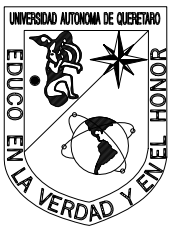
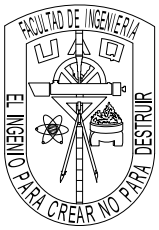


Universidad Autónoma de Querétaro
Facultad de Ingeniería



Design and Implementation of an Ubiquitous Positioning System Based on GNSS Pseudoranges and Assisted With WLAN-RSSI

Philipp Richter

A dissertation submitted for the degree of
Doctor of Engineering

Santiago de Querétaro 2016

This dissertation is the result of my own work and includes nothing that is the outcome of work done in collaboration except where specifically indicated in the text.

No part of this dissertation has already been, or is currently being submitted by the author for any other degree or diploma or other qualification.

This work was supported by the Consejo Nacional de Ciencia y Tecnología (México), CONACYT.

Typset in LaTeX

Universidad Autónoma de Querétaro
Facultad de Ingeniería
Doctorado en Ingeniería

Design and Implementation of an Ubiquitous Positioning System Based on GNSS Pseudoranges and Assisted With WLAN-RSSI

A dissertation submitted for the degree of Doctor of Engineering

by

Philipp Richter

supervised by Dr. Manuel Toledano Ayala

Dr. Manuel Toledano Ayala
(Presidente)

Dr. Eduardo Castaño Tostado
(Secretario)

Dr. Edgar Rivas Araiza
(Vocal)

Dr. Jair García Lamont
(Suplente)

Dr. Genaro Soto Zarazúa
(Suplente)

Dr. Aurelio Domínguez González
(Director de la Facultad)

Dra. Ma. Guadalupe Flavia Loarca Piña
(Directora de Investigación y Posgrado)

Centro Universitario
Santiago de Querétaro, Qro

September 2016

To Sophia and Dalia

Abstract

Already today numerous applications, such as tracking people or assets, navigation and geofencing for people or robots, intelligent transport systems and location sensitive marketing and billing, require location information. The Internet-of-Things will further increase the amount of location-based services (LBSs) and also the demanded accuracy will increase. Outdoors, global navigation satellite systems (GNSSs) provide reliable location information. Other technologies such as pervasive computing systems, wireless communication networks, miniaturised sensors, and so forth enable the localisation of an object or person in indoor environments. Many systems that enable indoor localisation require particular infrastructure or a multitude of sensors; especially if the aim is ubiquitous and seamless localisation. Taking into consideration the use of available infrastructure, the potential accuracy and precision and the error sources, GNSS and wireless local area network (WLAN) location fingerprinting with received signal strength indicator (RSSI) are the most promising technologies to achieve ubiquitous and seamless indoor/outdoor localisation.

This thesis proposes a method to fuse GNSS pseudoranges and WLAN RSSI in order to yield a general, ubiquitous positioning system that is likewise accurate indoors, outdoors and in the transition zones. We use the recursive Bayesian estimation framework to tightly integrate global positioning system (GPS) pseudoranges with WLAN RSSI. We present a state space model that relies on statistical models for the object's/person's motion and for the pseudorange and RSSI observations. This study addresses the fundamental issue of different state space of measurements: pseudoranges reside on a spatially continuous state space and RSSI on a spatially discrete space.

To overcome this problem, Gaussian process regression for WLAN RSSI is revised and used to interpolate RSSIs on space. Thus, we yield a continuous RSSI model facilitating the accurate data fusion with GPS pseudoranges. To find and

propose the best suited Gaussian process regression model for RSSI, we revive the discussion about RSSI distributions and explore and assess several Gaussian process models in depth. It was also hypothesised whether different models for indoor and outdoor environments would improve the systems localisation performance. The model for the GPS observations is based on the well-known pseudorange model.

Once developed the observation models, a particle filter is presented that integrates the Gaussian process based likelihood function and fuses the two measurements. The filter deals intuitively with issues as the weighting of the two sources of information, the availability of less than four pseudoranges and the spatial limitation of the fingerprinting radio map.

To demonstrate the effectiveness of the proposed algorithm we developed software to record real world data, regarding synchronisation and the different coordinate systems and projections. In experiments, conducted in an environment challenging for GNSS and WLAN fingerprinting localisation using off-the-shelf sensors, we achieved, accurate and robust seamless localisation with an average accuracy of 5 m.

Resumen

Ya en la actualidad numerosas aplicaciones, como el rastreo de personas o bienes, navegación o geovallado para personas o robots, sistemas de transporte inteligente, mercadotecnia o facturación consciente de la ubicación. El Internet de las Cosas aumentará aún más la cantidad de los servicios basados en la posición y también aumentará la exactitud exigida. Al exterior los sistemas globales de navegación por satélite proveen información de la ubicación fiable. Tecnologías como sistemas de computación ubicua, redes inalámbrica de comunicación, sensores miniaturizados, entre otras, pueden determinar la ubicación de personas u objetos en interiores. Muchos sistemas de localización en interiores necesitan infraestructura particular o una multitud de sensores; especialmente si se busca la localización ubicua y continua. Considerando la disponibilidad de infraestructura, el potencial de exactitud y precisión, y las fuentes de errores de las tecnologías, los sistemas globales de navegación por satélite y el fingerprinting en redes de área local inalámbrica con la fuerzas de las señales, son las tecnologías más prometedoras para lograr localización ubicua tanto en interiores como en exteriores.

Esta tesis propone un método en el que combina sistemas globales de navegación por satélite y fuerzas de señales de redes de área local inalámbricas con el fin de lograr una localización general, ubicua, con la misma exactitud y precisión en interiores, exteriores y en las zonas de transición. Se utiliza un filtro Bayesiano recursivo para integrar profundamente pseudodistancias de sistemas globales de navegación por satélite con fuerzas de señales de redes de área local inalámbricas. Se presenta un modelo del espacio de estado basado en modelos estocásticos para el movimiento del objeto/personas y para las pseudodistancias y observaciones de fuerza de señales de redes de área local inalámbricas. Este estudio aborda al problema fundamental que las pseudodistancias están definidas en un espacio espacialmente continuo y las fuerzas de señales de redes de área local inalámbricas

están definidas en un espacio discreto.

Para resolver este problema, retomamos la regresión de procesos Gaussianos. Se aplica a la interpolación de las fuerzas de señales en el espacio para lograr un modelo continuo en el espacio del las fuerzas de señales de redes de área local inalámbricas que facilita la integración con pseudodistancias del sistema de posicionamiento global. Para encontrar y proponer el mejor modelo de procesos Gaussianos para las fuerzas de señales, reactivamos la discusión sobre la distribución de fuerzas de señales de redes de área local inalámbricas y examinamos y evaluamos varios modelos de procesos Gaussianos en detalle. También se planteó la hipótesis si modelos diferente para interiores y exteriores mejoraría la exactitud del sistema de posicionamiento. El modelo de las observaciones del sistema global de posicionamiento se basa en el modelo de pseudodistancias estándar.

Una vez desarrollados los modelos de las observaciones, se presenta un filtro de partículas que integra la función de verosimilitud basada en procesos Gaussianos y combina las dos mediciones. El filtro maneja intuitivamente cuestiones como la ponderación de las dos fuentes de información, la disponibilidad de menos de cuatro pseudodistancias y la limitación espacial de la base de datos de fingerprints de fuerzas de señales.

Para demostrar la eficacia del algoritmo propuesto desarrollamos un software para registrar datos del mundo real, tomando en cuenta la sincronización y las diferentes sistemas de coordenadas y proyecciones. En experimentos, efectuados en ambientes desafiantes para sistemas globales de navegación por satélite y fingerprinting con fuerzas de las señales con sensores comerciales, hemos logrado una exactitud media de 5 m.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1. Motivation

Since thousands of years human beings had the necessity to identify their location and to find their way in order to accomplish the daily tasks. Perhaps the most common use of (geo-)localisation might just be 'finding home'. The ancient skill of navigation was and is indispensable for getting around and exploring new territories. The term navigation is twofold. On the one hand it is understood as determining locations over time, and on the other hand it refers to planning and maintaining of a course from one location to another. Groves (2008) calls these two concepts 'science of navigation' and the 'art of navigation', where the latter may also be described by routing and is not much of concern here. Navigation is intuitively performed by estimating a location over time, relative to known visual references, such as landmarks, waypoints, celestial objects, and so on. This technique of navigation is based upon recognition of known features and is referred to as position fixing. The same method was applied when ships were navigated along shores, using landmarks as the needed reference points. As soon it became necessary to navigate ships distant from shores, references in the landscape were lacking and new navigation methods had to be found. The alternative approach that was found is called dead reckoning. It requires knowledge about an initial position, and also observations of the velocity and direction of motion. The averaged velocity and the heading give the change of position, which is added to the last known position. Repeating this consecutively, the current position can always be derived from the last one and the motion since then (Titterton et al. 2004). Clearly, the key to navigation is the determination of

locations, or positions.

When purely visual observations were replaced by more precise measurements, made with tools like compass, sextants, precise clocks, et cetera, the accuracy of location estimates and thereby the performance of navigators was increased significantly. The next step towards an refined navigation was taken at the beginning of the 20th century, when radio frequency signals were dominated and radio navigation was explored. Also important developments of instruments measuring inertial forces fall into that time and contributed to the that improvement. Again, much more precise and accurate measurements could be taken, leading to better location estimates and hence improved navigation.

The best known positioning system based upon radio frequency signals is the NAVSTAR global positioning system (GPS). It is the standard positioning system of today, and it was declared operational about twenty years ago. With an today accuracy about 10 m of a low priced GPS receiver (Dardari et al. 2012; Hightower et al. 2001), in unobstructed environments even less (FAA 2016; Zandbergen et al. 2011)[1], it fulfils most of the commercial demands for self-localisation and navigation on sea, air and land. (For an extensive study about global navigation satellite system (GNSS) applications and market segments see Kaplan et al. (2006, chap. 12).)

GPS receiver disseminated fast into consumer products and can nowadays be found in an increasing number of vehicles and mobile devices. This in turn, especially the exponentially emerging mobile device market, created a vast demand of additional services using somehow information about the user's whereabouts. For these services the term location-based services (LBSS) was coined. LBSS took the use of location information beyond that of pure localisation and navigation. Numerous applications such as tracking of objects and users, finding points of interests (restaurants, pharmacies), assistance for elderly and challenged persons, sharing of resources by sharing their location, spatially filtered advertising or autonomous vehicles have been explored (Mautz 2012, sec. 1.4). Location information already interferes considerably with the modern daily life, its significance is evident without even considering military applications.

The mentioned applications require not only more accurate and precise location

1. Absolute numbers are hardly comparable because they most likely origin from different accuracy metrics and different measurement conditions. Employing differential positioning, multiple constellations and/or frequencies, carrier-phase measurements, postprocessing, etc. also centimetre accuracy can be achieved. Cost and size constrain their use in consumer grade receivers.

information than current consumer grade GNSS receiver can deliver, but also must be more robust and provide near 100 % availability independent of the environment. For these but also other reason indoor localisation has become a vivid research field. Virtually any signal or sensor technology is employed in indoor positioning systems, ranging from optical, infrared and radio frequency signals over sound and inertial or magnetic measurements to floor tiles (Mautz 2012); systems relying on wireless local area network (WLAN) infrastructure are probably the most prominent.

Problem formulation The largest contributions to GNSS positioning error are shadowing and multipath propagation (Kaplan et al. 2006). Shadowing occurs when the links between the GNSS receiver and the satellite is obstructed. Signal powers are already very low near the earth's surface and after penetrating obstructing objects they are too low to be received by most receivers. Position fixes can usually not be obtained inside buildings. Multipath phenomenon is caused by obstructions of the satellite-receiver link too. But the signal reaches the receiver by indirection, such as reflections at objects, instead of being blocked. These two error sources are dominant in harsh environments such as dense urban areas, tunnels and so forth.

Ubiquitous indoor localisation must still overcome several challenges. Just the diversity of approaches poses a problem, as an optimal, unifying method does not exist. Scalability and accuracy are further problems for many system. The most important issue when it comes to indoor positioning is the initial deployment of the systems (**ymberopoulosLiuEtAl15:Realistic**; Mautz 2012), either hardware and/or software. Because to attain a certain performance this usually has to be done manually.

Albeit GNSS modernisation and augmentation reduced the positioning error considerably and also indoor positioning methods advanced considerably, neither GNSS nor a particular indoor positioning technology alone is able to satisfy all demands of today's and future LBSs. Robust, ubiquitous and seamless localisation, purely relying on GNSS or on any other single indoor localisation system, is still not possible; less when a large number of affordable mobile devices are crucial for the service.

