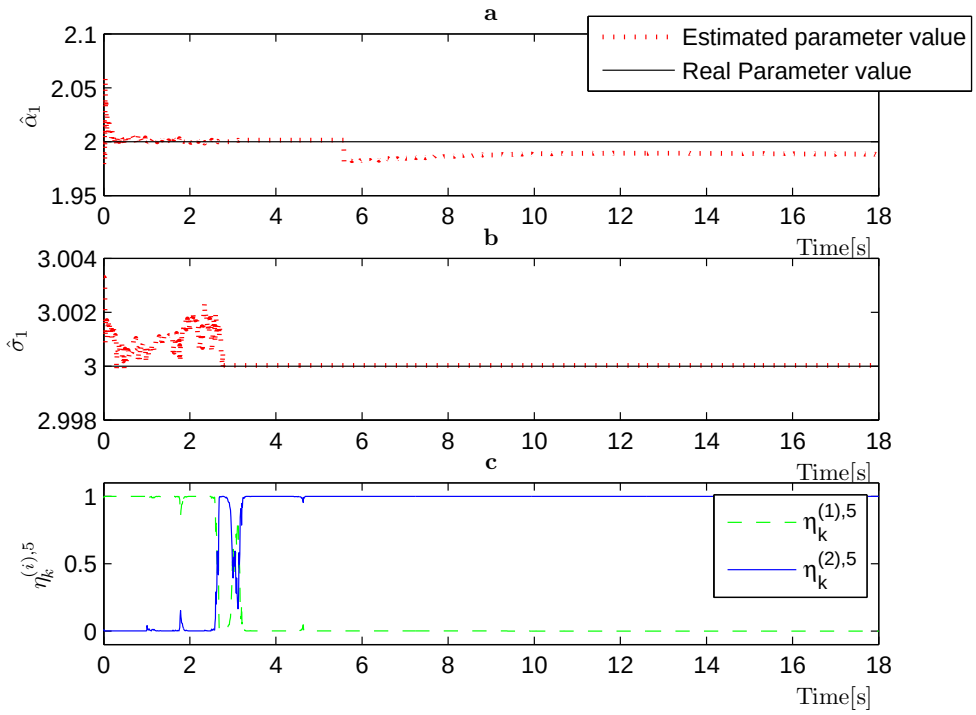


Figure 4.8: α_1 and σ_1 estimation performance for a realization on AP 5.

of position estimation under d_{bp} mismatches, showing that the algorithm has more sensitivity when the value of d_{bp} is overestimated. For the case of having σ_d^2 deviations of $10\sigma_d^2$, and $100\sigma_d^2$, we provide the average RMSE over the simulated trajectory, for the sake of brevity. The average RMSE are 0.3571 m, 0.47551 m, and 0.5701 m, for true σ_d^2, $10\sigma_d^2$, and $100\sigma_d^2$, respectively. These results confirm that the method is not very stressed when σ_d^2 is not perfectly known.

4.4.0.2 Validation with Real data

The measurements were obtained in a real office environment as shown in Figure 4.15 in a NLOS scenario. The architectural plan is the second floor of a typical multi-story office building with drywall and Wood Wall panelings reinforced with aluminum bars. These RSS measurements were used for the model calibration and distance estimation also. For the tracking task, two

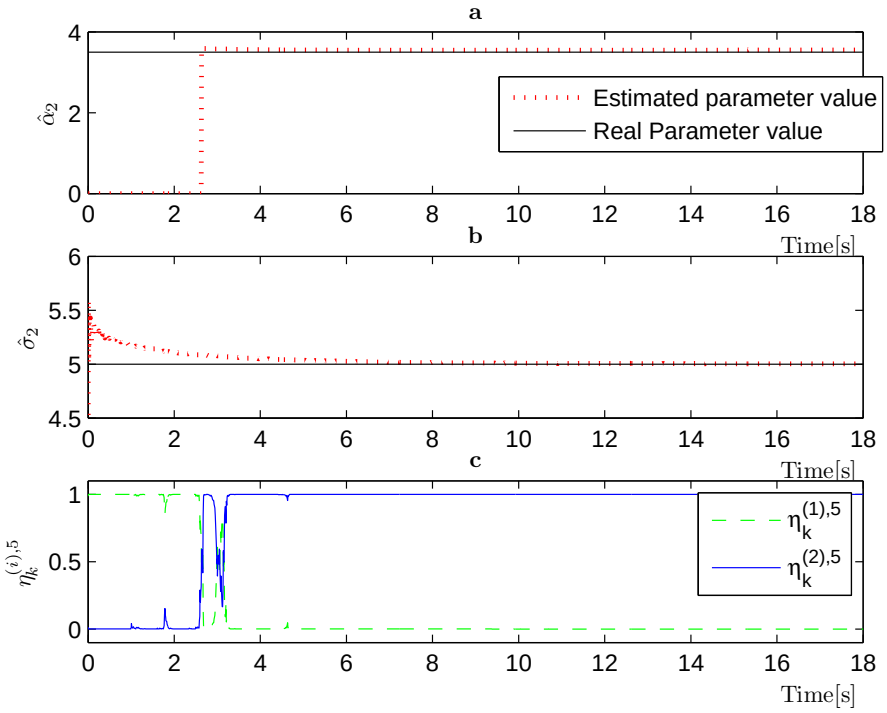


Figure 4.9: α_2 and σ_2 estimation performance for a realization on AP 5.

algorithms were employed. An only EKF modeled with the classical path loss model and the other with the EKF-IMM algorithm. The test-bed used to collect the RSS measurements includes a *RaspberryPi* board. In order to mitigate the antennas orientation problem and the interference due to floor reflection, the antenna's test bed was always faced up and oriented parallel to the AP and with 1 meter from the floor. The measurements were collected during evening and on weekends to ensure that the channel is mostly static during the campaign.

The test-bed used for validation with Real data was described in Section 1.5 and the initial value for $\hat{\boldsymbol{\theta}}_{0|0}$ is the first real distance measurement given by the Robot.

The Mobile path tracking is shown in Figure 4.15 and the distance estimation error to every AP can be seen in Figure 4.14. With real RSS measurementes and an online channel parameterization, the implementation of the EKF-IMM algorithm has a good consistency in positioning terms and is depicted in Figure 4.16. This Figure shows that the accuracy of the EKF-IMM algorithm

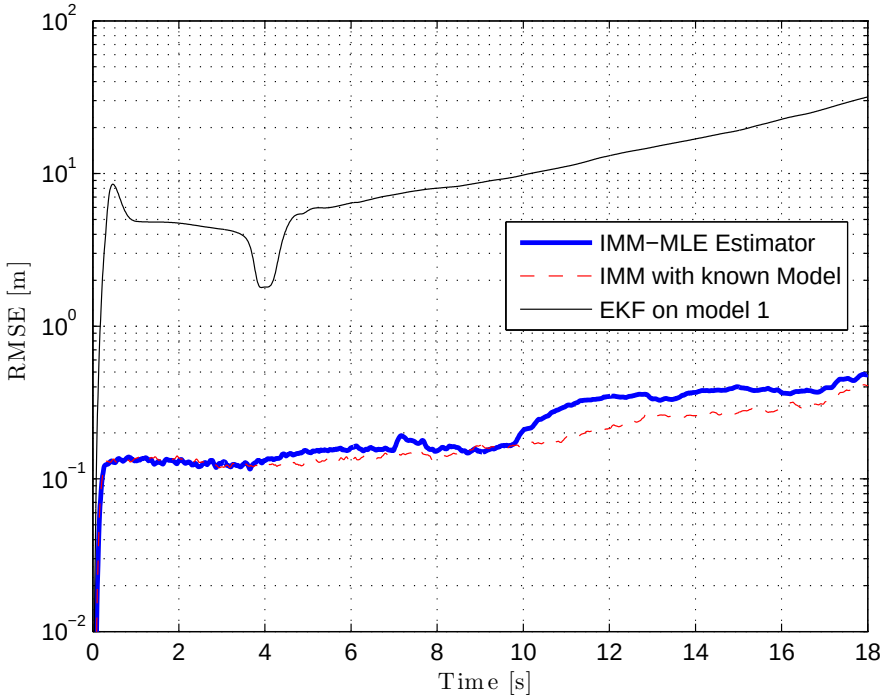


Figure 4.10: Average RMSE performance of the distance estimation between the mobile target to every AP.

developed in this thesis gives a good accuracy in terms of tracking.

4.5 Results with Real RSS/INS measurements

The RSS measurements were taken in a real multi-story office building with drywall and Wood Wall panelings reinforced with aluminum bars. The architectural plan and mobile target's path is shown in Figure 4.18. The RSS measurements were collected in a Line-of-Sight (LOS) condition.

For the tracking task, two algorithms were employed:

- The IMM-based localization algorithm developed in [5] for the two-slope model.

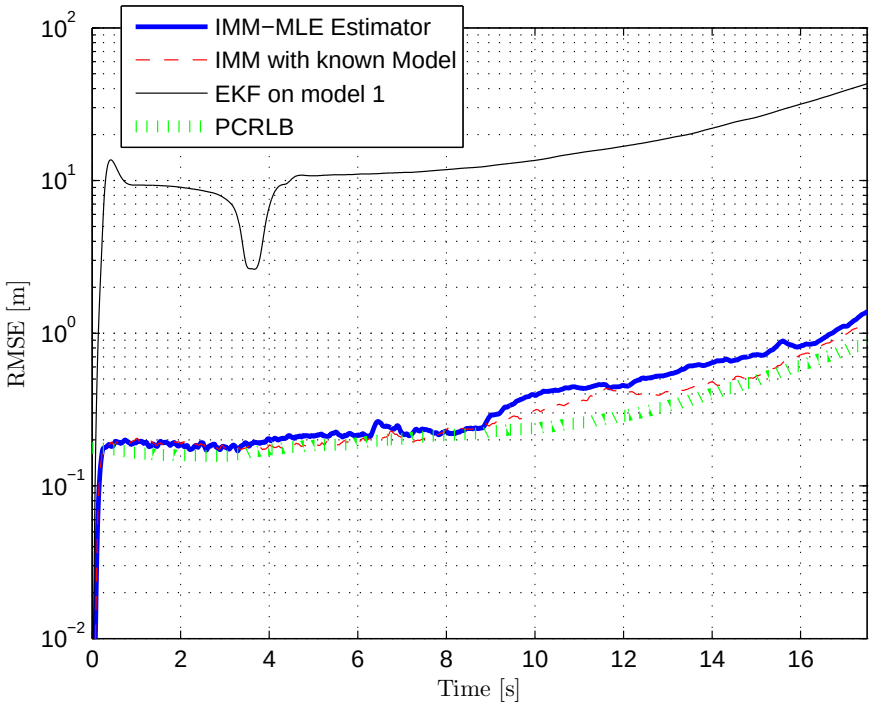


Figure 4.11: RMSE performance of position estimation.

- The comparative of the improvement in the accuracy of indoor localization adding INS measurements. The channel calibration was previously realized in [113].

Following the same Hardware setup described in 1.5, the *RaspberryPi* is connected to an IMU through a I^2C bus. This IMU (Pololu AltIMU-10), acquires the INS measurements. The Pololu AltIMU-10 features a three-axes accelerometer/magnometer (L3GD20) and a three-axes gyroscope (LSM303DLHC). The rotation, acceleration and magnetic readings are independent providing an attitude and heading reference system. The respective axes of the two chips are aligned on the board to facilitate these sensor fusion calculations. In the Figure 4.17, an schematic of the overall system is seen where the ranging/positioning payload reads RSS/INS measurements employing a TL-WN722N WiFi card. Data fusion algorithms and ranging models can be implemented and tested in this platform or logged into a database for offline processing purposes.