A possibility to further increase the accuracy, precision and robustness of positioning systems and overcoming some flaws coming by design of GNSS, it

is widely suggested to aid GNSS with additional sensor information (Fernández-Prades et al. 2011; Liu et al. 2007; Sun et al. 2005). The integration of different systems is also a common method to improve indoor positioning systems. This approach is also called hybridisation or information/data fusion and investigated in this study.

Many different technologies have been used in combination with GNSS, the most promising rely on inertial measurement units or WLAN infrastructure; both achieve accuracies in the range of GNSS or better and both enable seamless localisation – smooth, continuous localisation indoors and outdoors.

A inertial navigation systems (INS) measures inertial forces over time, from which velocity and attitude are computed. Velocity and attitude permit to update the former position via dead reckoning. INS measure inertial force, which depends on the motion of the moving unit and thus makes them completely self-contained, independent of any reference except an initial position. Inertial sensors are already deployed in many mobile devices so that hardware requirements are not an issue. However, when using mobile devices, due to the dead the reckoning principle the errors accumulate rapidly and the accuracy deteriorates fast. To compensate this error an additional localisation source is usually needed (typically communication- and sensor networks), or the INS system requires the sensor unit to be mounted in a specific position.

WLAN signals have been used in many ways to determine location estimation, fingerprinting is mostly used because it requires no alteration of the system and no knowledge of the access points positions. Fingerprinting has in addition properties that complement GNSS very well, most importantly, it is not prone to multipath propagation. Compared to other communication- and sensor networks WLANS has the advantage that the infrastructure is already widely existing. Especially in urban environments, which are usually harsh for GNSS. An open challenge is of course the high burden of deploying and maintaining the fingerprint database. Nonetheless, in this work we use WLAN fingerprinting to complement GNSS.

1.2. Related work

In the following we present an overview about approaches that combine GNSS with WLAN positioning system (WPS). To improve the positioning performance of

localisation systems there are several different levels on which data can be fused.

First, the most high-level approach is to use a position determined by one system instead the position estimate of the primary system. This increases the availability of the hybrid system compared with the availability of the stand-alone systems.

Second level, positions from two or more positioning systems are combined, for example by taking their average or by applying filter techniques. These two methods address mainly GNSS outages and augment the coverage of the positioning system.

A third group of approaches integrates raw observation data of two or more positioning systems directly. In addition to assist during GNSS outages, this approach can improve the overall location accuracy and precision and has the potential to increase the robustness.

1.2.1. Switching between GNSS and WPS positions

The key idea of positioning systems that fall into this category is to switch between two independent systems. To do so usually a criterion that determines when to switch is needed. Intuitively this could be the availability of one or the other system and indeed this method is quite commonly used; GNSS is used outdoors when usually a position fix is available and WPS is used indoors when no GNSS position fix is available. An other switching criterion is an accuracy metric. These two basic switching criteria may be further controlled by the recognition of a previously visited location or a time constraints that must be fulfilled.

For example, mobile devices may apply these basic methods as indicated by application programming interfaces and their location frameworks (blackberry.com 2013; developer.android.com 2013; developer.apple.com 2013; developer.nokia.com 2012). Also the studies by Anagnostopoulos et al. (2015), Bittins et al. (2011), E et al. (2013), Gallagher et al. (2011), Hansen et al. (2009), Papandrea et al. (2012), Pei et al. (2009), Reyero et al. (2008), Wang et al. (2012) and Yang et al. (2009) use these ideas in one form or an other.

1.2.2. Weighting of GNSS and WPS positions

This section summarises GNSS-WPS hybridisation methods that combine independently determined location estimates to a common solution. Position estimates

are combined linearly by either simple weighting methods, such as the weighted average, or by adaptive filters.

When simple weighting methods are used, the weights have to be determined in an additional step. Very diverse information, such as the weather conditions, the availability of GNSS position fix, the number of GNSS satellites in view, the signal-to-noise ratio, sample statistics, and so forth is employed (Li, Tan et al. 2011; Singh et al. 2004; Yeh, Hsu and Chiou 2010; Yeh, Hsu, Su et al. 2009).

The standard adaptive filter for data fusion is the Kalman filter (KF), though its variants and other approximations to the sequential Bayesian estimation framework are feasable. The weights are commonly derived from some accuracy measure obtained from the filter. The variance of the estimate is the most common measure. Eck et al. (2012) and Kuusniemi et al. (2011) and Xu et al. (2009) use the KF to fuse at least two position estimates. Pany et al. (2010) and Lorga et al. (2010) also employ a KF. They run it as one part of their (multi-sensor) high-sensitivity receiver, where it integrates a WPS position estimate in order to backup the system in GNSS denied areas.

In (Hejc, Seitz, Boronat et al. 2013; Shah et al. 2007) a particle filter is employed to fuse the outcomes of different positioning systems.

1.2.3. Integration of GNSS pseudoranges with WPS features

Now we consider data fusion architectures that deduce a common position from features that were derived from sensory data. For GNSS, the feature of choice is always pseudorange, whereas in the case of WPS different features have been chosen.

Some studies fuse ranges from WLAN signals with GNSS pseudoranges (Fernandez et al. 2011; Li and O'Keefe 2013; Nur et al. 2013; Yan et al. 2012; Zirari et al. 2010), where the ranges from WLANs are either obtained by measureing the signal propagation time or by deducing ranges from RSSIs. This approach has two drawbacks: Knowledge of the WLAN access point positions is required and the line-of-sight (LOS) to the access points is assumed. Whereas it advantageous that the features are of the same physical quantity. This facilitates the use of the same position estimation method that is used in GNSS, compare section 2.4.1.

In (Hejc, Seitz and Vaupel 2014; Richter, Seitz et al. 2012) an approach integrating WLAN RSSI and GNSS pseudoranges is studied. As the two measurements are not of the same quantity the GNSS position estimation algorithms can not be applied.

Instead the authors form likelihood functions of the measurements and combine these in the correction step of a particle filter.

Using pseudoranges makes the localisation process more robust, because two pseudoranges or very few RSSIs observation can already contribute to the solution. Thus, potentially preventing complete failure of the system in environments that are harsh for GNSS and WPS.

1.3. Scope

This dissertation explores a multi-sensor data fusion approach for general purpose location estimation, targeting medium- to large-scale indoor/outdoor environments. It focuses on a tight integration of pseudoranges of GNSSs with signal strength measurements of WLANs to enable seamless positioning with meter-level accuracy. We study algorithms to improve the localisation performance with commercial off-the-shelf GNSS receivers and WLAN transceivers.

Hypothesis The filter which will be developed, combining statistically WLAN received signal strength measurements with GNSS pseudo ranges, will increase the localisation accuracy and precision, reliability and availability in comparison with general purpose localisation systems such as GNSS.

Objective Design and implementation of a particle filter for ubiquitous and seamless localisation, which permits to increase the accuracy, precision and availability of positioning systems.

This incorporates the following particular objectives:

1. Developing and evaluating the WLAN positioning system appropriate to be tightly fused with GNSS.

2. Developing a particle filter, which is able to estimate the user's position fusing the received WLAN signal powers and GNSS pseudoranges.

3. Analysing, evaluating and tuning the filter developed to improve robustness, precision and accurateness in comparison with the stand-alone systems, to achieve the goals of the hypothesis.

The combination of GNSS pseudoranges and WLAN signal strength has in addition the potential to mitigate errors occurring from multipath propagation of the GNSS signals.

1.4. Outline and contributions

The essential contribution of this thesis is the fusion of GNSS pseudoranges with WLAN RSSIs on a continuous state space.

The fundamentals of the methods, techniques and concepts used to achieve this objective are reviewed in chapter 2. It deals with the basics of location determination and estimation theory. Mainly, the maximum likelihood method and the recursive approach to Bayesian state estimation are described. We discuss briefly two numerical methods for the location estimation problem: the KF and the particle filter (PF). Furthermore, we introduce Gaussian processes and Gaussian process regression and conclude chapter 2 with the principles of GNSS and WLAN fingerprinting positioning system.

Chapter 3 describes the interpolation of the WLAN fingerprint database. We model the spatial RSSI distribution by Gaussian processes that allow to regress RSSI values in the neighbourhood of the discrete distributed fingerprints. The key contribution lies in an sound analyses of different Gaussian process models with real data in order to determine the best suited model for WLAN RSSIs. This analysis examines the necessity of different Gaussian process models for indoor and outdoor environments. While interpolating fingerprints reduces the labour of creating accurate fingerprint databases, it is an essential step to achieve accurate fusion of RSSI with pseudoranges.

In chapter 4 we present the methodology how the two measurements are combined within the Bayes filter. It describes the used models, whereof the model for the RSSI observation is based on the results of chapter 3. Subsequently, we derive the particle filter that actually combines the data via the measurement likelihood functions and validate the proposed method experimentally.

Finally, conclusions are drawn in chapter 5 and possible future work is pointed out.

Publications

The following publications appeared previously to this dissertation and are based on certain material of this work:

Refereed articles and papers

Richter, P., A. Peña-Torres and M. Toledano-Ayala. 2015. 'A Rigorous Evaluation of Gaussian Process Models for WLAN Fingerprinting'. In *International Conference on Indoor Positioning and Indoor Navigation (IPIN), 2015*. October. doi:10.1109/IPIN.2015.7346753.

Richter, P., and M. Toledano-Ayala. 2015. 'Revisiting Gaussian Process Regression Modeling for Localization in Wireless Sensor Networks'. *Sensors* 15, no. 9 (15th August): 22587–22615. doi:10.3390/s150922587.

Richter, P., M. Toledano-Ayala, G. M. Soto-Zarazúa and E. A. Rivas-Araiza. 2014. 'A Survey of Hybridisation Methods of GNSS and wireless LAN based Positioning System'. *Journal of Ambient Intelligence and Smart Environments* 6, no. 6 (November): 723–738. doi:10.3233/AIS-140289.

Patent applications

Richter, P. (Universidad Autónoma de Querétaro). 2014. Método y sistema para identificación de ubicación hibrido. Patent MX/E/2014/085924, filed 28th November 2014. filed.

Chapter 2

Background

Even though applications and use cases exist that deal with static objects whose position shall be identified; this work concentrates on the dynamic case. As objects in consideration are moving, so is location and the data we may observe about that process changing. Without further defining the object we know nothing about its motion or its location. Hence, we model the object's motion as a random process and the position, in a particular point of time, as a random variable.

This chapter considers the mathematical background and the models to cope with that kind of data, providing the foundations for the algorithm fusing GNSS pseudoranges and WLAN RSSI.

2.1. Determination of location

2.1.1. Positioning techniques

A location is always always determined in a certain reference frame of which certain reference points are known. These may be landmarks, the positions of the satellites for GNSS, the initial position for dead reckoning or some base stations of a communication network.

Based on the particular realisation of the system there are two possible modes of operation: *terminal-* or *infrastructure based*. A positioning system is terminal based if the position is calculated inside the mobile unit, and infrastructure based if the position is calculated inside the infrastructure. This depends basically on where the measurements are observed and processed. However, a communication channel between terminal and infrastructure leaves this decision to the system

designer, thus, details like privacy protection, energy consumption, etcetera can be incorporated.

Furthermore, four different principles are used to determine location information:

The simplest of the three positioning principles is just based on the concept of *proximity*. Assume that the object to be localised is noticed by one node of the corresponding network. This may happen by receiving radio frequency signals or perceiving what ever physical quantity the network uses for operation. If the position of that node is known, it is clear that the object is in the proximity of that node and its location can be roughly identified. In cellular networks for example, if a mobile terminal is connected to a base station, rather a large installation whose fix position are known, the network needs to identify the terminal's cell and automatically knows its location. Of course in other networks such a localisation method must be implemented.

The second positioning technique is known as *triangulation*. It enables localisation of on object by applying the geometrical properties of a triangle. Dependent on the available baselines or angles of the triangle, the missing information can be determined. If ranges to reference points are known or deduced from some observed quantities, the localisation method is called *lateration*. It might be trilateration, where the intersection of three spheres determines the three-dimensional solution. The radii of the spheres are the distances between the object and the reference points. Or it might be multilateration, where differences of distances to reference points are used to determine the solution. Instead of three spheres, three two-sheeted hyperboloids are formed by distance difference to four reference points. The intersection point of these hyperboloids determines the position in three-dimensional space. The observables are for both techniques often time measurements which are converted to distances via the velocity of the signal. As well signal powers are used to derive a distance to a signal emitter. Here, the difference between transmitted and received signal power is applied to a law of signal propagation, relating signal power to the distance to the emitter. Finally, the object is located by the rules of basic geometry that are applied to a set of distances (or differences of distances) to various base stations which form triangles. The other geometric approach is named *angulation*. Angulation is the traditional method of land surveying until the rise of GPS. The position solution is determined by intersecting the direction lines at three reference points obtained by the measured angles. In communication networks proper antennae

or antennae arrays are used to determine the angles relative to reference points, from which the position of the object is calculated.

The principle of the third positioning technique is to match features or patterns. Features are usually a set of a certain metrics of signals. It is assumed that the feature is unique at the each location and therefore forms together with the position a reference point for localisation, which can be recognised. This location dependence is exploited in two phases. In phase i), a survey process is carried out by assembling a geographical map of that feature. Throughout the considered site, it is scanned for the feature whose value/magnitude is recorded together with the coordinate where the scan took place. If the feature map is obtained once, the second phase ii) can commence. This is the localisation phase. Again it is scanned for the feature, this time at the sought position(s). Then the observed value is compared with the complete set of features recorded in the map. Finally, the sought position is obtained from the map where the feature matched (best). This method is quite often called *location fingerprinting*, contemplating the uniqueness of fingerprints[1]. The positioning technique is only reliable when the feature is sufficiently heterogeneous distributed to estimate reliable a position.

Inertial navigation is the last technique. INS operate completely without infrastructure, the measurements are made inside the inertial frame of the system. A pose has to be known initially. Then, after some specific time the distance and direction since the last post are determined and added to the subsequently added to the previous pose. The measurements are usually the time since the last position, the acceleration and rotation, which are converted into a position increment[2]. The current pose and time is the starting point for the following incremental update.

For more detailed information about the different positioning algorithms it is referred to (Gustafsson et al. 2005; Liu et al. 2007; Samama 2008; Titterton et al. 2004).

1. Remark, the assumption of unique reference point can in reality not be met. To overcome this problem a whole set of features (from various transmitters) is processed instead of a single feature. A larger set of features is more likely to be unique at a certain position.

2. The position increment can actually be obtained from different sensory data, such as distance from an odometer or speed from a speed sensor.

2.1.2. Location estimation

Determining geographic information is to infer some position or symbolic description of a place from our world. We use physical quantities to model and describe that world or certain environments of it in order to define functions that describe how we interact with that world and all its complexity and uncertainty. Location estimation requires measurements, which are at least subject to noise. To obtain reliable location information the value of interest (position, velocity, pitch, roll, yaw) must be infered from possibly indirect, inaccurate and imprecise observations. In other words, estimation is the 'process of selecting a point from a continuous space – the best estimate' (Bar-Shalom et al. 2001). Estimating the position of an object which moves in time, involves also the change of position over time. This can be modelled as a dynamic system.

The most common form to describe the dynamic system constituting the localisation problem is the *state space* notation: A set of first-order differential equations, describing the evolution of the location in time domain. The position of the object is part of the system's internal *state* or *state vector*. The state is actually a set of variables that are sufficient (beside some possible inputs to the system) to describe the current state of the system, and together with a model of the dynamic system it even describes its future behaviour. In the state space model, the state is hidden, described by so called latent variables, and is observed through measurements. The estimation of the object's state is often referred to as *filtering*, since the goal is to filter out the noise from the measurements approximating the state, the true position.

Many information, additional to the measurements, can reduce the estimation errors. Including knowledge about a former position, about the dynamics of the system, about the measurement errors and even redundant measurements. All this information can be processed, under certain assumptions, in an optimal fashion by the sequential Bayesian estimator, a method of Bayesian inference. This estimator represents the available information probabilistically. It establishes a hypothesis of the state in form of a probability density function and updates this hypothesis as additional evidence is obtained. What is more, as this framework deals universally with probability density functions (PDFs), information from completely different sources can be combined. It allows to 'fuse' information. The combination of different sensor information is known as multi-sensor data fusion.

For the localisation estimation problem we have the state vector, $\mathbf{x}_t$, that embeds the location information, some noise, $\mathbf{v}_t$, representing the uncertainty and some possible input to the system $\mathbf{u}_t$. As the object moves the model needs to adapt to these changes. To describe the the evolution of the dynamic system over time a differential equation is used – the *process model*

$$\dot{\mathbf{x}}_t = f(\mathbf{x}_t, \mathbf{u}_t, \mathbf{v}_t),$$

where $f(\cdot)$ is a not further specified non-linear function. The second part of the state space model relates the measurements, $\mathbf{y}_t$, to the system's state:

$$\mathbf{y}_t = h(\mathbf{x}_t, \mathbf{w}_t),$$

known as the *measurement model*, where $\mathbf{w}_t$ is the sensor noise. As well $h(\cdot)$ is a possibly non-linear function describing the sensor. The noises are usually assumed to be zero mean, uncorrelated and mutually independent.

2.2. Estimation techniques

The problem of location estimation can also be described statistically. Bayesian inference is very commonly used for that matter, but also frequentist methods such as the maximum likelihood (ML) estimator are often applied.

We begin this section with briefly presenting two estimation paradigms: The Bayesian method in section 2.2.1 and the ML method in section 2.2.2. Furthermore common estimation techniques for dynamic state estimation are presented in the sections 2.2.3, 2.2.4 and section 2.2.5.

2.2.1. Bayesian formalism

The Bayesian method provides a flexible and consistent 'formalism for reasoning about partial beliefs under conditions of uncertainty.' (Pearl 1988, p. 29). It includes all relevant information, such as prevenient knowledge, the history of observations and any kind of uncertainty. Unknowns in a Bayesian model are considered as a random variables. Accordingly, we model the real world with probabilities, sometimes called beliefs. To consider these probabilities as hypothesis is as well a common point of view.

To infer information via a Bayesian model one starts off with some initial knowledge, one may have or assume, about the system. Not until data, measurements, arrive from the system the belief (based on previous knowledge) is modified according to that data. Calculating the probability of the variable of interest, conditioned on the data, concludes the concept of Bayesian inference. The laws of probability and Bayes' theorem is all what is needed. Bayes' theorem for the PDFs[3] of the two stochastic variables x and y reads

$$\overbrace{p(x \mid y)}^{\textit{a posteriori}} = \frac{\overbrace{p(y \mid x)}^{\textit{likelihood}} \overbrace{p(x)}^{\textit{a priori}}}{\underbrace{p(y)}_{\textit{evidence}}}. \tag{2.1}$$

In this notation x represents the random parameter/variable we like to estimate and y some measurements the system provides. The term $p(x \mid y)$ is called the *a posteriori* PDF, or just posterior (density), where posterior is related to the arrival of data. The term $p(x)$ is known as the *a priori* PDF. It describes the information about the parameter x available prior to receiving data. If only a few or no information is known before the data arrives the prior is rather wide up to completely flat. The PDF that describes observation procedure is the *likelihood* function of x given y, reflecting the chances that y was obtained when x had taken certain values. The denominator of equation (2.1) is the often called the *evidence* and is a normalising constant (independent on x), making sure the posterior is a valid PDF integrating to one. If the system evolves in time and data is obtained repeatedly one can start off the estimation again but with the posterior as the current prior.

To estimate the parameter x a point estimate of the posterior distribution is to be calculated. Therefore several options exist (Ho et al. 1964). The maximum a posteriori (MAP) estimate is one choice, in particular on the basis of empirical data, usually assumed to be symmetric and unimodal. One looks for the value of x for which the posterior is maximised:

$$\hat{x}^{\text{MAP}} = \arg\max_{x} p(x \mid y). \tag{2.2}$$

3. This text does not formally distinguish between probabilities and PDFs of continuous random variables, because they differ only in a constant prefactor which we omit for the sake of simplicity, see also (Barber 2014, ch. 1).

That procedure results simply in the mode of the posterior.

A more general solution can be formulated if one minimise the error. Let $\tilde{x} = x - \hat{x}$ the error of the estimate. The optimal point estimate is the estimator that minimises that error in average. Choosing the minumum mean square error (MMSE) as optimisation criterion, the resulting estimator minimises the expectation of the quadratic error of the estimate $\hat{x}$

$$\arg\min_x \mathbb{E}\left[(\hat{x} - x)^2 \mid y\right]$$

Bar-Shalom et al. (2001) shows that if the posterior is differentiable the solution of the minimisation problem is the *conditional mean*

$$\hat{x}^{\text{EAP}} = \mathbb{E}[x \mid y] \triangleq \int x p(x \mid y)\, \mathrm{d}x. \tag{2.3}$$

Equation (2.3) is also referred to as expected a posteriori (EAP) estimate. It is the optimal estimate regardless the posterior PDF. If the posterior is symmetric and unimodal, such as the Gaussian distribution, the MAP and the EAP estimate are identical, since in that case the mean and the median coincide (Bar-Shalom et al. 2001). As well other cost functions have been used, as for example $\arg\min_x \mathbb{E}\left[|\hat{x} - x| \mid y\right]$. One obtains the minimax estimate resulting in the median of $p(x \mid y)$, compare (Ho et al. 1964).

2.2.2. Maximum likelihood estimator

In contrast to Bayesian methods, the ML estimator, as a frequentist method, considers the sought parameters as unknown but fixed, not random as in the Bayesian context. In this context, the notion of previous information is accepted neither, information is infered only on the base of (random) data.

To estimate a parameter in the ML sense first the likelihood function is to be constructed (compare equation (2.1)). For different possible values of the parameter the likelihood function provides probabilities of getting a certain realization from the random data. The ML estimate is simply the value corresponding to the maximum of the likelihood function

$$\hat{x}^{\text{ML}} = \arg\max_x p(y \mid x) \tag{2.4}$$

The estimate depends on the data and is therefore a function of the random data. Nevertheless, the likelihood function is viewed as function of the parameter because the probability of the observed, hence fixed, data is calculated for possible values of the parameter. One often writes $L(x)$ for $L(y \mid x) \triangleq p(y \mid x)$.

Remark: Assuming a flat prior distribution[4] in equation (2.1) the MAP estimate in equation (2.2) results in the ML estimate (2.4).

2.2.3. Sequential Bayesian estimation

Consider a stationary *random process* $\{\mathbf{x}_t, t \in \mathcal{T}\}, t \in \mathbb{R}^1, \mathbf{x} \in \mathbb{R}^{n_x}$ describing the object's motion, and a (vector valued) random variable $\mathbf{x}_t$ – the state or state vector that representing the physical system. Let $\mathbf{x}_{t_0}, \ldots, \mathbf{x}_{t_k}$ be the system's state as at several consecutive time instances $t_0 < t_2 < \ldots < t_k$ up to t_k, expressing that the data is sequential, and reflecting its change with time.

Even though the problem – to find an object's position over time – is a continuous-time problem, the objective of this work is an computable solution for the data fusion problem. That is, to consider a discrete parameter set for the random process which therefore becomes a random sequence $\{\mathbf{x}_k, k = 1, 2, \ldots\}$. Since we will deal with events consecutive in time we abbreviate a sequence of states up to time k with $\mathbf{x}_{0:k}$. The characteristic of a random process or sequence are embodied in its PDF. We can denote the process' joint density function by $p(\mathbf{x}_0, \ldots, \mathbf{x}_k)$ or just $p(\mathbf{x}_{0:k})$ and its *aposteriori* PDF by $p(\mathbf{x}_{0:k} \mid \mathbf{y}_{1:k})$. Bayes' theorem for sequences of the states and observations $\mathbf{y}_{1:k}$ becomes

$$p(\mathbf{x}_{0:k} \mid \mathbf{y}_{1:k}) = \frac{p(\mathbf{y}_{1:k} \mid \mathbf{x}_{0:k}) p(\mathbf{x}_{0:k})}{p(\mathbf{y}_{1:k})}. \tag{2.5}$$

Where the observations, that arriving from the system, are modelled as a random sequence $\{\mathbf{y}_k, k = 1, 2, \ldots\}, \mathbf{y}_k \in R^{n_y}$ too.

In the localisation problem the last position (incorporated into the state vector) is intuitively correlated to its predecessor, from which the object moved to the last position. This dependence of the most recent position on previous ones represents the conditional PDF $p(\mathbf{x}_k \mid \mathbf{x}_{0:k-1})$. One can qualify this further, such

4. Note that a flat prior or *diffuse* prior has infinite variance resulting in an improper PDF, since it does not integrate to one and has no moments. Nonetheless the posterior is usually a proper PDF. (Bar-Shalom et al. 2001)

that the most recent state only depends on the directly previous state, yielding

$$p(\mathbf{x}_k \mid \mathbf{x}_{0:k-1}) = p(\mathbf{x}_k \mid \mathbf{x}_{k-1}). \tag{2.6}$$

This property is known as the Markov property and random processes that inherit that property are first-order *Markov process*es. The history of the process does not contain any new information if the current state is known; that is, the system is memoryless. Bar-Shalom et al. (2001) puts that nicely as 'The future is independent of the past if the present is known.'.

A similar assumption can be made about the sequence of observations. To look at a measurement as making a snapshot of the current state makes obvious that an observation only depends on the current state, but only the state. Expressing that probabilistically obeys

$$p(\mathbf{y}_k \mid \mathbf{y}_{1:k-1}, \mathbf{x}_{0:k}) = p(\mathbf{y}_k \mid \mathbf{x}_k). \tag{2.7}$$

The latter two equations describe statistically the discrete-time process model and the measurement model. Equation (2.6) exhibits the Markov property and is called the *transition probability density*, describing the evolution from one time instance to the next Equation (2.7) is again a likelihood function.