The benefits of using a two-slopes path loss model in the distance estimation to a given AP was presented previously in [113]. The results of localization

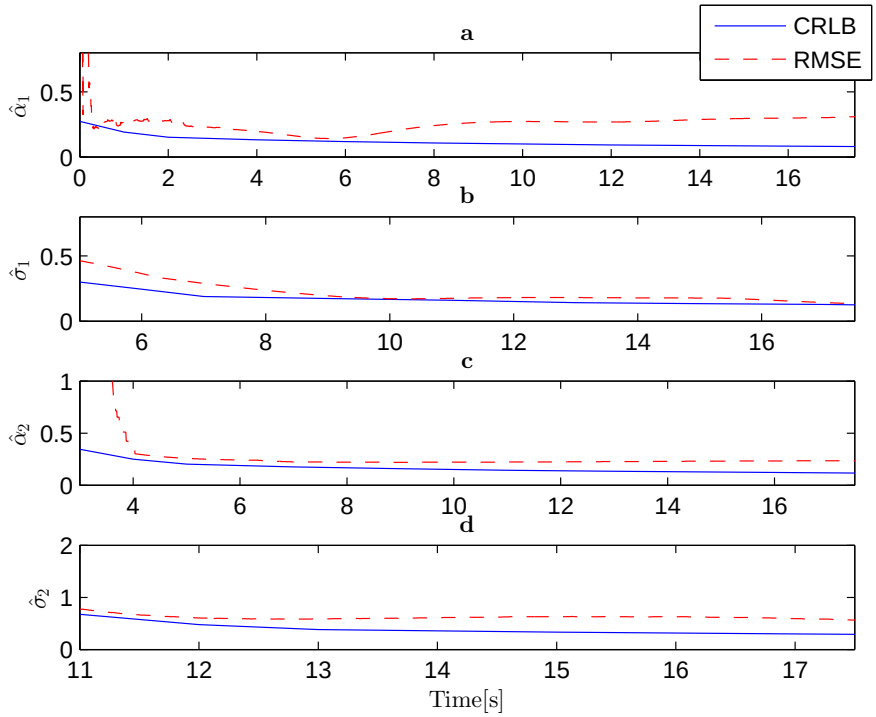


Figure 4.12: Parameter estimation performance. RMSE of the estimation of the four parameters for AP 5 compared with CRLB derived in [5]

estimation presented in the current work correspond to the addition of INS measurements to the EKF-IMM algorithm and how they compare to the results where only RSS measurements were used.

The EKF-IMM algorithm was implemented in a LOS environment as shown in Figure 4.18. The trajectory of the robot last 86 seconds following the same trajectory as in Figure 4.15. The real mobile path versus the estimated trajectory obtained with the IMM algorithm is plotted in this figure, as well as the estimates using RSS measurements only.

For the sake of comparison, Figure 4.19 shows the error cumulative distribution (CDF). Integrating INS measurements to our EKF-IMM Algorithm, gives an improvement of 31.99% in comparison to using only RSS measurements. The mean error with integration of INS measurement is 0.4125 meters and for RSS-only measurements is 0.6065 meters.

With real measurements and an online channel parametrization [113], the implementation of the EKF-IMM Algorithm integrated with INS measurements

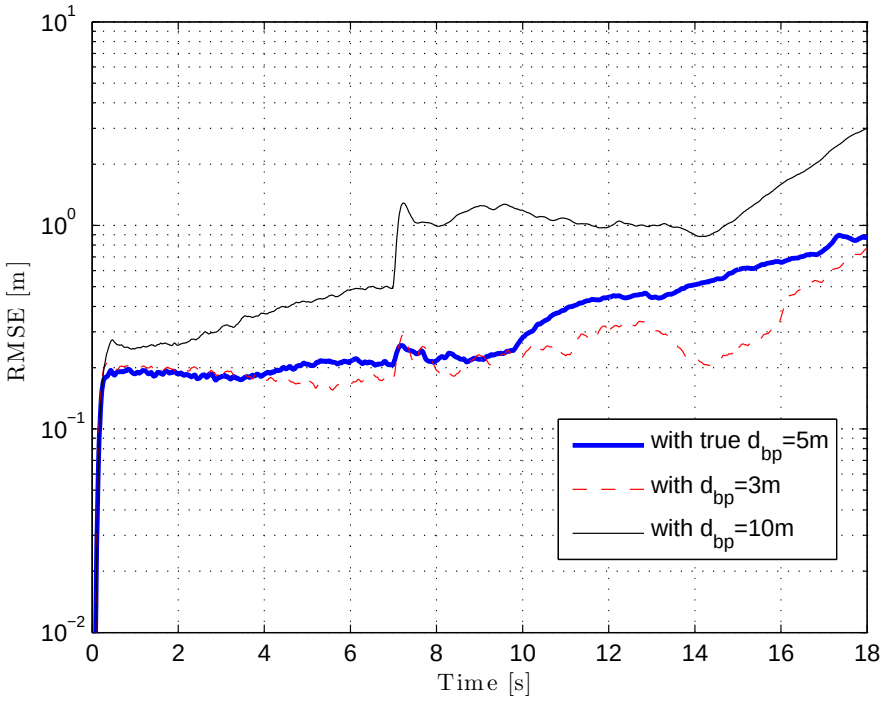


Figure 4.13: RMSE Position estimation performance considering the impact of an overestimated and underestimated value of d_{bp}.

gives a good consistency in terms of indoor location.

4.6 Conclusions

The integration of RSS and INS measurements in a EKF-IMM based solution algorithm was presented in this Chapter. Particularly, we consider an extension to the path-loss model where two slopes are allowed for the path loss exponent, model validated in Chapter 3. Experimental results with real data show that the integration of INS measurements increase positioning accuracy with respect to using RSS measurements only. The accelerometer bias is estimated within the method, and thus corrected at each time instant.

Mobile location via RSS measurements has been formulated as a switching non-linear state-space problem, accounting for realistic conditions where RSS

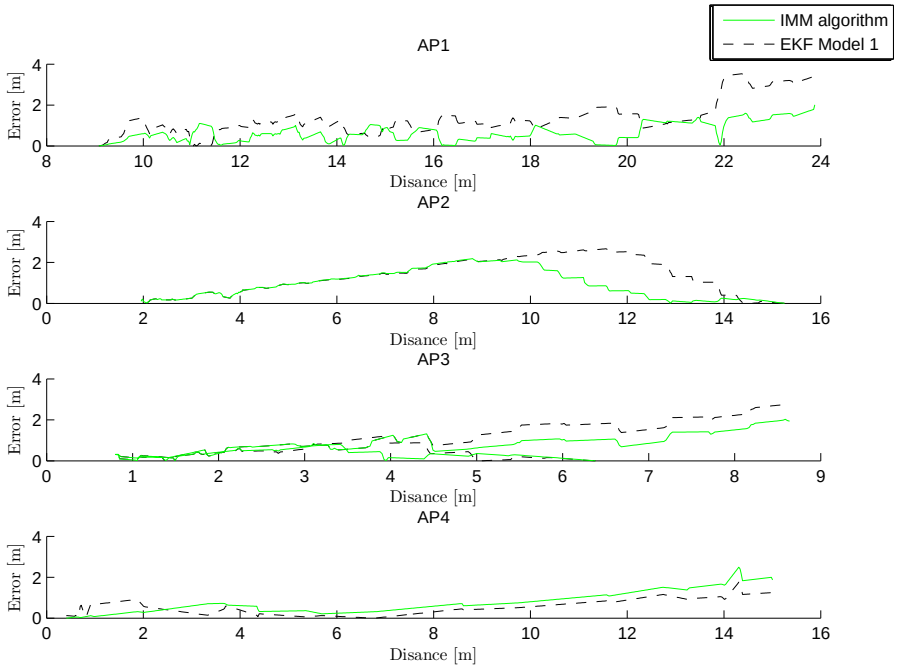


Figure 4.14: Error for distance estimation for every AP .

measurements were seen to follow two propagation models depending on the relative distance to the reference nodes.

A robust EKF-IMM algorithm, including an on-line ML estimation procedure to sequentially adapt the model parameters was proposed. Using the model likelihood functions, the proposed method provides accurate distance estimates between the mobile and each anchor point, which are used for position tracking. Simulation results under realistic WLAN scenarios showed that the proposed EKF-IMM algorithm provides both a good mobile estimation and channel calibration, and much better performance compared to the state-of-the-art techniques.

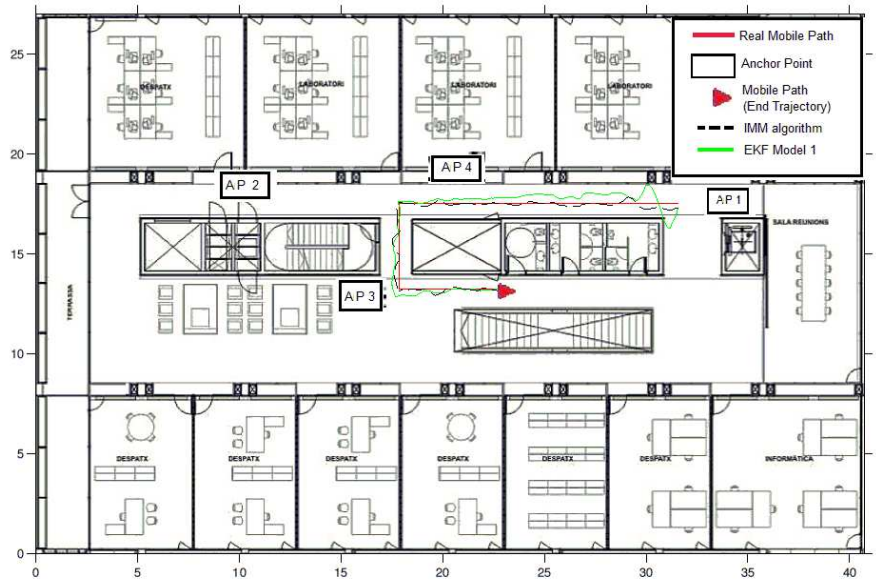


Figure 4.15: Office map. Anchor point locations and mobile path are marked. This mobile target's path corresponds to the second experiment.

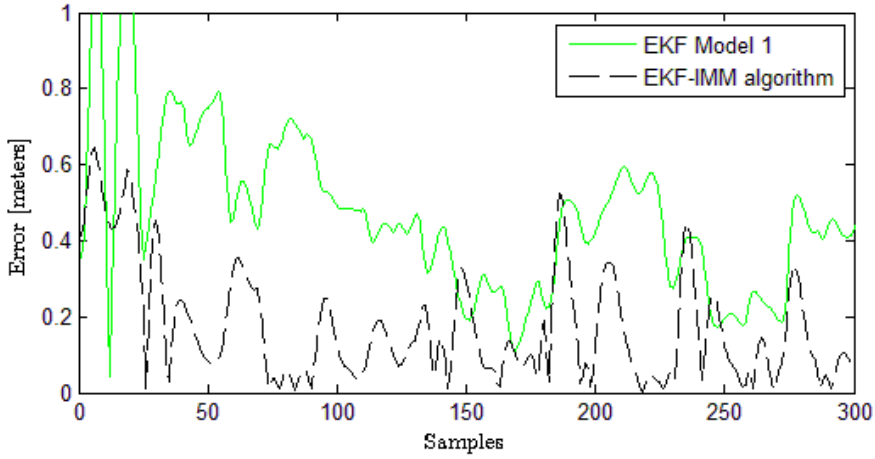


Figure 4.16: Error for position estimation.