However, we can also formulate the state space model as analogue to the continuous-time state space model in section 2.1.2, though with additive noise. The discrete-time process model becomes

$$\mathbf{x}_k = f_{k-1}(\mathbf{x}_{k-1}) + \mathbf{v}_{k-1}, \tag{2.8}$$

where f_{k-1} defines a possibly non-linear function describing state's evolution in time and $\{\mathbf{v}_k, k = 1, 2, \dots\}, \mathbf{v}_k \in \mathbb{R}^{n_v}$ denotes the identical, independent distributed (i.i.d.) process noise. Notice that as well the left hand side of the general model (2.8) depends solely on the last state.

Above we stated that, because of the intuitive characteristics of the localisation problem – moving from one position to the next position, and observing a single point in time – one can assume that the state is Markov. But it can be shown that any process being an outcome of deterministic dynamic system that is described by ordinary differential or difference equations and that is driven by uncorrelated noise is a continuous- or discrete-time Markov process (Bar-Shalom et al. 2001)

To complete the state space equation, we describe the observation as function

of the state $\mathbf{x}_k$ too; the measurement model is

$$\mathbf{y}_k = h_k(\mathbf{x}_k) + \mathbf{w}_k. \tag{2.9}$$

The random sequence $\{\mathbf{w}_k, k = 1, 2, \dots\}$, $\mathbf{w}_k \in setR^{n_w}$ is termed the measurement noise, which is also I.I.D., independent of the process noise, the state and the observations.

Recall equation (2.5), that infering the parameter $\mathbf{x}$ means to estimate the (underlying) posterior PDF $p(\mathbf{x}_{0:k} \mid \mathbf{y}_{1:k})$. Equation (2.6) and (2.7) allow us to derive a two step recursive procedure updating the *marginal posterior* PDF, $p(\mathbf{x}_k \mid \mathbf{y}_{1:k})$; the density reflecting the current state, its uncertainty and the history of the observations, see appendix A. The *sequential Bayesian estimator* consists of two steps; the *process update* (2.10a)

$$p(\mathbf{x}_k \mid \mathbf{y}_{1:k-1}) = \int p(\mathbf{x}_k \mid \mathbf{x}_{k-1}) p(\mathbf{x}_{k-1} \mid \mathbf{y}_{1:k-1}) \, \mathrm{d}\mathbf{x}_{k-1}. \tag{2.10a}$$

that predicts the PDF of the state at time k using the process model in form of the transitional PDF and the posterior from time $k-1$; and the *measurement update* step (2.10b), correcting (2.10a) by applying the observation in form of the likelihood PDF, thus yielding the marginal posterior PDF of the state

$$p(\mathbf{x}_k \mid \mathbf{y}_{1:k}) = \frac{p(\mathbf{y}_k \mid \mathbf{x}_k) p(\mathbf{x}_k \mid \mathbf{y}_{1:k-1})}{p(\mathbf{y}_k \mid \mathbf{y}_{1:k-1})}. \tag{2.10b}$$

Estimator (2.10) incorporates all statistical information needed to infer $\mathbf{x}_k$ in an optimal fashion from the filtering PDF. It processes new information sequentially as it becomes available instead of storing all information of all time instances. It furthermore avoids to eventually process information that has been already processed before.

The denominator of equation (2.10b) is given by

$$p(\mathbf{y}_k \mid \mathbf{y}_{1:k-1}) = \int p(\mathbf{y}_k \mid \mathbf{x}_k) p(\mathbf{x}_k \mid \mathbf{y}_{1:k-1}) \, \mathrm{d}\mathbf{x}_k,$$

being a scalar constant that normalises the posterior PDF such that it integrates to one.

To estimate the posterior PDF the process and measurement model, f_k and h_k, and the disturbances, $\mathbf{v}_k$ and $\mathbf{w}_k$, must be known. In addition an initial (prior) state $\mathbf{x}_0$ and its PDF, $p(\mathbf{x}_0)$, must be available. The prediction of the posterior involves the state transition PDF that is determined by $p(\mathbf{x}_k \mid \mathbf{x}_{k-1}) = p_{\mathbf{v}}(\mathbf{x}_k - f_{k-1}(\mathbf{x}_{k-1}))$ and the likelihood function that is determined by $p(\mathbf{y}_k|\mathbf{x}_k) = p_{\mathbf{w}}(\mathbf{y}_k - h_k(\mathbf{x}_k))$.

The described method, estimating $\mathbf{x}_l$ based on the observations $\mathbf{y}_{1:k}$ at current time $l = k$, is often referred to with *filtering*, since it infers the presence $p(\mathbf{x}_l \mid \mathbf{y}_{1:k})$. In contrast, one speaks of smoothing when the past is of interest and the sought-after posterior is $p(\mathbf{x}_l \mid \mathbf{y}_{1:k})$, $l < k$, or of predicting when one wants to infere a future state $p(\mathbf{x}_l \mid \mathbf{y}_{1:k})$, $l > k$. Therefore we use the denotation *filtering density* for the left hand side of equation (2.10b).

The most common Bayesian filtering problems comply with the following interpretation: Equation (2.10a) propagates the state's PDF forward in time. The random noise $\mathbf{v}_k$ usually increases uncertainty of the believe about the state. Equation (2.10b) incorporates the likelihood function into the estimation process. It introduces new data, which usually decreases the uncertainty about the state's believe, resulting in an updated posterior PDF of the state.

The sequential Bayesian estimator is quite general framework that applies to many kinds of stochastic filtering tasks, including non-linear process and measurement models, multimodal noises of any distribution. But a general analytic solution for stochastic filtering problem (2.10) does not exist. For the specific case of linear process- and measurement model with Gaussian process- and measurement noise the Kalman filter (Kalman 1960) is the optimal solution in the mean square sense. In other cases only approximations exist. For example to approximate a multi-model posterior by its moments the Gaussian mixture approach is a common choice, pioneered by Alspach et al. (1972). If the state space is discrete the PDFs can be evaluated at the finite set of points. These methods are often called grid-based method and in the case of a finite, discrete state space they provide an optimal solution to the Bayesian recursion. A further option is to solve the integrals numerically for example by a method called particle filtering, which is based on stochastic integration.

2.2.4. Kalman filter

This section shows an analytic solution to the Bayesian filtering problem (2.10) for linear Gaussian state space models.

A linear a process and linear a measurements procedure, both with additive noises are described by the discrete-time state space equations (2.11).

$$\mathbf{x}_k = F_k \mathbf{x}_{k-1} + G\mathbf{v}_{k-1} \tag{2.11a}$$

and

$$\mathbf{y}_k = H_k \mathbf{x}_k + \mathbf{w}_k. \tag{2.11b}$$

The noises $\mathbf{v}_{k_1}$ and $\mathbf{w}_k$ are mutually independent, zero mean Gaussian distributed noises of mean zero. They are also independent of the state vector $\mathbf{x}_k$ and the observation vector $\mathbf{y}_k$. Due to the linear models and the Gaussian distributed noises the state is a linear combination of normally distributed random variables and therefore itself Gaussian. At any stage of the recursion (2.10), the corresponding distributions (prior, predictive or posterior) remain Gaussians:

$$\begin{aligned} p(\mathbf{x}_{k-1} \mid \mathbf{y}_{1:k-1}) &= \mathcal{N}(\hat{\mathbf{x}}_{k-1|k-1}, P_{k-1|k-1}) \\ p(\mathbf{x}_k \mid \mathbf{y}_{1:k-1}) &= \mathcal{N}(\hat{\mathbf{x}}_{k|k-1}, P_{k|k-1}) \\ p(\mathbf{x}_k \mid \mathbf{y}_{1:k}) &= \mathcal{N}(\hat{\mathbf{x}}_{k|k}, P_{k|k}), \end{aligned}$$

where $\hat{\mathbf{x}}_{k|k} = \mathbb{E}[p(\mathbf{x}_k \mid \mathbf{y}_{1:k})]$ is the conditional mean of the state and $P_{k|k} = \mathbb{E}[\tilde{\mathbf{x}}_{k|k}\tilde{\mathbf{x}}_{k|k}^T \mid \mathbf{y}_{1:k-1}]$ is the conditional covariance matrix of the state's error $\tilde{\mathbf{x}}_k = \hat{\mathbf{x}}_k - \mathbf{x_k}$.

Instead of propagating the distributions with time, only the parameters of the distributions, the conditional mean and covariance – the the distribution's the first- and second order moment – are updated through (2.8) and (2.9). Kalman (1960) developed that approach for nonstationary problems.

Assume, $\hat{\mathbf{x}}_{0|0}$ being the initial (prior) conditional mean and $P_{0|0}$ being the conditional covariance matrix of the state $\mathbf{x}_0 \sim p(\mathbf{x}_0) = \mathcal{N}(\hat{\mathbf{x}}_{0|0}, P_{0|0})$ given the measurements $\mathbf{y}_{0|-1}$. Let the process noise be given by $\mathbf{v}_k \sim \mathcal{N}(0, Q_k)$, then the process update of the Kalman filter is

$$\hat{\mathbf{x}}_{k|k-1} = F_k \hat{\mathbf{x}}_{k-1|k-1} \tag{2.12a}$$

$$P_{k|k-1} = F_k P_{k-1|k-1} F_k^T + G_k Q_{k-1} G_k^T. \tag{2.12b}$$

Equation (2.12a) propagates the estimate of the previous state through the

process model and equation (2.12b) adapts the uncertainty about the state accordingly, while adding the mapped covariance of the process noise. The results are predictions of the posterior PDF's parameters.

Using the results of equations (2.12) and $\mathbf{w}_k \sim \mathcal{N}(0, R_k)$, the Kalman filter measurement update obeys

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + K_k(y - H_k\hat{\mathbf{x}}_{k|k-1}) \tag{2.13a}$$

$$P_{k|k} = P_{k|k-1} - K_k\, H_k P_{k|k-1}. \tag{2.13b}$$

The matrix K is the *Kalman gain* matrix, given by $K_k = P_{k|k-1}H_k^T(H_k P_{k|k-1}H_k^T + R_k)^{-1}$.

The term $\mathbf{y} - H_k\hat{\mathbf{x}}_{k|k-1}$ is often referred to as innovation and the term in the Kalman gain $H_k P_{k|k-1}H_k^T + R_k$ is the innovation covariance respectively, they correspond to the likelihood function in equation (2.10b). The innovation covariance is a measure for the quality of the observations. The innovation is added to the predicted state estimate, see (2.13a). But the Kalman gain amplifies the influence of the innovation if the prediction covariance is large and the innovation covariance is small, and vice versa. The interpretation of the covariance update (2.13b) is likewise straight forward. The posterior PDF's covariance results in the predicted covariance minus the mapped prediction covariance modified by the Kalman gain.

If all assumptions are met, the mean and covariance parametrise the posterior PDF exactly. Then the state estimate, the conditional mean of the filtering density and the MMSE estimate. The Kalman filter is thus the optimal solution.

2.2.5. Particle filter

In the sequel a very versatile method solving the sequential Bayesian estimation problem is presented: the *particle filter*. It carries out the integrations involved in (2.10) numerically by approximating the continuous PDFs by discrete random measures and solving the resulting sums. Their advantages lie in their possibility to approximate arbitrary functions (PDFs), their convergence properties and their possibility to be implemented in parallel (Djurić et al. 2003). Compared to grid-based methods particle filter are independent of the dimension of the state space.

Particle filter rely on the principle of *importance sampling* which can be formulated

sequentially such that it can be employed for sequential Bayesian filtering. Gordon et al. (1993) presented the first efficient particle filter (Doucet, Freitas et al. 2001), based on *sequential importance sampling* with an resampling step, and demonstrated the application of particle filter for localisation and tracking.

Monte Carlo integration

The objective of Bayesian filtering is to estimate the filtering PDF $p(\mathbf{x}_k \mid \mathbf{y}_{1:k})$ and some parameter of that density. But to simplify the expressions, we consider just the random variable x. In general one needs to solve integrals of the form

$$\mathbb{E}[b(x)] = \int b(x)p(x)\,\mathrm{d}x.$$

A parameter which is often of interest is a point estimate like the expectation $b(x) = x$, or perhaps a moment $b(x) = (x - \mathbb{E}[x])^n$. Assume that a large number of I.I.D. samples could be draws uniformly from the function of interest $\{x^{(i)}, i = 1, 2, \ldots, N\} \sim p(x)$. Then one could use

$$p(x) \approx \frac{1}{N} \sum_{i=1}^{N} \delta(x - x^{(i)}), \tag{2.14}$$

to approximate the integral with a simple sum

$$\begin{aligned}\mathbb{E}[b(x)] = \int b(x)p(x)\,\mathrm{d}x &\approx \int b(x)\frac{1}{N}\sum_{i=1}^{N}\delta(x - x^{(i)})\,\mathrm{d}x \\ &\approx \frac{1}{N}\sum_{i=1}^{N} b(x^{(i)}).\end{aligned} \tag{2.15}$$

Such an estimate is unbiased and converges almost surely to the true estimate as the number of random samples goes to infinity Doucet, Freitas et al. (2001). This method is often called perfect importance sampling.

But perfect importance sampling is still not the solution to the general Bayesian filter. Generating samples that belong to standard distributions such as the Gaussian, or Student distribution is usually done via transformations of I.I.D. uniform distributed samples obtained from pseudo random sequences. Mixtures of

standard distributions can be used in turn to generate more complex distributions. But the generation of samples representing an arbitrary posterior distribution, especially if it is of high dimensionality, is not straight forward (MacKay 2005). The PDFs of the Bayesian filter are in general quite complex functions. In addition, these functions or their possible parametrisation is unknown.

Importance sampling

The idea of perfect importance sampling relies on the knowledge of the posterior density. But in our filtering problem it is the sought distribution; thus we can not sample from it. But let's assume we knew a density that is proportional up to a multiplicative constant to the sought state posterior and from which it is easy to draw samples $p(x) \propto q(\cdot)$. The function $q(\cdot)$ is known as the *importance function*.

To use the samples generated from the importance function $x^{(i)} \sim q(\cdot)$ in equation (2.14), we must account for the values where the importance function deviates from sought distribution $p(\cdot)$. This is achieved by defining some weights that multiplicatively adjust for sampling from the importance function instead of the sought distribution. The proportionality factor is simply:

$$\tilde{\omega}^{(i)} = \frac{p(x^{(i)})}{q(x^{(i)})},$$

named the importance weights. In this probabilistic context the weights are normalised so that they sum to unity.

$$\omega^{(i)} = \frac{\tilde{\omega}^{(i)}}{\sum_{j=1}^{N} \tilde{\omega}^{(j)}}. \tag{2.16}$$

Each particle $x^{(i)}$ is assigned an normalised importance weight $\omega^{(i)}$. The set of particles and associated weights form the random measure $\mathcal{X} = \{x^{(i)}, \omega^{(i)}\}_{i=1}^{N}$.

Replacing the equally weighted samples in equation (2.14) with this random

measure the approximation (2.15) is still correct and results in

$$\begin{aligned}\mathbb{E}[b(\mathbf{x})] &= \int b(x)p(x)\,\mathrm{d}x \approx \int b(x)\sum_{i=1}^{N}\frac{1}{\sum_{j=1}^{N}\tilde{\omega}^{(j)}}\frac{p(x^{(i)})q(x^{(i)})}{q(x^{(i)})}\,\mathrm{d}x \\ &\approx \int b(x)\sum_{i=1}^{N}\frac{1}{\sum_{j=1}^{N}\tilde{\omega}^{(j)}}\frac{p(x^{(i)})}{q(x^{(i)})}\delta(x-x^{(i)})\,\mathrm{d}x = \sum_{i=1}^{N}\frac{\tilde{\omega}^{(i)}}{\sum_{j=1}^{N}\tilde{\omega}^{(j)}}b(x^{(i)}) \\ &\approx \sum_{i=1}^{N}\omega^{(i)}b(x^{(i)}). \end{aligned} \tag{2.17}$$

For finite N this estimate is biased, but if $N \to \infty$ it converges towards equation (2.15) (Doucet, Freitas et al. 2001). For the particles approximation of the sought density to be accurate enough a few conditions must be met. The most important one is that the importance function is sufficiently heavy tailed to 'cover' the sought distribution completely. More details on this can be found in (Doucet, Freitas et al. 2001; MacKay 2005).

As the mean is computed by $\hat{x} = \sum_{i=1}^{N}\omega^{(i)}x^{(i)}$, one can also compute the standard deviation of the particles. This serves as quality measure of the point estimate and is given by

$$\sigma_p = \sqrt{\mathbb{E}[(\hat{x}-x^{(i)})^2]} = \sqrt{\frac{\sum_{i=1}^{N}\omega^{(i)}(\hat{x}-x^{(i)})^2}{\frac{(N'-1)\sum_{i=1}^{N}\omega^{(i)}}{N'}}}, \tag{2.18}$$

where N' denotes the non-zero weights (Weyn 2001).

It follows the adaption of this concept to the sequential estimation problem, in particular, we clarify how to determine the weights recursively.

Sequential importance sampling

To adapt the concept of importance sampling to the sequential problem, we consider again sequences of states. Let the posterior PDF at any time instance, $p(\mathbf{x}_{0:k} \mid \mathbf{y}_{1:k})$ to be the PDF to be represented by particles. The random measure at time k is denoted by $\mathcal{X}_k = \{\mathbf{x}_k^{(i)}, \omega_k^{(i)}\}_{i=1}^{N}$, constituting $\mathbf{x}_k^{(i)}$, a state vector of the ith particle and $\omega_k^{(i)}$ an associated weight. We look for an recursive particle approximation expressing the joint posterior PDF (2.5).

Suppose one can choose an importance function (proportional to the posterior of the state sequences) obeying a structure that contains the marginal importance density at time $k-1$, $q(\mathbf{x}_{0:k-1} \mid \mathbf{y}_{1:k-1})$, as given by

$$q(\mathbf{x}_{0:k} \mid \mathbf{y}_{1:k}) = q(\mathbf{x}_k \mid \mathbf{x}_{0:k-1}, \mathbf{y}_{1:k}) q(\mathbf{x}_{0:k-1} \mid \mathbf{y}_{1:k-1}). \tag{2.19}$$

This suggests that previous samples $\mathbf{x}_{0:k-1}^{(i)} \sim q(\mathbf{x}_{0:k-1}^{(i)} \mid \mathbf{y}_{1:k-1})$ can be augmented with new samples $\mathbf{x}_k^{(i)} \sim q(\mathbf{x}_k \mid \mathbf{x}_{0:k-1}^{(i)}, \mathbf{y}_{1:k})$ to determine $\mathbf{x}_{0:k-1}^{(i)} \sim q(\mathbf{x}_{0:k}^{(i)} \mid \mathbf{y}_{1:k})$ (Djurić et al. 2003); in analogy to the process update. Furthermore, one can show (see A.2), similar to the sequential Bayesian estimator, that the prior joint posterior PDF of the states, $p(\mathbf{x}_{0:k-1} \mid \mathbf{y}_{1:k-1})$, can be updated by observations of the system. Omitting the constant scalar denominator gives

$$p(\mathbf{x}_{0:k} \mid \mathbf{y}_{1:k}) \approx p(\mathbf{y}_k \mid \mathbf{x}_k) p(\mathbf{x}_k \mid \mathbf{x}_{k-1}) p(\mathbf{x}_{0:k-1} \mid \mathbf{y}_{1:k-1}), \tag{2.20}$$

that corresponds to the measurement update of the Bayesian filter. Now, expressing the importance weights as if samples were drawn from $p(\mathbf{x}_{0:k} \mid \mathbf{y}_{1:k})$

$$\tilde{\omega}^{(i)} \propto \frac{p(\mathbf{x}_{0:k}^{(i)} \mid \mathbf{y}_{1:k})}{q(\mathbf{x}_{0:k}^{(i)} \mid \mathbf{y}_{1:k})} \tag{2.21}$$

and substituting (2.20) and (2.19) into the latter equation (2.21), one yields

$$\tilde{\omega}_k^{(i)} \propto \frac{p(\mathbf{y}_k \mid \mathbf{x}_k^{(i)}) p(\mathbf{x}_k^{(i)} \mid \mathbf{x}_{k-1}^{(i)})}{q(\mathbf{x}_k^{(i)} \mid \mathbf{x}_{0:k-1}^{(i)}, \mathbf{y}_{1:k})} \underbrace{\frac{p(\mathbf{x}_{0:k-1}^{(i)} \mid \mathbf{y}_{1:k-1})}{q(\mathbf{x}_{0:k-1}^{(i)} \mid \mathbf{y}_{1:k-1})}}_{\propto \omega_{k-1}^{(i)}}, \tag{2.22}$$

an expression that evolves the importance weights in time. After normalising the weights, they can be plugged into equation (2.17) completing the sequential importance sampling method to the determine the sought estimate.

Particle depletion An intrinsic problem of sequential importance sampling not discussed so far is the degeneracy problem. It is that after a few iterations only a few particles have large weight and the majority of particles will have negligible weights, and the particle approximation fails to represent the posterior. With

an importance function as in equation (2.19) the variance of the weight can only increase over time (Doucet, Godsill et al. 2000). The degeneracy problem can be reduced by the choice of the importance function. The optimal importance function is one that minimises the variance of the weights and is given by $p(\mathbf{x}_k \mid \mathbf{x}_{k-1}^{(i)} \mathbf{y}_k)$; but it has usually no closed form. Doucet, Godsill et al. (2000) states further strategies to obtain an adequate importance functions. However, the degeneracy problem persists.

To overcome the sample degeneracy problem of sequential importance sampling one introduces a further *resampling* procedure that removes particles that have insignificant weight and reproduces particles with considerable weight. This redistributes the particles and spreads them across the posterior. Several resampling schemes have been proposed in the literature, Doucet and Johansen (2009) states a few of them.

A criterion that measures the degeneracy of the particles, indicating the goodness of the particle filter is the effective sample size. An estimator of the effective sample size is

$$\hat{N}_{\text{eff}} \approx \frac{1}{\sum_{i=1}^{N} \left(\omega^{(i)}\right)^2} \tag{2.23}$$

(Doucet, Godsill et al. 2000).

The sequential importance sampling with resampling provides a base for many different versions of particle filters. A particular implementation is presented in the sequel.

Bayesian bootstrap filter

The choice of the importance function can be quite complicated. A convenient choice not mentioned before is the process model. Let the importance function be $q(\mathbf{x}_k^{(i)} \mid \mathbf{x}_{0:k-1}^{(i)}, \mathbf{y}_{1:k}) = p(\mathbf{x}_k \mid \mathbf{x}_{k-1})$. Substituting this into equation (2.22), the importance weight update depends only on the likelihood function

$$\tilde{\omega}_k^{(i)} \propto p(\mathbf{y}_k \mid \mathbf{x}_k^{(i)}) \omega_{k-1}^{(i)}. \tag{2.24}$$

Sequential importance sampling with the state transition PDF as importance function and a resampling step after every weight update and normalisation is known Bayesian bootstrap filter according to (Gordon et al. 1993). A variety of

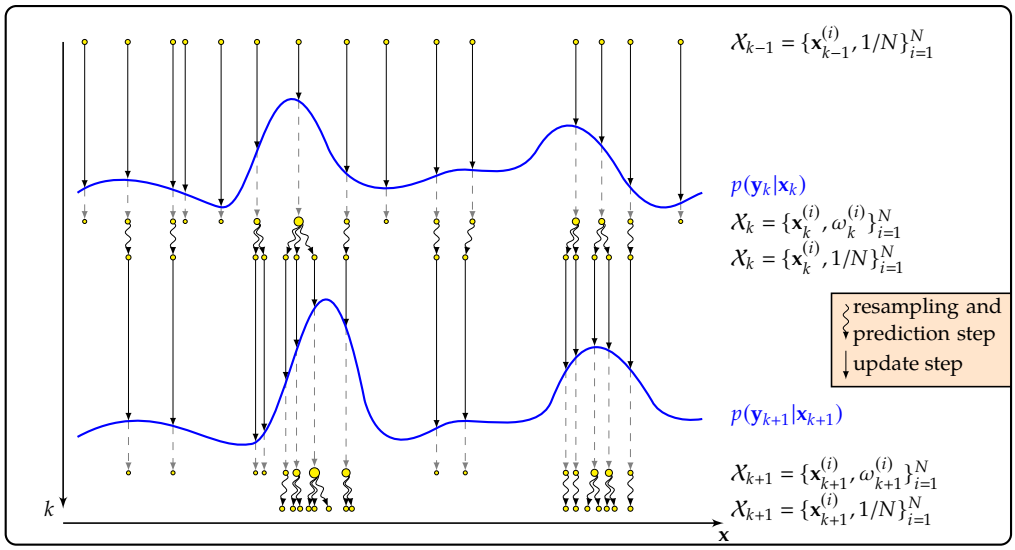


Figure 2.1.: Bootstrap Particle Filter scheme for one-dimensional state estimation. (Doucet, Freitas et al. 2001)

resampling procedures are available, we refer to (**Hol04:Resampling**) for the four most common algorithms. The Bayesian bootstrap filter resamples at each time step and after the resampling the particles are assigned equal weights. This is illustrated in figure 2.1.

Note that this kind of particle filter has certain drawbacks. Choosing a prior PDF as importance function bears the consequence that particles are chosen without any knownledge about the measurements, thus making it prone to outliers (Doucet, Godsill et al. 2000).