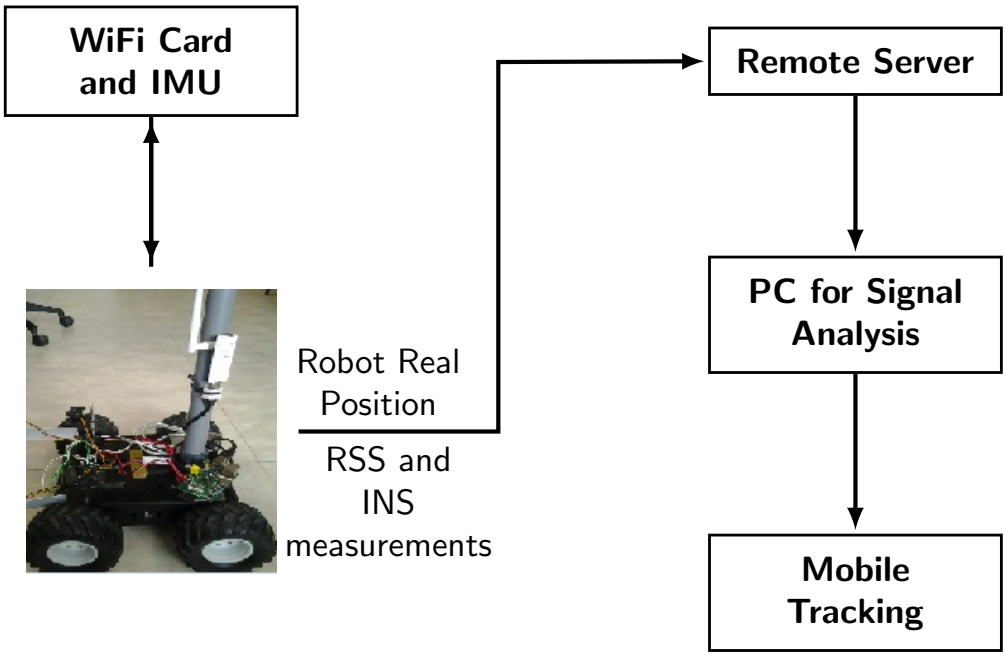


Figure 4.17: Experimental system connection diagram.

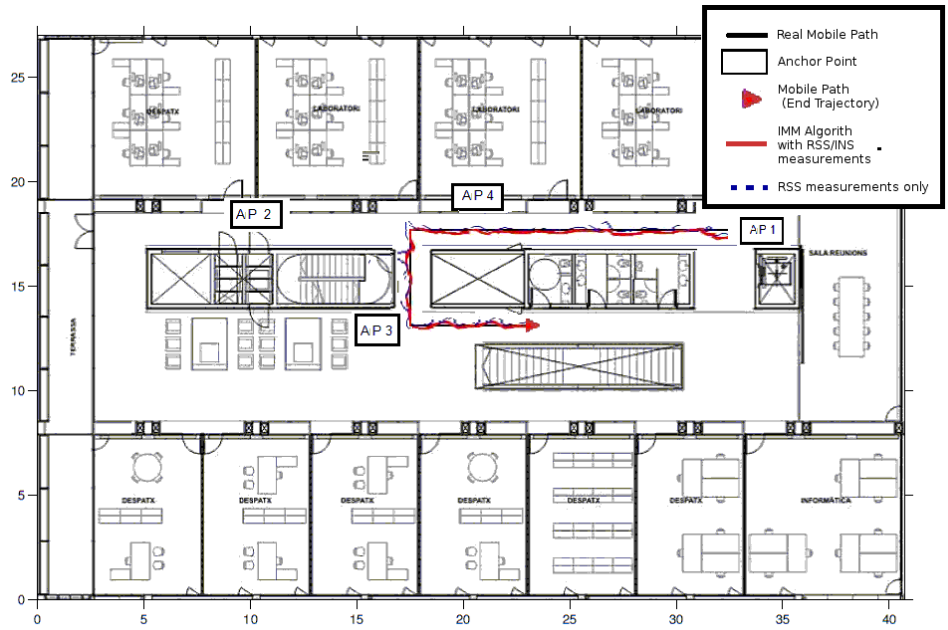


Figure 4.18: Office map. APs location and mobile path are marked. The real trajectory is a straight line between the initial and the endpoint of the mobile target.

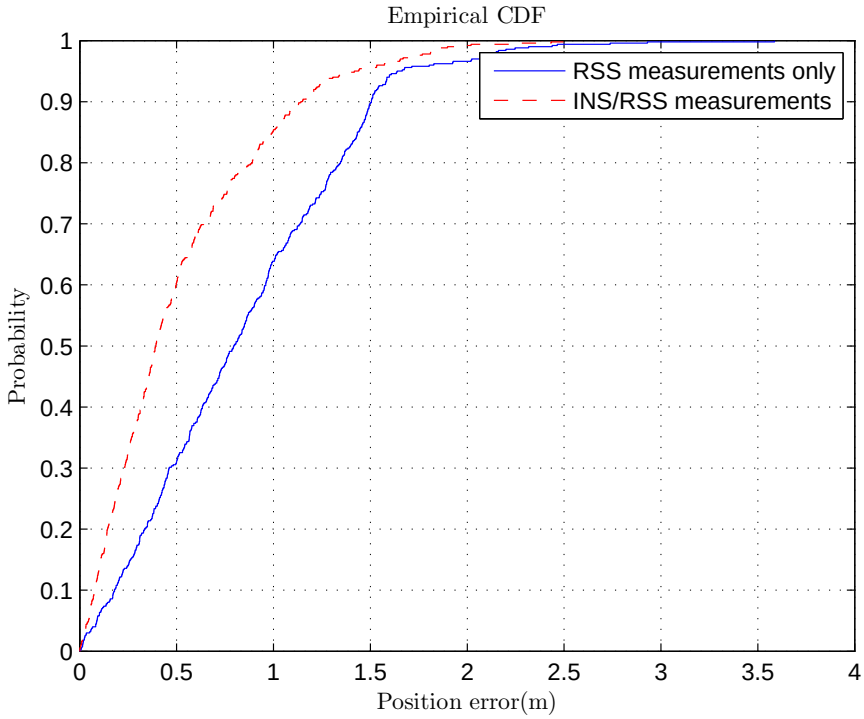


Figure 4.19: Error cumulative distribution function using IMM-based location algorithm with RSS only and RSS/INS measurements.

5

Conclusions and Future Research

This thesis addressed the problem of indoor positioning in WLAN Networks. The main concern in this thesis is tracking a mobile node along with a online channel model calibration. Several methodologies and techniques were studied to develop an optimal Robust positioning indoor algorithm attempting to obtain a good accuracy in terms of error of positioning. In this chapter, the main contributions of this thesis and future research suggestions are mentioned.

In **Chapter 3** the problem of channel model parameterization is addressed. The two-slope path loss model, is considered. This channel model can be described by two models which depend on a breakpoint distance value. That is, for far distances below the breakpoint value, the RSS measurement obeys a first model, and above such breakpoint distance a second one,which overcomes the practical limitations of the standard path loss formulation. Considering this channel model, this contribution proposes a novel channel calibration algorithm based on Bayesian statistics as well as to derive the theoretical accuracy bound of a method calibrating such a model. The use of real data is considered in this work as well to validate the computational simulations.

Several channel parameter estimation methods were studied in **Chapter 2**. In this thesis, MCMC sampling methods appears as a good approach referring to Metropolis - Hastings algorithm and Gibbs sampling which is particularly used in this thesis as our parameter estimation method. As a benchmark for the channel method calibration, the CRLB parameters were derived. The theoretical lower bound is derived under the assumption of an unknown d_{bp} which has not been addressed previously in the literature. The comparative between the computational results and CRLB illustrates the convergence and efficiency of the Bayesian estimator. This comparative was made with CRLB and not with the Bayesian CRLB because the prior is not informative.

The algorithm developed in this chapter was performed in a simulated realistic scenario by two experiments to evaluate the effect of the RSS measurements and its impact in the channel parameter inference:

- The maximum distance between the receiver and the mobile target was divided in several interval distances. Per every interval distance, one RSS sample was taken and in this way, the granularity of RSS measurements in distance was adjusted.
- For the other experiment, the interval distance between mobile target and the AP was fixed and the granularity of RSS measurements was modified by increasing the samples per each interval distance.

For both experiments, a comparative between the lower bounds computed in this work and the lower bound that considers only a classical one-slope model was also derived in this contribution. It can be observed that the RMSE of the proposed algorithm attains the derived CRLB, regardless the assumptions of the bound are more optimistic when d_{bp} is known.

The second experiment corresponding when we take more than one RSS measurements sample for a fixed interval distance, has given us very promising results. These two experiments and our proposed algorithm was validated with real RSS measurements in a real scenario. The results with real data shows that our Bayesian channel calibration algorithm has good accuracy in both cases leading to a future work as a robust RSS indoor localization algorithm.

For future research work, the development and comparative with other parameters estimation algorithms focused for the two-slope model described in **Chapter 2** and the measuring of their respective computational complexities, appears as an interesting area of studio .

Chapter 4 addresses the development of a Robust indoor algorithm. In literature, few works have considered the localization problem using RSS measures together with the adaptive estimation of the propagation model parameters taking into account the classic slope model as the propagation model. In this contribution, a generalized two-slope path loss model to overcome the practical limitations of the conventional approach is used. This can be seen as an improvement of the standard localization case. To solve this problem a more sophisticated solution is needed to cope with the different distance-dependent RSS measurement models. A solution provided in this thesis can be simplified to optimally cope with it when model parameters are known also.

This work proposed a robust EKF-IMM algorithm,including an on-line ML estimation procedure to sequentially adapt the model parameters. Using the model likelihood functions, the proposed method provides accurate distance estimates between the mobile and each anchor point, which are used for position tracking. Simulation results under realistic WLAN scenarios showed that the proposed EKF-IMM algorithm provides both a good mobile estimation and channel calibration, and much better performance compared to the state-of-the-art techniques.

The algorithm was implemented under a simulated realistic scenario. A batch of Monte Carlo simulations were performed to evaluate the RMSE performance of our method. The channel parameter estimated by the ML were compared with the CRLB derived in this thesis (see **Chapter 3**) giving us a good approach for our method. As well, the accuracy of our EKF-IMM algorithm is measured by the error in distances estimates.

For the estimation of the mobile target's position, the PCRLB derived in this thesis was compared with the RMSE of our simulation results showing a good approach.

Our EKF-IMM algorithm was implemented with realistic RSS measurements obtained of a real WLAN scenario to support our computational results and different trajectories of the mobile target were performed. The results obtained with real data has shown a satisfactory estimation of the trajectory of our mobile target.

Another experiment with real data was performed but adding INS measurements in this case. The position and trajectory of the mobile target was compared with RSS measurements only and with the addition of INS measurements to our EKF-IMM algorithm. The error obtained was significantly less when we use RSS/INS measurements than when we use only RSS measurements. This is because we are adding information (orientation and acceleration) to our system giving a more suitable performance of our algorithm in the estimation of the position of our mobile target.

As future research work suggested is:

- An on-line breakpoint distance estimation because just an analysis of the breakpoint distance sensitivity was made in this thesis.
- Because our algorithm is based on EKF filters, the use and the comparative of the performance between other filter approaches such as the UKF, CKF and the mentioned in **Chapter 1** in terms of position error and computational complexity, is suggested.
- A preliminary study before using real data was to know the impact in the estimation of position of the mobile target when we have RSS measurements in a NLOS or a LOS scenario. This aspect showed a significant error in the position of our mobile target when we compared the results of both scenarios. For this reason, a LOS environment was taken into account in this thesis. Thus, as a future work, an NLOS/LOS algorithm identifier/mitigation is proposed also.

Appendices

A

CRLB Derivation for Parameter Space Vector

The likelihood function of the N measurements for the two-slope model conditioned to the unknown parameters $\boldsymbol{\psi}$ can be expressed as:

$$p(\mathbf{y}|\boldsymbol{\psi}) = p(y_1, y_2, ..., y_{N_{d_{bp}}}) \cdot p(y_{N_{d_{bp}}+1}, y_{N_{d_{bp}}+2}, ..., y_N), \tag{A.1}$$

where $\left\{y_1, y_2, ..., y_{N_{d_{bp}}}\right\}$ are distributed according to the first model (before d_{bp}) and
$\left\{y_{N_{d_{bp}}+1}, y_{N_{d_{bp}}+2}, ..., y_N\right\}$ following the second model (after d_{bp}), as defined in 3.4.

The likelihood function of the N independent measurements for the two-slope model conditioned to the unknown $\boldsymbol{\psi}$ is:

$$p(\mathbf{y} \mid \boldsymbol{\psi}) = p(y_1, y_2, \cdots, y_{N_{d_{bp}}} \mid \boldsymbol{\psi}) \cdot p(y_{N_{d_{bp}}+1}, y_{N_{d_{bp}}+2}, \cdots, y_N \mid \boldsymbol{\psi}) \tag{A.2}$$

where $\{y_1, \cdots, y_{N_{d_{bp}}}\}$ are distributed according to the first model (before d_{bp}) and
$\{y_{N_{d_{bp}}+1}, \cdots, y_N\}$ following the second model (after d_{bp}), as defined in (3.4).

Under the assumption that measurements are independent and Gaussian distributed we have that

$$p(\mathbf{y} \mid \boldsymbol{\psi}) \sim \prod_{n=1}^{N_{d_{bp}}} \mathcal{N}\left(h^{(1)}(d_n), \sigma_1^2\right) \cdot \prod_{n=N_{d_{bp}}+1}^{N} \mathcal{N}\left(h^{(2)}(d_n), \sigma_2^2\right) \tag{A.3}$$

where

$$h^{(1)}(d_n) = L_o + 10\alpha_1 \log_{10}(d_n) \tag{A.4}$$

$$h^{(2)}(d_n) = h^{(1)}(d_{bp}) + 10\alpha_2 \log_{10}(d_n/d_{bp}). \tag{A.5}$$

Then, by definition of the Gaussian function[1]

$$p(\mathbf{y} \mid \boldsymbol{\psi}) = \prod_{n=1}^{N_{d_{bp}}} \frac{1}{(2\pi\sigma_1^2)^{\frac{N_{d_{bp}}}{2}}} \exp\left[-\frac{(y_n - h^{(1)}(d_n))^2}{2\sigma_1^2}\right] \cdot \\ \cdot \prod_{n=N_{d_{bp}}+1}^{N} \frac{1}{(2\pi\sigma_2^2)^{\frac{N-N_{d_{bp}}}{2}}} \exp\left[-\frac{(y_n - h^{(2)}(d_n))^2}{2\sigma_2^2}\right]. \tag{A.6}$$

Further, natural logarithm of the Gaussian function is given by:

$$\log \mathcal{N}\left(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}^2\right) = -\frac{1}{2}\log 2\pi - \frac{1}{2}\log \boldsymbol{\Sigma}^2 - \frac{1}{\boldsymbol{\Sigma}^2}\left(\mathbf{x} - \boldsymbol{\mu}\right)^2, \tag{A.7}$$

and applying natural logarithm to (A.6), we obtain

$$\log p(\mathbf{y} \mid \boldsymbol{\psi}) = -\frac{N_{d_{bp}}}{2}\left(\log 2\pi + \log \sigma_1^2\right) - \frac{N - N_{d_{bp}}}{2}\left(\log 2\pi + \log \sigma_2^2\right) - \\ - \frac{1}{2\sigma_1^2}\sum_{n=1}^{N_{d_{bp}}}\left(y_n - h^{(1)}(d_n)\right)^2 - \frac{1}{2\sigma_2^2}\sum_{n=N_{d_{bp}}+1}^{N}\left(y_n - h^{(2)}(d_n)\right)^2. \tag{A.8}$$

Pluggin (A.8) in (3.6), the elements of the non-null FIM are calculated. This yields to:

$$\boldsymbol{\mathcal{I}}(\boldsymbol{\psi}) = \begin{pmatrix} I_{11} & 0 & I_{13} & 0 & I_{15} \\ 0 & I_{22} & 0 & 0 & 0 \\ I_{31} & 0 & I_{33} & 0 & I_{35} \\ 0 & 0 & 0 & I_{44} & 0 \\ I_{51} & 0 & I_{53} & 0 & I_{55} \end{pmatrix}, \tag{A.9}$$

[1]The probability density of the Gaussian distribution is: $f\left(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}^2\right) = \frac{1}{(2\pi\boldsymbol{\Sigma}^2)^{\frac{1}{2}}} \cdot e^{-\frac{(\mathbf{x}-\boldsymbol{\mu})^2}{2\boldsymbol{\Sigma}^2}}$

where the FIM elements ofA.9 are:

$$I_{11} = \frac{100}{\sigma_1^2} \sum_{n=1}^{N_{d_{bp}}} (\log_{10}(d_n))^2 + \frac{100(\log_{10}(d_{bp}))^2}{\sigma_2^2}(N - N_{d_{bp}}) \tag{A.10}$$

$$I_{22} = \frac{N_{d_{bp}}}{2\sigma_1^4} \tag{A.11}$$

$$I_{33} = \frac{100}{\sigma_2^2} \sum_{n=N_{d_{bp}}+1}^{N} \left(\log_{10} \frac{d_n}{d_{bp}}\right)^2 \tag{A.12}$$

$$I_{44} = \frac{N - N_{d_{bp}}}{2\sigma_2^4} \tag{A.13}$$

$$I_{55} = \frac{100(N - N_{d_{bp}})(\alpha_2 - \alpha_1)^2}{\sigma_2^2 d_{bp}^2} \tag{A.14}$$

$$I_{13} = I_{31} = \frac{100 \log_{10}(d_{bp})}{\sigma_2^2} \sum_{n=N_{d_{bp}}+1}^{N} \log_{10} \frac{d_n}{d_{bp}} \tag{A.15}$$

$$I_{51} = I_{15} = \frac{100(N - N_d)(2\alpha_1 - \alpha_2) \log_{10}(d_{bp})}{\sigma_2^2 d_{bp}} \tag{A.16}$$

$$I_{53} = I_{35} = \frac{100(\alpha_1 - 2\alpha_2)}{\sigma_2^2 d_{bp}} \sum_{n=N_{d_{bp}}+1}^{N} \log_{10} \frac{d_n}{d_{bp}} \ , \tag{A.17}$$

Then, the CRLB can be computes as

$$\mathbf{CRLB}(\boldsymbol{\psi}) \geq \boldsymbol{\mathcal{I}}^{-1})(\boldsymbol{\psi}) \tag{A.18}$$

B

Conditional Distributions for Gibbs Sampling

In this appendix is summarized the used distribution forms in this work.

Since the likelihood function has the form

$$p(\mathbf{y}|\mu_2(n), \sigma_2^2, \mu_2(n), \sigma_2^2) = \tag{B.1}$$

$$p(\mathbf{y_1}|\mu_1(n_{1\to N_{bp}}), \sigma_1^2) \cdot p(\mathbf{y_2}|\mu_2(n_{N_{dbp+1}\to N}) \tag{B.2}$$

where

$$p(\mathbf{y_1}|\mu_1(n_{1\to N_{bp}}), \sigma_1^2) \sim \mathcal{N}(\mu_1(n_{1\to N_{bp}}, \sigma_1^2) \tag{B.3}$$

$$p(\mathbf{y_2}|\mu_2(n_{N_{dbp+1}\to N}) \sim \mathcal{N}(\mu_2(n_{N_{dbp}+1\to N}), \sigma_2^2) \tag{B.4}$$

the natural conjugate prior has the form

$$p(\alpha_i) \propto \exp\left[-\frac{\alpha_i^2}{2\mathcal{V}_{\alpha_i}^2}\right] \propto \mathcal{N}(\alpha_i|0, \mathcal{V}_{\alpha_i}^2) \tag{B.5}$$

B.0.0.3 Normal Posterior

The posterior is given by

$$p(\alpha_i|\mathbf{y_i}) \propto p(\mathbf{y_i}|\mu_i(n), \sigma_i^2,) \cdot p(\alpha_i|0, \mathcal{V}_{\alpha_i}^2) \tag{B.6}$$

where $i = 1, 2$. For α_1 parameter inference, B.6 is,

$$
\begin{aligned}
p(\alpha_1|\mathbf{y_1}) &\propto \exp\left[-\frac{1}{2\sigma_1^2}\sum_{n=1}^{N_{dbp}}(y_n-\mu_1(n))^2\right]\cdot\exp\left[-\frac{\alpha_1^2}{2\mathcal{V}_{\alpha_1}^2}\right]\\
&\propto \exp\left[-\frac{1}{2\sigma_1^2}\sum_{n=1}^{N_{dbp}}(y_n^2+\mu_1^2(n)-2y_n\mu_1(n))+\frac{-\alpha_1^2}{\mathcal{V}_{\alpha_1}^2}\right]\\
&\propto \exp\Bigg\{-\frac{1}{2\sigma_1^2}\Bigg[\sum_{n=1}^{N_{dbp}}y_2^n-2\sum_{n=1}^{N_{dbp}}y_n(L_o+10\alpha_1\log_{10}d_n)+\\
&+(10\alpha_1\log_{10}d_n))^2\Bigg]-\frac{\alpha_1^2}{\mathcal{V}_{\alpha_1}^2}\Bigg\}\\
&\propto \exp\Bigg[-\frac{\alpha_1^2}{2}\left(\frac{100\sum_{n=1}^{N_{dbp}}\log_{10}d_n}{\sigma_1^2}+\frac{1}{\mathcal{V}_{\alpha_1}^2}\right)+\\
&+\frac{\alpha_1}{2\sigma_1^2}\left(20\sum_{n=1}^{N_{dbp}}y_n\log_{10}d_n-20L_o\sum_{n=1}^{N_{dbp}}\log_{10}d_n)\right)-\\
&-\frac{1}{2\sigma_1^2}\left(\sum_{n=1}^{N_{dbp}}(y_n-L_o)^2\right)\Bigg]\\
&\overset{def}{=}\exp\left[-\frac{1}{2\sigma_{1,o}^2}(\alpha_1^2-2\alpha_1\alpha_{1,o}+\alpha_{1,o}^2)\right]\\
&=\exp\left[-\frac{1}{2\sigma_{1,o}^2}(\alpha_1-\alpha_{1,o})^2\right]\propto\mathcal{N}(\alpha_{1,o},\sigma_{1,o}^2)
\end{aligned}
$$

matching coefficients of $\alpha_{1,o}^2$, $\sigma_{1,o}^2$ is found:

$$
\begin{aligned}
-\frac{\alpha_1^2}{2\sigma_{1,o}^2} &= \frac{\alpha_1^2}{2}\left(\frac{\sum_{n=1}^{N_{dbp}}\log_{10}^2 d_n}{\sigma_1^2}+\frac{1}{\mathcal{V}_{\alpha_1}^2}\right)\\
\frac{1}{\sigma_{1,o}^2} &= \frac{100\mathcal{V}_{\alpha_1}^2+\sigma_1^2}{\sigma_1^2\mathcal{V}_{\alpha_1}^2}\\
\sigma_{1,0}^2 &= \frac{\sigma_1^2\mathcal{V}_{\alpha_1}^2}{100\mathcal{V}_{\alpha_1}^2\sum_{n=1}^{N_{dbp}}\log_{10}^2 d_n+\sigma_1^2}
\end{aligned}
\tag{B.7}
$$

Now, matching in the same procedure the coefficients of α_1, $\alpha_{1,o}$ is obtained:

$$\frac{-2\alpha_1\alpha_{1,o}}{-2\sigma_{1,o}^2} = \frac{\alpha_1}{\sigma_1^2}\left(20\sum_{n=1}^{N_{dbp}} y_n \log_{10} d_n - 20L_o\sum_{n=1}^{N_{dbp}} \log_{10} d_n)\right)$$

$$\alpha_{1,o} = 20\frac{\sigma_{1,0}^2}{\sigma_1^2}\left(\sum_{n=1}^{N_{dbp}} y_n \log_{10} d_n - L_o\sum_{n=1}^{N_{dbp}} \log_{10} d_n)\right) \tag{B.8}$$