Even though resampling is effective against the degeneracy problem by removing particles with very small weights, it is worth noting that it introduces an additional issue. Repeated draws of the same particle are likely and reduce the support of the density function, hence, degrades the quality of the particle approximation. This is known as sample impoverishment. For that reason one should only resample if necessary. A commonly applied scheme is to resample when the effective sample size reaches a certain threshold, for example $N_{\text{th}} = N/2$. Moreover, resampling limits parallelisation as particles are combined.

2.3. Gaussian process regression

2.3.1. Gaussian process modelling

A Gaussian process is a stochastic process. It can be seen as a generalised multi-dimensional Gaussian distribution with infinite many random variables. Each subset of these random variables is again Gaussian distributed.

Definition 2.3.1. *A* Gaussian process *is a collection of random variables, any finite number of which have a joint Gaussian distribution (Rasmussen et al. 2006).*

According to the idea of a Gaussian distribution of infinite many variables, a Gaussian process is not specified by scalars or vectors, but by functions. As a multi-dimensional Gaussian distribution is completely described by its two parameters, a vector of expectations and a covariance matrix, a Gaussian process is completely described by its *mean function* $m(\mathbf{x}) = \mathbb{E}[f(\mathbf{x})]$ and *covariance function* $k(\mathbf{x}, \mathbf{x}') = \mathbb{E}[(f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}') - m(\mathbf{x}'))]$. Hence, Gaussian processes are distributions over functions. (Rasmussen et al. 2006)

With the mean and covariance function of a real process $f(\mathbf{x})$, the Gaussian process will be denoted with

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')).$$

The variables $\mathbf{x}$ and $\mathbf{x}'$ are values of the function f. The mean function characterises a global structure of the Gaussian process and the covariance function describes how the two points, x and x', are correlated.

Gaussian process modelling can also be seen from a Bayesian point of view. A non-linear function $f(\mathbf{x})$ is assumed to underlie the data $\{\mathbf{x}_n, f_n\}_{n=1}^N$. Inferring the function $f(\mathbf{x})$ in this context means seeking its posterior PDF. With the definitions of the matrix of input vectors $X \equiv \{\mathbf{x}_n\}_{n=1}^N$ and the vector of observed target values $\mathbf{f} \equiv \{f_n\}_{n=1}^N$, the posterior probability is given, according to Bayes' theorem (2.1), with

$$p(f(\mathbf{x}) \mid \mathbf{f}, X) = \frac{p(\mathbf{f} \mid f(\mathbf{x}), X)p(f(\mathbf{x}))}{p(\mathbf{f} \mid X)}. \tag{2.25}$$

The function $f(\mathbf{x})$ is usually a single sample function of the data to be modelled. $p(f(\mathbf{x}))$ is the a priori PDF over functions assumed on the model. The prior's importance lies in determining the general characteristic of the function which is to be inferred. This characteristic includes smoothness, mean and uncertainty,

continuity or the spectral distribution of energy. These properties can be specified or manipulated by means of the mean and covariance function. The likelihood function $p(\mathbf{f} \mid f(\mathbf{x}), X)$ describes how probable are the observed target values for different settings of the function $f(\mathbf{x})$; it expresses the observations. These two probabilities are the relevant terms to infer future target values f_n or to predict/interpolate $f(\mathbf{x})$, based on samples.

The covariance functions are specified by kernels. They express the correlation between two points of the random process. Two nearby points are usually highly correlated and close by points are stronger correlated; covariance functions drop off with larger distance. A wide variety of covariance functions with different properties are readily available (ch. 4), they can be determined analytically or obtained by combination of valid covariance functions. The choice of the covariance function depends greatly on the underlying data. An example of a very common covariance function is the *squared exponential* covariance function

$$\operatorname{cov}(f(\mathbf{x}_p), f(\mathbf{x}_q)) = k(\mathbf{x}_p, \mathbf{x}_q) = \exp\left(-\frac{1}{2}||\mathbf{x}_p - \mathbf{x}_q||^2\right). \tag{2.26}$$

The covariance and possible as well the mean function have free parameters, the, so-called *hyperparameter*. How to determine these parameters is outlined in the next section.

2.3.2. Gaussian process regression

Gaussian process regression is based on the idea of Bayesian inference and can be used to predict or interpolate values of a unknown, sought non-linear function f. Regression based on Gaussian process modelling is a non-parametric regression method.

The principle concept of regression analysis is to infer an unknown function that describes a sought relationship between a dependent variable and data samples, with help of these available data samples. Consider the unknown relation $y(\mathbf{x}) = f(\mathbf{x}) + \varepsilon$, which shall be modelled with a Gaussian process, of which only a set of noisy *training data* $\mathcal{D} = \{(\mathbf{x}_n, y_n)\}_{n=1}^N$ can be obtained. Where $\mathbf{x}$ denotes an input vector of dimension D and y denotes a scalar output value or target value. For sake of convenience the N input vectors are collected in the columns of a matrix, X of input data, the output values in a vector of target values $\mathbf{y}$. For now also the noise will be neglected, thus the vector of target values

becomes $\mathbf{f}$, so that the noiseless training data set can be written as $\mathcal{D} = \{X, \mathbf{f}\}$. The data which is to be predicted is denoted by $\mathbf{f}^*$, hence $\mathcal{D}^* = \{X^*, \mathbf{f}^*\}$ denotes the set of predicted target values and its corresponding input values. To this set it is referred to as *test data*. As the input data forms a matrix of N input data vectors also the covariance of equation (2.26) for N input vectors can be rewritten in matrix form, $K(X, X)$.

Posterior distribution

As mentioned earlier, a Gaussian process is a process of Gaussian distribution with infinite many variables, of which each subset follows again a Gaussian distribution. Consider the process $(f_1, f_2) \sim \mathcal{N}(\mu, \Sigma)$, then it follows $f_1 \sim \mathcal{N}(\mu_1, \Sigma_{11})$ and $f_2 \sim \mathcal{N}(\mu_2, \Sigma_{22})$, where the parameters are determined according to the marginalised distribution f_1 of the joint Gaussian distribution (f_1, f_2) (see Rasmussen et al. 2006, pp. 200). The same follows for the N noise free outputs of the training data, $\mathbf{f}$ and the N^* test outputs $\mathbf{f}^*$. Their joint distribution is given with

$$\begin{bmatrix} \mathbf{f} \\ \mathbf{f}^* \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} K(X, X) & K(X, X^*) \\ K(X^*, X) & K(X^*, X^*) \end{bmatrix}\right). \tag{2.27}$$

Here, a zero-mean function is assumed, which is a common choice if no more information about the process is known. The covariance matrix of the joint distribution consists of four partitions. $K(X, X^*)$ is the $N \times N^*$ covariance matrix, which is determined by the covariances of all pairs of training and test input values. The other partitions hold the covariances of other pairs of input values, according the matrices defined above.

To obtain the posterior distribution the joint distribution (2.27) must be restricted to the set of possible functions that fit to the training data. This can be achieved by *conditioning* the joint distribution on the observations, which results in

$$\mathbf{f}^* \mid X^*, X, \mathbf{f} \sim \mathcal{N}(\bar{\mathbf{f}}^*, \operatorname{cov}(\mathbf{f})). \tag{2.28}$$

The mean function of this Gaussian process is

$$\bar{\mathbf{f}}^* \equiv \mathbb{E}[\mathbf{f}^* \mid X^*, X, \mathbf{f}] = K(X^*, X)K(X, X)^{-1}\mathbf{f} \tag{2.29}$$

and the covariance function is given by

$$\operatorname{cov}(\mathbf{f}^*) = K(X^*, X^*) - K(X^*, X)K(X, X)^{-1}K(X, X^*). \tag{2.30}$$

The distribution (2.28) describes which observations $\mathbf{f}^*$ are likely to be made at position X^*, given the previous observations $\mathbf{f}$ made at the inputs X. If the mean and covariance function of the Gaussian process (2.28) are known, the posterior distribution can be calculated and samples obtained directly. Drawing samples from the posterior distribution means making predictions $\mathbf{f}^*$ corresponding to the test inputs X^*. The posterior distribution is sometimes also called the *predictive distribution*. (Bishop 2006; Rasmussen et al. 2006)

Observed data is usually not free of noise. Consequently, the latent function becomes $y(\mathbf{x}) = f(\mathbf{x}) + \varepsilon$, with ε being i.i.d. Gaussian noise. We denote the variance of this noise by σ_ε^2. Due to the independence of the noise, the updated covariance for two the noisy training outputs simply becomes the sum of both variance terms $\operatorname{cov}(y_p, y_q) = k(\mathbf{x}_p, \mathbf{x}_q) + \sigma_\varepsilon^2 \delta_{pq}$. Considering a whole set of N training input vectors it follows the covariance function,respectively:

$$\operatorname{cov}(\mathbf{y}) = K(X, X) + \sigma_\varepsilon^2 I.$$

Compared to the noise free case in equation (2.27), a diagonal matrix with the auto-covariance values is added to the covariance function of training data; and the joint distribution of noisy training and test data becomes

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{f}^* \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} K(X, X) + \sigma_\varepsilon^2 I & K(X, X^*) \\ K(X^*, X) & K(X^*, X^*) \end{bmatrix}\right). \tag{2.31}$$

Again, by conditioning equation (2.31) on the observations, $\mathbf{y}$, the posterior distribution can be obtained:

$$\mathbf{f}^* \mid X, \mathbf{y}, X^* \sim \mathcal{N}(\bar{\mathbf{f}}^*, \operatorname{cov}(\mathbf{f}^*)). \tag{2.32}$$

Is mean and covariance functions are given respectively by

$$\bar{\mathbf{f}}^* = K(X^*, X)[K(X, X) + \sigma_\varepsilon^2 I]^{-1}\mathbf{y}, \tag{2.33}$$

$$\operatorname{cov}(\mathbf{f}^*) = K(X^*, X^*) - K(X^*, X)[K(X, X) + \sigma_\varepsilon^2 I]^{-1}K(X, X^*). \tag{2.34}$$

The posterior distribution relates the known training data to unseen test data; it allows to make predictions. The mean function (2.33) specifies the predicted values and the and the covariance function (2.34) describes the confidence in these predictions.

However, the prior distribution specifies only vaguely the underlying function, because enough information about the underlying process is in practice not available. Therefore, the Gaussian process model, specified by solely the posterior distribution, usually models poorly the sought function.

Training a Gaussian process

In contrast to the mean and covariance function of the prior distribution, the posterior distribution depends on the data, so does the quality of the predictions. The Gaussian process model needs to be adapted to the data in order to fit well the (unknown) underlying process.

Adapting or training of a Gaussian process model means to choose the prior distribution and to learn the optimal hyperparameters of the mean and covariance function, such that the resulting model fits the data well – and hopefully the underlying function, too.

The choice of the prior mean and covariance function can be difficult and usually requires some information about principle structure of the underlying process, for example periodicity or global trend.

To learn the hyperparameters one determines the probability of the data given the model, with respect to the parameters, see appendix B. The logarithmic marginal[5] likelihood function describes this relationship, and it has an analytic solution if we assume Gaussian distributed noise

$$\log p(\mathbf{y} \mid X, \boldsymbol{\theta}) = -\frac{1}{2}\mathbf{y}^T[K(P,P)+\sigma_\epsilon^2 I]^{-1}\mathbf{y} - \frac{1}{2}\log|K(P,P)+\sigma_\epsilon^2 I| - \frac{n}{2}\log 2\pi, \quad (2.35)$$

where $\boldsymbol{\theta}$ collects the model's hyperparameters.

Optimising this function involves no further parameters, which eventually need to be found. This likelihood function automatically balances the complexity of the model in the light of the data and its uncertainty, so that it avoids overly complex models. This simplifies greatly the training of the models.

5. This likelihood function is usually attributed with the term *marginal*, since it is obtained by integrating out the not obtained observations.

Finally, the mean predictive function of the Gaussian process posterior – with already adapted hyperparameters – can be obtained by drawing samples from it. The predicted covariance function is of great value, describing the uncertainty of the predictions.

Non-zero mean functions

Above, we assume a zero prior mean function letting the Gaussian process converge to zero in the absence of training data. Although the prior covariance function essentially controls the structure of the process, since it also can account for uncertainty of the mean function, the convergence to a zero mean may lead to erroneous models.

Non-zero mean functions (for instance providing a linear or a polynomial global structure) can be incorporated in the Gaussian process regression framework. For a fixed mean function $m(\mathbf{x})$ equation (2.33) is scaled by the chosen prior mean function and becomes

$$\bar{\mathbf{f}}^* = \mathbf{m}(X^*) + K(X^*, X)[K(X, X) + \sigma_\epsilon^2 I]^{-1}(\mathbf{y} - \mathbf{m}(X)).$$

The non-zero mean function is simply subtracted from the observations, transforming them to have again a mean of zero. As the covariance function is independent of the observation equation (2.34) still holds. Once the posterior distribution is inferred the fixed zero mean (parameters known) is added back to the predictive mean function. Notice that the mean function of the data is not necessarily the mean function of the process.