The process for α_2 estimating, follows the same procedure used to obtain α_1. The likelihood is written as

$$p(\alpha_2|\mathbf{y_2}) \propto \exp\left[-\frac{1}{2\sigma_1^2}\right] \cdot \exp\left[-\frac{\sigma_2^2}{\mathcal{V}_{\alpha 2}^2}\right] \tag{B.9}$$

and the natural conjugate prior for α_2 is

$$p(\alpha_1) \propto \exp\left[-\frac{\alpha_1^2}{2\mathcal{V}_{\alpha_1^2}}\right] \propto \mathcal{N}(\alpha_1|0, \mathcal{V}_{\alpha_1}^2) \tag{B.10}$$

the posterior distribution for α_2 is defined as:

$$\begin{aligned} p(\alpha_2|\mathbf{y_2}) &\propto \exp\left[-\frac{1}{2\sigma_2^2}\sum_{n=N_{dbp}+1}^{N}(y_n - \mu_2(n))^2\right] \cdot \\ &\cdot \exp\left[-\frac{\alpha_2^2}{2\mathcal{V}_{\alpha_2}^2}\right] \\ &\stackrel{def}{=} \exp\left[-\frac{1}{2\sigma_{2,o}^2}(\alpha_2 - \alpha_{2,o})^2\right] \propto \mathcal{N}(\alpha_{2,o}, \sigma_{2,o}^2) \end{aligned} \tag{B.11}$$

Following the same procedure to obtain $\alpha_{2,o}$, $\sigma_{2,o}^2$, we obtain:

$$\sigma_{2,0}^2 = 100\frac{\sigma_2^2\mathcal{V}_{\alpha_2^2}^2}{\mathcal{V}_{\alpha_2}^2\sum_{n=N_{dbp}+1}^{N}\log_{10}\frac{d_n}{d_{bp}}} \tag{B.12}$$

$$\alpha_{2,o} = 20\frac{\alpha_{2,o}^2}{\sigma_2^2}\left[\alpha 2\sum_{n=N_{dbp}+1}^{N} y_n \log_{10}\frac{d_n}{d_{bp}} - \right. \tag{B.13}$$

$$\left. - (L_o + 10\alpha_1 \log_{10} d_{bp})\sum_{n=N_{dbp}+1}^{N}\log_{10}\frac{d_n}{d_{bp}}\right] \tag{B.14}$$

C

ML Estimators derivation for $\mathcal{M}_1$

The considered first model parameters to be estimated are $\boldsymbol{\varphi}_1 = [\alpha_1, \sigma_1^2]^\top$. Recall that the set of available samples are gathered in $\mathcal{Y}_{1,k}^r$, which are Gaussian distributed,$y_{1,\ell}^r \sim \mathcal{N}(\bar{y}_{1,\ell}^r, \sigma_1^{2,r})$, Then, for the r^{th} AP, the likelihood function at instant k is expressed as:

$$\mathcal{L}(\boldsymbol{\varphi}_1^r|\mathcal{Y}_{1,k}^r) = \prod_{\ell=1}^{U_1} \mathcal{N}(\bar{y}_{1,\ell}, \sigma_1^2) = \frac{1}{(\sqrt{2\pi\sigma_1^2})^{U_1}} \prod_{\ell=1}^{U_1} \exp\left[-\frac{(y_{1,\ell} - \bar{y}_{1,\ell})^2}{2\sigma_1^2}\right]. \quad \text{(C.1)}$$

Applying natural logarithm, the log-likelihood is

$$\log \mathcal{L}(\boldsymbol{\varphi}_1|\mathcal{Y}_{1,k}^r) = -\frac{U_1}{2}\log(2\pi\sigma_1^2) - \frac{1}{2\sigma_1^2}\sum_{\ell=1}^{U_1}\left(y_{1,\ell} - L_0 + 10\alpha_{1,1}\log_{10} d_\ell\right)^2. \quad \text{(C.2)}$$

The ML estimator of $\boldsymbol{\varphi}_1^r$ is obtained from the following

$$\frac{\partial \log \mathcal{L}(\boldsymbol{\varphi}_1|\mathcal{Y}_{1,k}^r)}{\partial \sigma_1^2} = -\frac{U_1}{2}\frac{2\pi}{2\pi\sigma_1^2} - \frac{1}{2\sigma_1^2}\sum_{\ell=1}^{U_1}\left(y_{1,\ell} - L_0 + 10\alpha_{1,1}\log_{10} d_\ell\right)^2 = 0, \quad \text{(C.3)}$$

what leads to the following estimator,

$$\hat{\sigma}_1^2 = \frac{1}{U_1}\sum_{\ell=1}^{U_1}\left(y_{1,\ell} - \bar{y}_{1,\ell}\right)^2. \quad \text{(C.4)}$$

Then, the path loss exponent, $\alpha_{1,1}$, is computed from

$$\frac{\partial \log \mathcal{L}(\boldsymbol{\varphi}_1|\mathcal{Y}_{1,k}^r)}{\partial \alpha_{1,1}} = -\frac{1}{2\sigma_1^2}\sum_{\ell=1}^{U_1}[(y_{1,t} - L_0 - 10\alpha_{1,1}\log d_{1,t})(-10\log d_{1,t})] = 0, \tag{C.5}$$

what leads to the following estimator,

$$\hat{\alpha}_{1,1} = \frac{\sum_{\ell=1}^{U_1}[(y_{1,\ell} - L_0)(\log d_\ell)]}{10\sum_{\ell=1}^{U_1}(\log d_\ell)^2}. \tag{C.6}$$

D

ML estimators derivation for $\mathcal{M}_2$

The ML estimators of the second model parameters, $\boldsymbol{\varphi}_2 = [\alpha_1, \alpha_2, \sigma_2^2]^\top$, are obtained following the same procedure. In this case, the observations imputed to $\mathcal{M}_2$ are gathered in $\mathcal{Y}_{2,k}^r$, which are Gaussian distributed, $y_{2,\ell}^r \sim \mathcal{N}(\bar{y}_{2,\ell}^r, \sigma_2^{2,r})$,. Similarly as in the derivation of (4.23), the estimator of $\hat{\sigma}_2^2$ reduces to (4.25).

To find the estimator of $\boldsymbol{\alpha} = \begin{bmatrix}\alpha_1 & \alpha_2\end{bmatrix}$ it is convenient to stack the U_2 observations from $\mathcal{M}_2$ available at instant k into a vector $\mathbf{y}_{2,k}$ such that $\mathbf{y}_{2,k} = L_0 + \mathbf{F}_k\boldsymbol{\alpha} + \mathbf{n}_k$ where

$$\mathbf{F}_k = \begin{bmatrix} 10\log d_{bp} & 10\log\left(\frac{d_1}{d_{d_{bp}}}\right) \\ \vdots & \vdots \\ 10\log d_{bp} & 10\log\left(\frac{d_{U_2}}{d_{d_{bp}}}\right) \end{bmatrix}, \tag{D.1}$$

and $\mathbf{n}_k \sim \mathcal{N}(\mathbf{0}, \sigma_2^2\mathbf{I})$. We omitted the dependence of certain parameters on r for the sake of clarity. With this formulation of the problem, it is straightforward to see that the ML estimator for $\boldsymbol{\alpha}$ is

$$\hat{\boldsymbol{\alpha}} = \left(\mathbf{F}_k{}^\top\mathbf{F}_k\right)^{-1}\mathbf{F}_k{}^\top(\mathbf{y}_{2,k} - L_0)\ . \tag{D.2}$$

following a least squares reasoning, which in this case is equivalent to the MLE.

E

PCRLB Derivation for position

The state evolution is given by:

$$\mathbf{x}_k = \boldsymbol{A}_x \mathbf{x}_{k-1} + \mathbf{B}_x \omega_x \tag{E.1}$$

where:

$$\mathbf{A}_x = \begin{bmatrix} 1 & 0 & \Delta_t & 0 \\ 0 & 1 & 0 & \Delta_t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{E.2}$$

and

$$\mathbf{B}_x = \begin{bmatrix} \frac{(\Delta_t)^2}{2} & 0 \\ 0 & \frac{(\Delta_t)^2}{2} \\ \Delta_t & 0 \\ 0 & \Delta_t \end{bmatrix} \tag{E.3}$$

.

The nonlinear observation function $\mathbf{g}_k(\mathbf{x}_k)$ is defined as the distance ob the mobile target to every AP:

$$\mathbf{g}_k(\mathbf{x}_k) = \begin{bmatrix} \sqrt{(x_k - x_p^1)^2 + (y_k - y_p^1)^2} \\ \vdots \\ \sqrt{(x_k - x_p^N)^2 + (y_k - y_p^N)^2} \end{bmatrix} = \begin{bmatrix} d_k^1 \\ \vdots \\ d_k^N \end{bmatrix} \tag{E.4}$$

Where N is the number of visible APs, and $\left\{x_p^r, y_p^r\right\}$ is the psition of the r_{th} AP, d_k^r is the true distance between the mobile and the r^{th} AP and the corresponding Jacobian of $\mathbf{g}_k(\mathbf{x}_k)$ is given by:

$$\mathbf{G}_k(\mathbf{x}_k) = \begin{bmatrix} \frac{x_k - x_p^1}{d_k^1} & \frac{y_k - y_p^1}{d_k^1} & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ \frac{x_k - x_p^N}{d_k^N} & \frac{y_k - y_p^N}{d_k^N} & 0 & 0 \end{bmatrix} \tag{E.5}$$

Tichavsky et al [80] show that the FIM $\boldsymbol{\mathcal{I}}_k$ can be recursevely calculated as:

$$\boldsymbol{\mathcal{I}}_{k+1} = \mathbf{D}_k^{22} - \mathbf{D}_k^{21} \left(\boldsymbol{\mathcal{I}}_k + \mathbf{D}_k 11\right)^{-1} \mathbf{D}_k^{12} \tag{E.6}$$

where:

$$\mathbf{D}_k^{11} = -\mathbb{E}\left\{\nabla_{\mathbf{x}_k} \nabla_{\mathbf{x}_k}^\top \log p(\mathbf{x}_{k+1}|\mathbf{x}_k)\right\} \tag{E.7}$$

$$\mathbf{D}_k^{21} = \left[\mathbf{D}_k^{12}\right]^\top = -\mathbb{E}\left\{\nabla_{\mathbf{x}_k} \nabla_{x+1,k}^\top \log p(\mathbf{x}_{k+1}|\mathbf{x}_k)\right\} \tag{E.8}$$

$$\mathbf{D}_k^{22} = -\mathbb{E}\left\{\nabla_{\mathbf{x}_{k+1}} \nabla_{\mathbf{x}_{k+1}}^\top \log p(\mathbf{x}_{k+1}|\mathbf{x}_k)\right\} - \mathbb{E}\left\{\nabla_{\mathbf{x}_{k+1}} \nabla_{\mathbf{x}_{k+1}}^\top \log p(\mathbf{z}_{k+1}|\mathbf{x}_k)\right\} \tag{E.9}$$

In the localization problem, we are facing the special case of Additive Gaussian noise. So for a Gaussian distribution with the form of:

$$p(\boldsymbol{\beta}) \sim \mathcal{N}\left(\boldsymbol{\beta}; \bar{\boldsymbol{\beta}}, \boldsymbol{\Sigma}\right) \tag{E.10}$$

for definition, we have:

$$\nabla_{\boldsymbol{\beta}} \log p(\boldsymbol{\beta}) = \nabla_{\boldsymbol{\beta}} \left\{ C - \frac{1}{2} \left[(\boldsymbol{\beta} - \bar{\boldsymbol{\beta}})^\top \boldsymbol{\Sigma}^{-1} (\boldsymbol{\beta} - \bar{\boldsymbol{\beta}}) \right] \right\} \tag{E.11}$$

Following this form, from Equation E.9, $\nabla_{\mathbf{x}_k} \log p(\mathbf{x}_{k+1}|\mathbf{x}_k)$ and $\nabla_{\mathbf{x}_k} \log p(\mathbf{z}_{k+1}|\mathbf{x}_{k+1})$ can be derivated as:

$$\begin{aligned}
\nabla_{\mathbf{x}_k} \log p(\mathbf{x}_{k+1}|\mathbf{x}_k) &= \nabla_{\mathbf{x}_k} \left[-\frac{1}{2} \left[\mathbf{x}_{k+1} - \mathbf{G}_k(\mathbf{x}_k) \right]^\top \mathbf{Q}_{\mathbf{x}_k} \left[\mathbf{x}_{k+1} - \mathbf{G}_k(\mathbf{x}_k) \right] \right] \\
&= \left[\nabla_{\mathbf{x}_k} \mathbf{G}_k^\top(\mathbf{x}_k) \right] \mathbf{Q}_{\mathbf{x}_k}^{-1} \left[\nabla_{\mathbf{x}_k} \mathbf{G}_k^\top(\mathbf{x}_k) \right] && \text{(E.12)} \\
\nabla_{\mathbf{x}_k} \log p(\mathbf{z}_{k+1}|\mathbf{x}_{k+1}) &= \left[\nabla_{\mathbf{x}_k} \mathbf{G}_k^\top(\mathbf{x}_k) \right] \boldsymbol{R}_{\boldsymbol{x}_k}^{-1} \left[\nabla_{\mathbf{x}_k} \mathbf{G}_k^\top(\mathbf{x}_k) \right] && \text{(E.13)}
\end{aligned}$$

So, we have:

$$\begin{aligned}
\mathbf{D}_k^{11} &= \mathbb{E} \left\{ \left[\nabla_{\mathbf{x}_k} \log p(\mathbf{x}_{k+1}|\mathbf{x}_k) \right] \left[\nabla_{\mathbf{x}_k} \log p(\mathbf{x}_{k+1}|\mathbf{x}_k) \right]^\top \right\} \\
&= \mathbb{E} \left\{ \left[\nabla_{\mathbf{x}_k} \mathbf{G}_k^\top(\mathbf{x}_k) \right] \mathbf{Q}_{\mathbf{x}_k}^{-1} \left[\nabla_{\mathbf{x}_k} \mathbf{G}_k^\top(\mathbf{x}_k) \right] \right\} \\
&= \mathbb{E} \left\{ \mathbf{G}_k^\top \mathbf{Q}_k^{-1} \mathbf{G}_k \right\} && \text{(E.14)} \\
\mathbf{D}_k^{12} &= -\mathbb{E} \left\{ \nabla_{\mathbf{x}_{k+1}} \left[\nabla_{\mathbf{x}_k} \log p(\mathbf{x}_{k+1}|\mathbf{x}_k) \right]^\top \right\} \\
&= -\mathbb{E} \left\{ \mathbf{G}_k^\top \right\} \mathbf{Q}_{\mathbf{x}_k}^{-1} && \text{(E.15)}
\end{aligned}$$

$$\begin{aligned}
\mathbf{D}_k^{22} &= -\mathbb{E} \left\{ \nabla_{\mathbf{x}_{k+1}} \left[\nabla_{\mathbf{x}_{k+1}} \log p(\mathbf{x}_{k+1}|\mathbf{x}_k) \right]^\top \right\} - \mathbb{E} \left\{ \nabla_{\mathbf{x}_{k+1}} \left[\nabla_{\mathbf{x}_{k+1}} \log p(\mathbf{z}_{k+1}|\mathbf{x}_{k+1}) \right]^\top \right\} \\
&= -\mathbb{E} \left\{ \nabla_{\mathbf{x}_{k+1}} \left[\nabla_{\mathbf{x}_{k+1}} \mathbf{G}_k^\top(\mathbf{x}_k) \right] \mathbf{Q}_{\boldsymbol{x}_k}^{-1} \left[\mathbf{x}_{k+1} - \mathbf{G}_k(\mathbf{x}_k) \right]^\top \right\} - \\
&- \mathbb{E} \left\{ \nabla_{\mathbf{x}_{k+1}} \left[\nabla_{\mathbf{x}_{k+1}} \mathbf{G}_{\mathbf{x}_{k+1}}^\top(\mathbf{x}_{k+1}) \right] \mathbf{R}_{\boldsymbol{x}_k}^{-1} \left[\mathbf{z}_{k+1} - \mathbf{G}_{k+1}(\mathbf{x}_{k+1}) \right]^\top \right\} \\
&= -\mathbb{E} \left\{ \mathbf{Q}_{\mathbf{x}_k}^{-1} \right\} - \mathbb{E} \left\{ \left[\nabla_{\mathbf{x}_{k+1}} \mathbf{G}_{k+1}^{-1}(\mathbf{x}_{k+1}) \right] \mathbf{R}_{\mathbf{x}_{k+1}}^{-1} \left[-\nabla_{\mathbf{x}_{k+1}} \mathbf{G}_{k+1}^\top(\mathbf{x}_{k+1}) \right] \right\} \\
&= -\mathbf{Q}_{\mathbf{x}_k}^{-1} + \mathbb{E} \left\{ \mathbf{G}_{k+1}^\top \mathbf{R}_{\mathbf{x}_{k+1}}^{-1} \mathbf{G}_{k+1} \right\} && \text{(E.16)}
\end{aligned}$$

where

$$\mathbf{G}_{k+1} = \left[\nabla_{\mathbf{x}_{k+1}} \mathbf{g}_{k+1}^\top(\mathbf{x}_{k+1}) \right]^\top \tag{E.17}$$

Jacobian matrices are independent of the target state, so $\mathbf{D}_k^{11}$, $\mathbf{D}_k^{12}$ and $\mathbf{D}_k^{22}$ are deterministic and the $\mathbb{E}\{\cdot\}$ is dropped out resulting in:

$$\begin{aligned}
\mathbf{D}_k^{11} &= \mathbf{A}_x^\top \mathbf{Q}_{\mathbf{x}_k}^{-1} \mathbf{A}_x \\
\mathbf{D}_k^{12} &= -\mathbf{A}_\mathbf{x}^\top \mathbf{Q}_{\mathbf{x}_k}^{-1} \\
\mathbf{D}_k^{22} &= \mathbf{Q}_{\mathbf{x}_k}^{-1} + \mathbf{G}_{k+1}^\top \mathbf{R}_{\mathbf{x}_{k+1}}^{-1} \mathbf{G}_{k+1} \\
\mathbf{D}_k^{21} &= \left[\mathbf{D}_k^{12}\right]^\top
\end{aligned} \tag{E.18}$$

Then, using Equation E.6, we have:

$$\begin{aligned}
\boldsymbol{\mathcal{I}}_{k+1} &= \mathbf{D}_k^{22} - \mathbf{D}_k^{21} (\boldsymbol{\mathcal{I}}_k + \mathbf{D}_k^{11})^{-1} \mathbf{D}_k^{12} \\
&= \mathbf{Q}_{\mathbf{x}_k}^{-1} + \mathbf{G}_{k+1}^\top \mathbf{R}_{\mathbf{x}_{k+1}}^{-1} \mathbf{G}_{k+1} - \mathbf{A}_\mathbf{x} \mathbf{Q}_{\mathbf{x}_k}^{-1} \left(\boldsymbol{\mathcal{I}}_k + \mathbf{A}_\mathbf{x}^\top \mathbf{Q}_{\mathbf{x}_k}^{-1} \mathbf{A}_\mathbf{x}\right)^{-1} \mathbf{A}_\mathbf{x}^\top \mathbf{Q}_{\mathbf{x}_k}^{-1}
\end{aligned} \tag{E.19}$$

Using the matrix inversion lemma:

$$\begin{aligned}
\boldsymbol{\mathcal{I}}_{k+1} &= \mathbf{Q}_{\mathbf{x}_k}^{-1} + \mathbf{G}_{k+1}^\top \mathbf{R}_{\mathbf{x}_{k+1}}^{-1} \mathbf{G}_{k+1} - \mathbf{A}_\mathbf{x} \mathbf{Q}_{\mathbf{x}_k}^{-1} \left[\boldsymbol{\mathcal{I}}_k^{-1} - \boldsymbol{\mathcal{I}}_k^{-1} \mathbf{A}_\mathbf{x}^\top \left(\mathbf{Q}_{\mathbf{x}_k} + \mathbf{A}_\mathbf{x} \boldsymbol{\mathcal{I}}_{k-1} \mathbf{A}_\mathbf{x}^\top\right)\right]^{-1} \mathbf{A}_\mathbf{x} \mathbf{Q}_{\mathbf{x}_k}^{-1} \\
&= \boldsymbol{Q}_{\mathbf{x}_k}^{-1} + \mathbf{G}_{k+1}^\top \mathbf{R}_{\mathbf{x}_{k+1}}^{-1} \mathbf{G}_{k+1} - \mathbf{A}_\mathbf{x} \mathbf{Q}_{\mathbf{x}_k}^{-1} \left[\boldsymbol{\mathcal{I}}_k^{-1} - \boldsymbol{\mathcal{I}}_k^{-1} \mathbf{A}_\mathbf{x}^\top \mathbf{Q}_{\mathbf{x}_k} - \boldsymbol{\mathcal{I}}_k^{-1} \mathbf{A}_\mathbf{x} \boldsymbol{\mathcal{I}}_k^{-1} \mathbf{A}_\mathbf{x}^\top\right]^{-1} \\
&= \mathbf{Q}_{\mathbf{x}_k} + \mathbf{G}_{k+1}^\top \mathbf{R}_{\mathbf{x}_{k+1}}^{-1} \mathbf{G}_{k+1} - \mathbf{A}_\mathbf{x} \mathbf{Q}_{\mathbf{x}_k}^{-1} \boldsymbol{\mathcal{I}}_k^{-2} + \mathbf{A}_\mathbf{x} \mathbf{Q}_{\mathbf{x}_k}^{-1} \boldsymbol{\mathcal{I}}_k^{-2} + \boldsymbol{\mathcal{I}}_k^{-2} \mathbf{A}_\mathbf{x} \mathbf{A}_\mathbf{x}^\top,
\end{aligned} \tag{E.20}$$

and finally we obtain the form of recursive FIM $\boldsymbol{\mathcal{I}}_{k+1}$ matrix to be implemented:

$$\boldsymbol{\mathcal{I}}_{k+1} \left(\mathbf{Q}_{\mathbf{x}_k} + \mathbf{A}_\mathbf{x} \boldsymbol{\mathcal{I}}_k^{-1} \mathbf{A}_\mathbf{x}^\top\right)^{-1} - \mathbf{G}_{k+1}^\top \mathbf{R}_{\mathbf{x}_{k+1}}^{-1} \mathbf{G}_{k+1}. \tag{E.21}$$

Bibliography

[1] Gints Jekabsons, Vadim Kairish, and Vadim Zuravlyov. An Analysis of Wi-Fi Based Indoor Positioning Accuracy. *J. Riga Technical University*, 44:131–137, 2011.

[2] Cyrill Stachniss. Robot Mapping Extended Kalman Filter. *Autonomous Intelligent Systems*, 2016.

[3] Giacomo Calanchi. Design and test of a prototype for indoor positioning. Master's thesis, University of Bologna, 2013.

[4] P. Closas. *Bayesian Signal Processing Techniques for GNSS Receivers: from multipath mitigation to positioning*. PhD thesis, Universitat Politècnica de Catalunya, June 2009.