The idea just described applies also for variable mean function. A set of fix basis functions is specified, whose parameters are then inferred from data. The uncertainty of the mean function's parameters can be modelled by the covariance function (see Rasmussen et al. 2006, pp. 28).

Throughout this section, we assumed the observation noise to be normally distributed. Nevertheless, as well other noise distribution are feasible in this framework. In this case the noise term of the training data ($\sigma_\epsilon^2 I$) must be replaced by an appropriate (full-rank) noise covariance matrix. Note that the marginal likelihood function will usually not have an analytic representation and that optimising the hyperparameters requires approximations, such as Laplace approximation or Monte Carlo methods.

This section is based primarily on the book by Rasmussen et al. (2006), that covers the topic in more depth.

2.4. Global satellite navigation system

The developments of positioning by satellites were military driven, as the developments of terrestrial positioning systems before. The Sputnik mission was used by US research to develop and test the basic positioning method of satellites. They used the Doppler shift of the signals emitted by the satellite and the exact knowledge about the location of signal reception to calculate the orbit of the Sputnik. Inverting that problem is a solution to determine a location on earth; if the orbit and the Doppler shift is known. In 1959 the TRANSIT program, was founded, the first (civil use) satellite positioning system. As well the Soviet Union launched a comparative system around 10 years later (Tsikada). Accuracies of both systems were in the range of 200 m to 500 m and the time-to-first-fix (TTFF) was around an hour. Moreover, only every 40 minutes a two-dimensional position estimates for static receiver could be provided. (Samama 2008)

To overcome these drawbacks they superseded TRANSIT with the NAVSTAR GPS in 1973. It is designed be operational 24 hours at 365 days per year, allowing three-dimensional position, velocity estimates, and time measurements from all over the earth. In contrast to TRANSIT, it employs ranging with digital pseudo random noise (PRN) modulated signals (Kaplan et al. 2006). In 1978 the first four (first generation) satellites were launched but it took until 1995 that the 24 nominal (second generation) satellites were in orbit and GPS could provide its full capabilities. In all this time modernisation have been implemented. That includes satellites, the signals, the complete control centre at ground (Bonnor 2012) and augmentation systems such as SBAS.

In 1976 the Soviet Union started its GLONASS program, with similar technical specification as the GPS. In contrast to GPS, it modulates the information to be transmitted on the frequency instead onto the pseudo random noise code (Bonnor 2012). The first satellite were brought into their orbits in the early- to mid-eighties and in 1995 the constellation with the 24 nominal satellite was established (Samama 2008). Due to the short life time of the satellites the full constellation was not operational until 2011. Glonass as well was modernised. A particular issue was the compatibility to GPS and GALILEO. Since 2011 the new satellites transmit in

addition code-modulated signals.

The European Union decided 1998 to launch a own satellite positioning system for world-wide civilian use: GALILEO. One of the main aims of GALILEO is to be interoperable with GPS (Kaplan et al. 2006). That regards the design of signal structure, coordinate reference frame, time reference system. In addition to the 'Open Service' for positioning other services are part of the GALILEO design, among others 'Search and Rescue Service', 'Public Regulated Service' (encrypted signals). It was planned that the full GALILEO constellation of 30 satellites is available in 2020 (Bonnor 2012).

Three further systems are under construction, they are regional navigation satellite systems; not covering the whole globe. The Chinese BeiDou system, in particular the BeiDou-II system because only this is designed to provide a real-time three-dimensional position fix. A sixteen satellite constellation will provide position service for the Asian-Pacific region. The second one is the Japanese QZSS. It aims for communication (media) and positioning services. It is not able to provide a position fix on its own. The last we mention is the INRSS, a positioning system by the Indian government. The INRSS will consist of seven satellite providing positioning service for the Indian subcontinent. (Bonnor 2012).

2.4.1. Principle of operation

The mentioned GNSS differ in specification and implementation due to physical limitations (space, frequency). Respecting the announced efforts to make GNSSS more compatible they (will) obey the same principle of operation. We intend to provide a general overview about GNSS instead of one or all very specific. We rather state ranges than single values and concentrate on information important for position estimation. For more detailed information about each GNSS is referred to (Kaplan et al. 2006; Samama 2008).

The three globally operating GNSS have constellations of 24 and 30 satellites, in medium earth orbits between 25 000 km to 30 000 km to provide at least four satellites at almost each place on earth (Groves 2008). Each satellite broadcasts its orbital information (ephemerides), its health information and time information. This data is referred to as 'navigation message' and it is transmitted on a the same communication channel for all satellites. Among the GNSSS different signals and at least two[6] carrier frequencies are used. These frequencies are between 1.1 GHz

6. A civil/open and a military/encrypted signal.

to 1.6 GHz.

To calculate a position trilateration is employed. Therefore ranges from different satellites must be obtained by the receiver. These ranges are computed from the time the signals need to propagate from the satellite to the receiver. To use these time measurements the satellites clock must be and are synchronised to a common time reference.

One of crucial concepts of GNSS is characterisation of the signals which allows to identity each satellite, while the signals use the same spectrum, and to measure the propagation time of the signal. The code division multiple access (CDMA) scheme that is applied provides properties to fulfil both objectives.

The navigation message is combined with the PRN sequence – still being a PRN sequence, because the low bitrate of the navigation messages does not change the noise character much. The use of PRN codes lets the signal look like noise, having the advantage that the energy of the information is distributed over the spectrum making it less susceptible to interferer signals. The PRN sequences are chosen such that they are orthogonal to each other and that each satellite is assigned one of the set of orthogonal sequences. In the satellite the PRN sequence is then periodically modulated onto the carrier signal and transmitted.

These signals arrive at the receiver where they are unmodulated such that the PRN sequence can be processed. It is correlated with local copies of the PRN codes residing in the receiver. Due to the orthogonality of the sequences the correlation peaks when the correct replica of the received sequence is (auto-)correlated, that way identifying the transmitter. If other sequences are correlated the correlation function stays flat[7].

The correlation as well allows to measure the propagation time. Assume also the receiver clock is synchronised and that the transmit time of the signal is already decoded. To correlate the signal the receiver has to find its beginning. Correlating consists then to shift the local replica of the sequence such that both sequences align. The time the code was shifted corresponds/equates the time the signal travelled. Multiplying this time by the speed of light yields the range from the satellite to the receiver. This range comprises an additional error source, the receiver clock and the satellites are actually not synchronised. The differences in time between the GNSS receiver and the reference time of the satellites. The

7. GLONASS actually employs frequency division multiple access, but is changing this in favour of CDMA (Bonnor 2012)

distance between the receiver position, $\mathbf{p}$ and the position of the jth satellite $\mathbf{p}^j$ is expressed as their geometric distance plus the receiver clock offset, t_r,

$$\begin{aligned}\varrho^j &= \|\mathbf{p}_s^j - \mathbf{p}\| + c t_r + \eta^j \\ &= \sqrt{(\mathsf{x}_s^j - \mathsf{x})^2 + (\mathsf{y}_s^j - \mathsf{y})^2 + (\mathsf{z}_s^j - \mathsf{z})^2} + c t_r + \eta^j,\end{aligned} \tag{2.36}$$

where η^j aggregates further error sources, see (Kaplan et al. 2006, ch. 6 & 7). Equation (2.36) states a single *pseudorange*.

Four satellite-to-receiver ranges form an equations system that can be solved to obtain the triple x, y and z, the coordinates of the receiver and also the receiver clock offset.

Trilateration

Relying on the geometry of triangles GNSS compute the users position. Here, each satellite's position at the time of signal transmission is known (transmitted in the navigation message); three satellites constructing a triangle, each being a corner. A satellite-receiver distance forms a sphere around a satellite. In three-dimensional space these two spheres intersect in a circular plane that is perpendicular to a line connecting the two satellite. Based on only two pseudorange measurement the receiver position is one of the positions forming the perimeter of the circle, see figure 2.2. Consider an additional sphere around a third satellite. The third sphere intersects with the perimeter of the circle in two points, as can be seen in figure 2.3; narrowing down the possible solutions. One excludes the solution that does not lie on the earth's surface and the left point is the sought receiver position. Of course, the pseudoranges are affected by errors. Therefore the intersection point is rather a spherical triangle kind intersection zone.

In the case of GNSS a fourth pseudorange is necessary because the receiver clock offset t_r introduces a fourth unknown variable that is common to all pseudoranges.

Position, Velocity, Time algorithm

The solution to the navigation equations above is commonly abbreviated with position, velocity and time (PVT), naming the important quantities one wants to known from a GNSS receiver. Several methods to determine a position fix algorithmically exist. In general the equations of pseudoranges are linearised

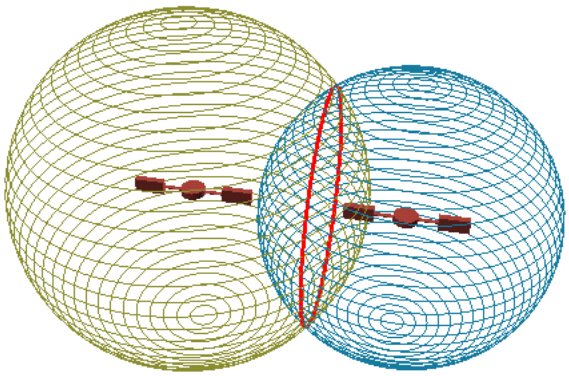


Figure 2.2.: Intersection of spheres constructed by two satellite-receiver ranges. Narrowing down the possible solutions to the circle of intersection. ('Keep Environment Nature's Blog' 2008)

around an approximate position and a predicted clock offset, that results system of linear equations relating changes of the user's position to changes of the pseudoranges. One actually solves for a correction to the initial estimate of the user position and clock offset (Kaplan et al. 2006). The position estimate is obtained by adding the correction on the initial values. The accuracy of this solution depends on the first approximations of the positions. Repeating the calculation using the last solution and recent pseudorange measurement may refine the position estimate. In practice, ten iterations are usually sufficient for the algorithm to converge (Samama 2008).If more than four pseudoranges were obtained – nowadays the case in most environments – the least squares or weighted least squares approach is used, since it is more accurate then just using four measurements.

Bancroft (1985) developed a closed-form solution to the problem. He transformed the time of arrival (TOA) problem to a time difference of arrival (TDOA) problem by calculating with differences of pseudoranges. That results in a quadratic equation to that the solution is easily obtained. The advantage of this approach is that it does not need initial values.

The calculation of PVT is usually implemented in iterative manner able to continuously determine a position estimate.

Additional information may provide further improvements for the PVT estima-

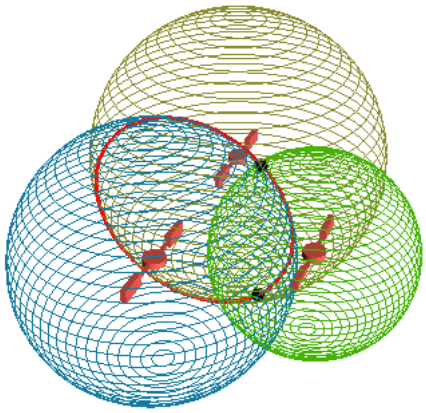


Figure 2.3.: Intersection of spheres constructed by three satellite-receiver ranges. The three spheres intersect in two points (black dots), forming the possible solutions. ('Keep Environment Nature's Blog' 2008)

tion. By means of an extended Kalman filter the history of the measurements and a model of the motion of the receiver can be incorporated into the estimation procedure; advantageous especially in dynamic use cases. A detailed description about the GNSS Kalman filter can be found in (Groves 2008; Kaplan et al. 2006).

2.5. WLAN based localisation

WLANS are radio frequency communication networks according to the standard IEEE 802.11. They provide wireless access to computer networks in at in the INDUSTRIAL, SCIENTIFIC AND MEDICAL frequency band at 2.4 GHz and 5.8 GHz. The access to a WLAN is limited by its outreach of up to 100 m dependent on the environment. The nodes of WLANS are known as access points.

WLANS offer some compelling advantages with respect to location determination. These in particular are a world-wide availability of access points, millions of WLAN enabled devices, wall penetrating signals. The localisation infrastructure and end-user devices are off-the-shelf, standardised and already widespread allowing almost everyone to develop a WPS. These properties made this technique the most common one for indoor localisation. That the WLAN signals penetrate solid objects makes them also useful in outdoor environments that is covered by

WLAN infrastructure; therefore existing WLANs often cover urban areas completely and that makes them potentially a large area position system. Companies make already use of that and provide localisation services based on WLAN in large areas (awiloc 2013; skyhookwireless 2013). The wide proliferation of access points in denser inhabited areas provides a sufficient number of access point for WPSs. Vaupel et al. (2010) state an average of 12.6 access points in an ordinary dense urban centre.