[5] J. Manuel Castro-Arvizu, J. Vilà-Valls, Pau Closas, and J. A. Fernández-Rubio. Simultaneous tracking and RSS model calibration by robust filtering. In *Asilomar Conference on Signals, Systems and Computers*, Pacific Grove CA, Nov 2014.

[6] T. Paul and T. Ogunfunmi. Wireless LAN Comes of Age: Understanding the IEEE 802.11n Amendment. *Circuits and Systems Magazine, IEEE*, 8(1):28–54, 2008.

[7] J. M. Castro-Arvizu, M.Sanchez Meraz, and A. J. A. Cruz. GNSS Receiver Based on a SDR Architecture Using FPGA Devices. In *Electronics, Robotics and Automotive Mechanics Conference (CERMA)*, pages 383–388, Cuernavaca, Morelos, Mexico, Nov 2011.

[8] James J. Parkinson and Spilker Jr. *Global Positioning System: Theory and Applications*. AIAA, 1996.

[9] E. D. Kaplan. *Understanding GPS. Principles and Applications*. Artech House Publishers, 1996.

[10] E. Dominguezand C. Moriana, L. Bonardi, E. Aguado, D. Lowe, M. Pattinson, M. Hutchinson, G. Seco-Granados, J. A. Lopez-Salcedo, D. Egea-Roca, D. Naberezhnykh, F. Dovis, J. P. Boyero, and I. Fernández-Hernández. Vehicular and Pedestrian GNSS Integrity Algorithms and Results for Urban and Road environments developed after an Extensive Real Data Collection Campaign. Tampa, Florida, September 16, 2015.

[11] M. S. Grewal, L. R. Weill, and A. P. Andrews. *Global Positioning Systems, Inertial Navigation, and Integration*. John Wiley & Sons, Inc., 2nd edition, 2007.

[12] D. Dardari, P. Closas, and P.M. Djuric. Indoor tracking: Theory, methods, and technologies. *Vehicular Technology, IEEE Transactions on*, 64(4):1263–1278, April 2015.

[13] Cesar Enriquez-Caldera Munoz David Lara, Frantz Bouchereau Vargas. *Position Location Techniques and Applications*. Academic Press, 2009.

[14] Anup Dhital, Pau Closas, and Carles Fernández-Prades. Bayesian filtering for indoor localization and tracking in wireless sensor networks. *EURASIP Journal on Wireless Communications and Networking*, 2012(1):21+, 2012.

[15] K. El-Kafrawy, M. Youssef, A. El-Keyi, and A. Naguib. Propagation Modeling for Accurate Indoor WLAN RSS-Based Localization. In *Vehicular Technology Conference Fall (VTC 2010-Fall), 2010 IEEE 72nd*, pages 1–5, Ottawa, ON, Canada, 2010.

[16] P., P. Bahl, and Padmanabhan. RADAR: an in-building RF-based user location and tracking system. In *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 2, pages 775–784 vol.2, Tel Aviv, Israel, 2000.

[17] Sinan Gezici. A Survey on Wireless Position Estimation. *Wireless Personal Communications*, 44(3):263–282, February 2008.

[18] A. Salazar-Palma Sarkar, T.K. Zhong Ji Kyungjung Kim Medouri. A survey of various propagation models for mobile communication. *Antennas and Propagation Magazine, IEEE*, 45(3):51–82, June 2003.

[19] David I. Laurenson. *Indoor Radio Channel Propagation Modelling by Ray Tracing Techniques*. PhD dissertation, University of Edinburgh, 1994.

[20] T. Jamsa, T. Poutanen, and J. Meinila. Implementation Techniques of Broadband Radio Channel Simulators. In *Vehicular Technology Conference, 2001. VTC 2001 Spring. IEEE VTS 53rd*, volume 1, pages 433–437 vol.1, Rhodes, Greece, 2001.

[21] M.A. Assad, M. Heidari, and K. Pahlavan. Effects of Channel Modeling on Performance Evaluation of WiFi RFID Localization Using a Laboratory Testbed. In *Global Telecommunications Conference, 2007. GLOBECOM '07. IEEE*, pages 366–370, Washington, D.C., 2007.

[22] Theodore Rappaport. *Wireless Communications: Principles and Practice*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2nd edition, 2001.

[23] A. Pahlavan K. Hatami. A comparative performance evaluation of RSS-based positioning algorithms used in WLAN networks. In *Wireless Communications and Networking Conference, 2005 IEEE*, volume 4, pages 2331–2337 Vol. 4, New Orleans, LA, March 2005.

[24] Fink A. and Beikirch H. Hybrid indoor tracking with Bayesian sensor fusion of RF localization and inertial navigation. In *Intelligent Data Acquisition and Advanced Computing Systems (IDAACS), 2011 IEEE 6th International Conference on*, volume 2, pages 823–827, Prague, Czech Republic, Sept 2011.

[25] Prashant Krishnamurthyin Proc. NSF Workshop. Position location in mobile environments. *Context-Aware Mobile Database Management (CAMM)*, 2002.

[26] K. Pahlavan, P. Krishnamurthy, and A. Beneat. Wideband radio propagation modeling for indoor geolocation applications. *IEEE Communications Magazine*, 36(4):60–65, Apr 1998.

[27] I. Guvenc and C. C. Chong. A Survey on TOA Based Wireless Localization and NLOS Mitigation Techniques. *IEEE Communications Surveys Tutorials*, 11(3):107–124, rd 2009.

[28] Hui Liu, H. Darabi, P. Banerjee, and Jing Liu. Survey of Wireless Indoor Positioning Techniques and Systems. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 37(6):1067–1080, 2007.

[29] Jeffrey Hightower and Gaetano Borriello. Location Sensing Techniques. UW CSE 01-07-01, University of Washington, Department of Computer Science and Engineering, Seattle, WA, July 2001.

[30] N. Patwari, J. N. Ash, S. Kyperountas, A. O. Hero, R. L. Moses, and N. S. Correal. Locating the nodes: cooperative localization in wireless sensor networks. *IEEE Signal Processing Magazine*, 22(4):54–69, July 2005.

[31] N. Patwari, A.O. Hero, M. Perkins, N.S. Correal, and R.J. O'Dea. Relative Location Estimation in Wireless Sensor Networks. *Signal Processing, IEEE Transactions on*, 51(8):2137–2148, 2003.

[32] B. Roberts and Pahlavan. Site-specific RSS signature modeling for WiFi localization. In *Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE*, pages 1–6, Honolulu, Hawaii, Nov 2009.

[33] Feng Yin, C. Fritsche, F. Gustafsson, and AM. Zoubir. Received signal strength-based joint parameter estimation algorithm for robust geolocation in LOS/NLOS environments. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 6471–6475, Vancouver, BC, Canada, May 2013.

[34] Feng Yin and C Gustafsson Fritsche. EM- and JMAP-ML Based joint estimation algorithms for robust wireless geolocation in mixed LOS/NLOS environments. *Signal Processing, IEEE Transactions on*, 62(1):168–182, Jan 2014.

[35] A. Ward, A. Jones, and A. Hopper. A new location technique for the active office. *IEEE Personal Communications*, 4(5):42–47, Oct 1997.

[36] Y. Ben-Shimol and N. Blaunstein. Localization and positioning of any subscriber in indoor environments on the basis of analysis of joint AOA-TOA signal distribution. In *Antennas and Propagation in Wireless Communications (APWC), 2011 IEEE-APS Topical Conference on*, pages 1420–1423, Turin, Italy, Sept 2011.

[37] D. Niculescu and Badri Nath. Ad hoc positioning system (APS) using AOA. In *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, volume 3, pages 1734–1743 vol.3, San Francisco CA, March 2003.

[38] H. Liu, H. Darabi, P. Banerjee, and J. Liu. Survey of Wireless Indoor Positioning Techniques and Systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 37(6):1067–1080, Nov 2007.

[39] Li Xinrong, Pahlavan K., Latva aho M., and Ylianttila M. Comparison of indoor geolocation methods in DSSS and OFDM wireless LAN systems. In *Vehicular Technology Conference, 2000. IEEE VTS-Fall VTC 2000. 52nd*, volume 6, pages 3015–3020 vol.6, Boston, MA, 2000.

[40] R. Zemek, S. Hara, K. Yanagihara, and K. Kitayama. A Joint Estimation of Target Location and Channel Model Parameters in an IEEE 802.15.4-based Wireless Sensor Network. In *Personal, Indoor and Mobile Radio Communications, 2007. PIMRC 2007. IEEE 18th International Symposium on*, pages 1–5, Athens, Greece, Sept 2007.

[41] V., L. Erceg, and Schumacher. TGn channel models. IEEE 802.11 Wireless LANs Document. *IEEE 802.11-03/940r4*, May, 2004.

[42] A.J. Coulson, A.G. Williamson, and R.G. Vaughan. A Statistical Basis for Lognormal Shadowing Effects in Multipath Fading Channels. *Communications, IEEE Transactions on*, 46(4):494–502, 1998.

[43] Ana Moragrega. *Optimization of Positioning Capabilities in Wireless Sensor Networks: from power efficiency to medium access.* PhD thesis, Universitat Politecnica de Catalunya, 2016.

[44] Kaemarungsi Kamol and Krishnamurthy Prashant. Analysis of WLAN's Received Signal Strength Indication for Indoor Location Fingerprinting. *Pervasive Mob. Comput.*, 8(2):292–316, April 2012.

[45] Nu no Barrau Galo and Páz-Borrallo José M. A New Location Estimation System for Wireless Networks Based on Linear Discriminant Functions and Hidden Markov Models. *EURASIP J. Appl. Signal Process.*, 2006:159–159, January 2006.

[46] A. Smailagic and D. Kogan. Location sensing and privacy in a context-aware computing environment. *IEEE Wireless Communications*, 9(5):10–17, Oct 2002.

[47] Paul Castro, Patrick Chiu, Ted Kremenek, and Richard R. Muntz. A Probabilistic Room Location Service for Wireless Networked Environments. In *Proceedings of the 3rd International Conference on Ubiquitous Computing*, UbiComp '01, pages 18–34, London, UK, UK, 2001. Springer-Verlag.

[48] Jouni Hartikainen, Arno Solin, and Simo Särkkä. Optimal Filtering with Kalman Filters and Smoothers, 2011.

[49] Gustafsson F., Gunnarsson F., Bergman Niclas, Forssell U., and Jansson J. Particle filters for positioning, navigation, and tracking. *Signal Processing, IEEE Transactions on*, 50(2):425–437, Feb 2002.

[50] Alireza Toloei and Saeid Niazi. State Estimation for Target Tracking Problems with Nonlinear Kalman Filter Algorithm. *International Journal of Computer Applications IJCA*, 2014.

[51] R. Faragher. Understanding the Basis of the Kalman Filter Via a Simple and Intuitive Derivation [Lecture Notes]. *Signal Processing Magazine, IEEE*, 29(5):128–132, 2012.

[52] I. Guvenc. Enhancements to RSS based indoor tracking systems using kalman filters. In *In GSPx International Signal Processing Conference*, 2003.

[53] Sangwoo Lee, Bongkwan Cho, Bonhyun Koo, Sanghwan Ryu, Jaehoon Choi, and Sunwoo Kim. Kalman Filter-based Indoor Position Tracking with Self-calibration for RSS Variation Mitigation. *Int. J. Distrib. Sen. Netw.*, 2015:10:10–10:10, January 2015.

[54] M.A. Caceres, F. Sottile, and Spirito. Adaptive location tracking by Kalman Filter in Wireless Sensor Networks. In *Wireless and Mobile Computing, Networking and Communications, 2009. WIMOB 2009. IEEE International Conference on*, pages 123–128, Marrakech, Morocco, Oct 2009.

[55] Wendong Xiao, Lihua Xie, Jianfeng Chen, and L. Shue. Multi-Step Adaptive Sensor Scheduling for Target Tracking in Wireless Sensor Networks. In *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, volume 4, pages IV–IV, Toulouse, France, May 2006.

[56] Y. Liu and Z. Sun. EKF-Based Adaptive Sensor Scheduling for Target Tracking. In *2008 International Symposium on Information Science and Engineering*, volume 2, pages 171–174, Pudong District, Shanghai, Dec 2008.

[57] X. Wang, M. Fu, and H. Zhang. Target Tracking in Wireless Sensor Networks Based on the Combination of KF and MLE Using Distance Measurements. *IEEE Transactions on Mobile Computing*, 11(4):567–576, April 2012.

[58] X. Hu, B. Xu, and Y. H. Hu. Target tracking with distance-dependent measurement noise in wireless sensor networks. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 5200–5203, Vancouver, BC, Canada, May 2013.

[59] H. Durrant-Whyte and T. Bailey. Simultaneous localization and mapping: part I. *IEEE Robotics Automation Magazine*, 13(2):99–110, June 2006.

[60] T. Bailey and H. Durrant-Whyte. Simultaneous localization and mapping (SLAM): part II. *IEEE Robotics Automation Magazine*, 13(3):108–117, Sept 2006.

[61] Branko Ristic, Sanjeev Arulampalam, and NeilGordon. *Beyond the Kalman filter : particle filters for tracking applications*. Artech House, Boston, London, 2004.

[62] Bar-Shalom Yaakov, Kirubarajan Thiagalingam, and Li X.-Rong. *Estimation with Applications to Tracking and Navigation*. John Wiley & Sons, Inc., New York, NY, USA, 2002.

[63] C. Chui and G. Chen. *Kalman Filtering with Real-time Applications*. Springer-Verlag New York, Inc., New York, NY, USA, 1987.

[64] Yaakov Bar-Shalom. Estimation and Tracking: Principles Techniques and software. *Artech House*, 1993.

[65] E. Mazor, A. Averbuch, Y. Bar-Shalom, and J. Dayan. Interacting multiple model methods in target tracking: a survey. *Aerospace and Electronic Systems, IEEE Transactions on*, 34(1):103–123, Jan 1998.

[66] H. A. P. Blom and Y. Bar-Shalom. The interacting multiple model algorithm for systems with Markovian switching coefficients. *IEEE Transactions on Automatic Control*, 33(8):780–783, Aug 1988.

[67] L. A. Johnston and V. Krishnamurthy. An improvement to the interacting multiple model (IMM) algorithm. *IEEE Transactions on Signal Processing*, 49(12):2909–2923, Dec 2001.

[68] X. Rong Li and V. P. Jilkov. Survey of maneuvering target tracking. part v. multiple-model methods. *IEEE Transactions on Aerospace and Electronic Systems*, 41(4):1255–1321, Oct 2005.

[69] El-Sheimy N. and Niu X. The Promise of MEMS to the Navigation Community. *InsideGNSS*, pages 6+, April 2007.

[70] W. Chai, C. Chen, E. Edwan, J. Zhang, and O. Loffeld. INS/Wi-Fi based indoor navigation using adaptive Kalman filtering and vehicle constraints. In *Positioning Navigation and Communication (WPNC), 2012 9th Workshop on*, pages 36–41, Dresden, Germany, March 2012.

[71] Adrian Schumacher. Integration of a gps aided strapdown inertial navigation system for land vehicles, 2006.

[72] Honghui Qi and J.B. Moore. Direct Kalman filtering approach for GPS/INS integration. *Aerospace and Electronic Systems, IEEE Transactions on*, 38(2):687–693, Apr 2002.

[73] J. Zhou. *Low-cost MEMS-INS/GPS Integration using Nonlinear Filtering Approaches*. Fakultät IV: Naturwissenschaftlich-Technische Fakultät. University of Siegen, 2013.

[74] Liu H.H.S. and Pang G.K.H. Accelerometer for mobile robot positioning. *Industry Applications, IEEE Transactions on*, 37(3):812–819, May 2001.

[75] Warren S. Flenniken IV. *Modeling Inertial Measurement Units and Analizing the Effect of their effect of their errors in Navigation Applications*. PhD thesis, University of Auburn, 2005.

[76] Raspberry Pi Foundation. https://www.raspberrypi.org. Technical report.

[77] TP-Link User guide. http://www.tp-link.es/resources/. Technical report.

[78] Fisher R. A. n the Mathematical Foundations of Theoretical Statistics. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 222(594-604):309–368, 1922.

[79] R. A. Fisher. Theory of statistical estimation. *Mathematical Proceedings of the Cambridge Philosophical Society*, 22:700–725, 7 1925.

[80] Petr Tichavsky, Carlos H. Muravchik, and Arye Nehorai. Posterior Cramer-Rao bounds for discrete-time nonlinear filtering. *Signal Processing, IEEE Transactions on*, 46(5):1386–1396, May 1998.

[81] J.M. Bernardo. *Bayesian Statistics*. University of Valencia, 46100, Burjassot, Valencia, Spain, November 2001.

[82] Andrew Thomas David Spiegelhalter. *WinBUGS User Manual*. Department of Epidemiology & Public Health, Imperial College School of Medicine, Norfolk Place, London W2 1PG, UK, 1.4 edition, January 2003.

[83] A. F. M. Smith and G. O. Robert. Bayesian computation via the Gibbs sampler and related Markov Chain Monte Carlo Methods. *Journal of the Royal Statistical Society. Series B (Methodological), Vol. 55, No. 1*, pages 3–23, 1993.

[84] Geman Stuart and Geman Donald. Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 6(6):721–741, November 1984.

[85] N. Kantas, A. Doucet, S.S. Singh, and J.M Maciejowski. An Overview of Sequential Monte Carlo Methods for Parameter Estimation. In *15th IFAC Symposium o System Identification (SYSID)*, 2009.

[86] Metropolis N., Rosenbluth A. W., and Rosenbluth. Equation of State Calculations by Fast Computing Machines. *Journal of Chemical Physics 21*, 21:1087–1092, 1953.

[87] Hastings W. K. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

[88] Musa Zeytinci, Veli Sari, Frederic Harmanci, Emin Anarim, and Mehmet Akar. Location estimation using RSS measurements with unknown path loss exponents. *EURASIP J. Wireless Comm. and Networking*, 2013:178, 2013.

[89] T. Roos, P. Myllymaki, and H. Tirri. A statistical modeling approach to location estimation. *Mobile Computing, IEEE Transactions on*, 1(1):59–69, Jan 2002.

[90] Xinrong Li. RSS-Based Location Estimation with Unknown Pathloss Model. *Wireless Communications, IEEE Transactions on*, 5(12):3626–3633, December 2006.

[91] Santiago Mazuelas, Alfonso Bahillo, Ruben M. Lorenzo, Patricia Fernandez, and Francisco A. Lago. Robust Indoor Positioning Provided by Real-Time RSSI Values in Unmodified WLAN Networks. *Selected Topics in Signal Processing, IEEE Journal of*, 3(5):821–831, Oct 2009.

[92] Hing Cheung So and Lanxin Lin. Linear Least Squares Approach for Accurate Received Signal Strength Based Source Localization. *Signal Processing, IEEE Transactions on*, 59(8):4035–4040, Aug 2011.

[93] Ahmad Hatami. *Application of Channel Modeling for Indoor Localization Using TOA and RSS*. PhD dissertation, Worcester Polytechnic Institute, May 2006.

[94] S. Chessa G. Barsocchi, P. Lenzi. Virtual calibration for RSSI-based indoor localization with IEEE 802.15.4. In *Communications, 2009. ICC '09. IEEE International Conference on*, pages 1–5, Dresden, Germany, June 2009.

[95] Stefano Chessa Francesco Barsocchi, Paolo Lenzi. Automatic virtual calibration of range-based indoor localization systems. *Wirel. Commun. Mob. Comput.*, 12(17):1546–1557, December 2012.

[96] J., O. Vallet, and J.and Saarinen Kaltiokallio, M.and Myrsky. Simultaneous RSS-based localization and model calibration in wireless networks with a mobile robot. *Procedia Computer Science*, 10(0):1106 – 1113, 2012. {ANT} 2012 and MobiWIS 2012.

[97] ISO-IEC 24730-1. Real time locating systems. part 1 application program interface (APi). 2006.

[98] F., F. Capulli, C. Monti, and Vari. Path loss models for IEEE 802.11a wireless local area networks. In *Wireless Communication Systems, 2006. ISWCS '06. 3rd International Symposium on*, pages 621–624, Valencia, Spain, Sept 2006.

[99] A.R. Sandeep, Y. Shreyas, Shivam Seth, Rajat Agarwal, and G. Sadashivappa. Wireless Network Visualization and Indoor Empirical Propagation Model for a Campus Wi-Fi Network. *World Academy of Science, Engineering and Technology*, 18(6):705 – 710, 2008.

[100] Guoqiang Mao, Brian D. O. Anderson, and Barış Fidan. Path loss exponent estimation for wireless sensor network localization. *Comput. Netw.*, 51(10):2467–2483, July 2007.

[101] A.H.M, N. Salman, M. Ghogho, and Kem. On the joint estimation of the RSS-based location and path-loss exponent. *Wireless Communications Letters, IEEE*, 1(1):34–37, February 2012.

[102] H. Hashemi. The Indoor Radio Propagation Channel. *Proceedings of the IEEE*, 81(7):943–968, 1993.

[103] Yanwei Wang Neal Patwari, Robert ODea. Relative location in wireless networks. *Vehicular Technology Conference,IEEE*, 2001.

[104] R.W. Ouyang, A.K.-S. Wong , and Chin-Tau Lea. Received Signal Strength-Based Wireless Localization via Semidefinite Programming: Noncooperative and Cooperative Schemes. *IEEE Transactions on Vehicular Technology*, 59(3):1307–1318, March 2010.

[105] Rao C. Radhakrishna. Minimum variance and the estimation of several parameters. *Mathematical Proceedings of the Cambridge Philosophical Society*, 43:280–283, 4 1947.

[106] Steven M. Kay. *Fundamentals of Statistical Signal Processing: Estimation Theory*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1993.

[107] Van Trees and Harry L. *Detection, estimation, and modulation theory. 1. , detection, estimation, and linear modulation theory*. J. Wiley and sons, New York, Chichester, 1968.

[108] Alan E. Gelfand, Susan E. Hills , and Amy Racine-Poon. Illustration of Bayesian Inference in Normal Data Models Using Gibbs Sampling. *Journal of the American Statistical Association*, 85(412):972–985, December 1990.

[109] Thomas B. Schön, Fredrik Lindsten, Johan Dahlinand Wågberg, and Christian A. Naesseth. Sequential Monte Carlo Methods for System Identification, March 2015.

[110] UK MRC Biostatistics Unit, Cambridge. The BUGS Project. WinBUGS 1.4.3, http://www.mrc-bsu.cam.ac.uk/bugs/.

[111] David D. Sworder, John E. Boyd, and Robert J. Elliott. Modal Estimation in Hybrid Systems. *Journal of Mathematical Analysis and Applications*, 245(1):225 – 247, 2000.

[112] J. Manuel Castro-Arvizu, Pau Closas, and J. A. Fernández-Rubio. Cramér-Rao lower bound for breakpoint distance estimation in a path-loss model. In *in Proceedings of IEEE International Conference on Communications (IEEE ICC 2014), 10-14 June 2014,*, Sydney (Australia), jun 2014.

[113] J. Manuel Castro-Arvizu, Ana Moragrega, Pau Closas, and Juan A Fernandez-Rubio. Assessment of rss model calibration with real wlan devices. In *The Twelfth International Symposium on Wireless Communication Systems*, Bruselles, Belgium, Aug 2015.