All positioning techniques mentioned in section 2.1.1 have been explored for WPS. But propagation-time based methods suffer from non-line-of-sight (NLOS) conditions (Galler et al. 2006; Koenig et al. 2011; Pahlavan et al. 2006) and the missing time synchronisation bear additional problems (Bagdonas et al. 2008; Günther et al. 2005; Li, Pahlavan et al. 2000; Yamasaki et al. 2005) diminishing the position accuracy. An other shortcoming is that accurate access point locations must be known, which is hard considering public areas where access points are often moved difficult to to survey. Systems that are based on proximity neither possess sufficient accuracy. The left over localisation technique is – the one we also will employ – WLAN location fingerprinting.

2.5.1. WLAN RSSI location fingerprinting principle of operation

Location fingerprinting's fundamental concept is that a metric or set of metrics is unique at every position and that a certain location can therefore be identified by that metric. For localisation in WLANs the RSSI is the signal metric most commonly chosen, as do we – but note that the assumption of spatial uniqueness of RSSI is flawed in reality.

As any localisation system as well for location fingerprinting needs some spatial references, which need to be created. That is done in a step prior to the localisation. This step is often called survey- or offline phase. Objective of this phase is to create a 'map' that comprises all the RSSI of all access points at all positions: the *radio map*. In practice the developer/operator of the system moves to a position, reads that position for example from a (digital) map, measures the RSSI, of all receivable WLAN access points and stores the RSSI together with the position in a database. This procedure is then repeated for many reference points. Each entry of database is referred to as *fingerprint*. To determine a position from such a system one measures again the RSSI, now at the unknown location, and compares these observations with the entries contained in the database.

The position corresponding the closest RSSI match is the sought position. This method was first published by Bahl et al. (2000). Each RSSI is identified via the media access control (MAC) address which is transmitted with the signal. This identifier is associated with each RSSI reading and also stored in the database. In practice Bahl et al. (2000) did not store a single RSSI reading in the database, but a time average. Also note that standard WLAN transceiver chip sets output a RSSI quantised to one dBm.

2.5.2. Modelling WLAN location fingerprinting

Received signal strength

Rssi are mainly determined by the attenuation that a radio-frequency signals experiences propagating through the atmosphere, in free space without encountering objects, interference by other signals, etc. it is described by free-space path loss model $P_{\text{fs}} = -10\gamma \log\left(\frac{\lambda}{4\pi d}\right)$. Where λ is the wavelength, γ the path loss exponent ($\gamma = 2$ in free space) and $d = \|p_T - p_R\|$ is the distance between transmitter and receiver.

In urban- and indoor scenarios this model holds only to a certain extend, since mostly the LOS between transmitter and receiver is obstructed. Rssi are additionally affected by fading, describing changes of signal power due to changes of the transmitter-receiver geometry; for examples when transmitter and/or receiver are in motion. The change of the signal path due to obstacles results in reflection, diffraction and scattering (Rappaport 2001), these are large-scale effects summarised by *shadowing*.

A third phenomena that affects the signal power in urban areas is called small-scale fading. It is due to changes in the multipath geometry and changes in velocity of the transmitter and/or receiver (Rappaport 2001). Therefore several copies of the signal are received at slightly different times, causing interference and the signal power variations. Small-scale fading occurs in relatively small time periods. Since WLAN RSSI are time averaged signal power measurements the attenuation due to fading can be neglected.

The large-scale effects attenuate the signal power logarithmically with distance, leading to the *log-distance path loss model* (Rappaport 2001). Various other suggestions to model the signal attenuation in complex areas exist, but most of are based on the log-distance path loss model. Additional time varying influences

are modelled as a random process with log-normal distribution. The path loss in urban and indoor environments is given by

$$P_{\text{ld}} = P_{\text{fs}}(d_0) + 10\gamma \cdot \log_{10}\left(\frac{d}{d_0}\right) + w_{\text{RSSI}}. \tag{2.37}$$

P_{fs} is the free-space path loss at a reference distance of $d_0 = 1\,\text{m}$. γ is the path loss distance exponent and w_{RSSI} is a noise term reflecting for log-normal shadowing (in dB). Given a transmitter emitting the signal with the power of P_T, the averaged signal power received at the receiver is described by

$$s = P_T - P_{\text{ld}}. \tag{2.38}$$

WLAN RSSI are signal strength, whose mean satisfies equation (2.38).

Due to variability and complexity of dense urban and indoor environment (geometry, humidity, moving persons/objects, a user's orientation, radio frequency inferences, etc.) of the WLAN radio propagation channel the path loss models are not accurate enough to model RSSI and vice versa (Chitte et al. 2009; Elnahrawy et al. 2004; Heurtefeux et al. 2012). The path loss exponent varies in dependence of the environment's geometry and materials and the LOS/NLOS conditions. Many different path loss models, advantageous only for particular scenarios, have been developed (Rappaport 2001); a unifying model for general environments is still lacking (Hatami et al. 2006; Mardeni et al. 2012). Therefore the ranges of the parameters be quite large: the path loss exponent γ ranges from 2 dB to 6 dB (Kushki 2009) and the standard deviation $\sigma_{w_{\text{RSSI}}}$ varies between 3 and 16 dB (Rappaport 2001). Such dynamics may even occur over short distances. To overcome the shortcoming of the propagation models the empirical WLAN location fingerprinting became the technique of choice.

Accounting for the random nature of RSSI, we denote the random process of RSSIs by $\{s_t, t \in \mathcal{T}\}$, on the signal space $s_t \in \mathcal{S}$. We like to point out that s_t is in general not stationary (Kaemarungsi et al. 2004b). Only during the relatively short RSSI capturing intervals, at the particular position, and relatively fix conditions of the environment, we assume stationarity. This process is also spatially correlated, thus depends on location; but for the sake of convenience we neglect this dependence. We will come back to it as it is import for the interpolation of signal strengths we will discuss later.

Fingerprints and radio map

Since the fingerprints are taken on discrete positions to establish the radio map. These discrete positions are defined on the space $\check{P} \subseteq \mathbb{R}^3$, which consists of $\mathrm{dom}(\check{P}) = \check{\mathbf{p}}^1 \cup \check{\mathbf{p}}^2 \cup \cdots \cup \check{\mathbf{p}}^{n_m}$ regions of equal volume, n_m denoting the number of regions. Each region has a volume $|\check{\mathbf{p}}^m|$ that can be assigned a probability. Presuming that the volumes $|\check{\mathbf{p}}^m|$ are equal size, a region can be approximated by its centroid $\bar{\mathbf{p}}^m$, see (Thrun et al. 2006).

Lets define a fingerprint as $S^m \triangleq \mathsf{F}(\bar{\mathbf{p}}^m)$, $S^m \in \mathcal{S}$ being a (set of) RSSI observed at the mth reference position $\bar{\mathbf{p}}^m = (\bar{\mathrm{x}}^m, \bar{\mathrm{y}}^m, \bar{\mathrm{z}}^m)^T$ in Cartesian coordinates. Each set of RSSI consists of possibly various RSSI from $n_{l,k}$ access points $S^m = \{\mathbf{s}^{m,l}\}_{l=1}^{n_{l,k}}$, whose elements $\mathbf{s}^{m,l} = (s_1^{m,l}, \ldots, s_{n_{s,k}}^{m,l})^T$ are realisations of s_t.[8]

For the sake of convenience we assume the number of receivable access points at a specific reference position as constant in time ($n_{l,k} = n_l$), as well as the number of of signal strength values received at a that reference position ($n_{s,k} = n_s$).[9]

A fingerprint in its most general definition is a set of PDFs – one for each access point – specific for a given location (Mirowski, Whiting et al. 2012), $p(\mathbf{s}^{m,l} \mid \bar{\mathbf{p}}^m)$. That is, $\mathbf{s}^{m,l}$ may contain sufficient data to construct a histogram of RSSI received from one access point approximating the underlying PDF. The classic WLAN fingerprinting method (Bahl et al. 2000) uses time averaged signal strength for fingerprinting; $\bar{s}^{m,l} = 1/n_s \sum_{q=1}^{n_s} s_q^{m,l}$ reduces to a scalar. The complete fingerprint database is then defined by

$$\mathcal{R} \triangleq \{\bar{\mathbf{p}}^m, S^m\}, \quad m = 1, \ldots, n_m.$$

Position estimation

The fundamental problem of WLAN fingerprinting is to deduce a position from the radio map while observing RSSI measurements from various access points. To estimate a position, when a set of RSSI is measured is to match the observed $S_k = \{\mathbf{s}_k^l\}_{l=1}^{n_l}$ with the database entries S^m; a classical pattern recognition problem.

8. The time index k is used for the cycles of the sequential Bayesian estimator and for observations that fall within one of these cycles.

9. Then number of access points and number of received RSSI values may be a design parameter making these two assumptions reasonable. Advanced models exist that consider in particular the issue of changing number of receivable access points, see (Beder et al. 2012; Mirowski, Steck et al. 2011).

One seeks to minimise a measure capturing the divergence/similarity of the chosen pattern or signal metric

$$\hat{\mathbf{p}}_k = \arg\min_{\bar{\mathbf{p}}^m} \left(d(S_k \mid \bar{\mathbf{p}}^m, S^m) \right).$$

This similarity indicates how likely it is to receive the observation at each of the reference positions.

To discriminate or compare between the current measurement and the fingerprints in the database many metrics are available. The classical fingerprinting for example uses the Euclidean distance in signal space. Assuming the number of access points and as well the number of measurements from each access point the same, the similarity between the fingerprints and the observation can be given by

$$d(S_k \mid \bar{\mathbf{p}}^m, S^m) = \sqrt{\sum_{l=1}^{n_l} (\mathbf{s}_k^l - \mathbf{s}^{m,l})^2}. \tag{2.39}$$

One can improve this estimator by finding the k closest fingerprints

$$\{\mathbf{p}^{m'} \mid \{\bar{\mathbf{p}}^n\}_{n=1}^{n_n - K} : d(S_k \mid \bar{\mathbf{p}}^n, S^n) \geq d(S_k \mid \bar{\mathbf{p}}^m, S^m)\}, \quad m' = 1, \ldots, K$$

and averaging over the corresponding positions

$$\hat{\mathbf{p}} = \frac{\sum_{m'=1}^{K} w^{m'} \bar{\mathbf{p}}^{m'}}{\sum_{m'=1}^{K} w^{m'}}, \quad \text{where } w = 1/K. \tag{2.40}$$

This algorithm is the k-nearest neighbours (k-NN) method. When weighting nearer neighbours more heavily by replacing the weights in equation (2.40) with the inverse of the divergence measure $w^m = d\left(S_k, \mid \mathbf{p}^m, S^m\right)^{-1}$, a slightly better estimator – the weighted k-NN method – is obtained.

A stochastic approach to this is to use the RSSI likelihood function. A large likelihood reflects a high similarity and a low likelihood a high divergence.

In the probabilistic framework the RSSI observations from different access points are assumed to be independent of each other [discuss independence assumption]. If the distribution of the noise w_{RSSI} is known, the joint PDF for independent random variables is given by the product of the likelihood functions of each

access point

$$
\begin{aligned}
p(\mathbf{s}_k^1, \ldots, \mathbf{s}_k^{n_l} \mid \bar{\mathbf{p}}^m) &= \prod_{l=1}^{n_l} p(\mathbf{s}_k^l \mid \bar{\mathbf{p}}^m) \\
&= \prod_{l=1}^{n_l} p_{\mathbf{w}^{m,l}}(\mathbf{s}_k^l - h_s^l(\mathbf{p}^m)) \\
&= \prod_{l=1}^{n_l} p_{\mathbf{w}^{m,l}}(\mathbf{s}_k^l - \mathbf{s}^{m,l}) \\
&\triangleq p(S_k \mid \bar{\mathbf{p}}^m).
\end{aligned}
\tag{2.41}
$$

The function $h_s^l(\bar{\mathbf{p}}^m)$ stands for the (empirical) measurement model of the lth access point and is determined by the mth radio map entry. A position estimate ($\hat{\mathbf{p}}$) can be obtained by the ML method (2.4).

The similarity measure clearly depends on the model of the RSSI likelihood and what data is stored in the radio base. For parametric models the 2-norm (see equation (2.39)) is the most widely applied measure of similarity between the RSSI fingerprints and the observed signal strength. Other options to compute the similarity in signal space are referred in (Corte-Valiente et al. 2009; Honkavirta et al. 2009). To compare the similarity between complete PDFs, hence including higher moments, either estimated by histograms or kernel density estimators (Roos et al. 2002) other kernel regression versions (Kushki et al. 2007; Le Dortz et al. 2012; Mirowski, Steck et al. 2011; Pan et al. 2006). As well Honkavirta et al. (2009) lists various possibilities to compare PDFs. Further methods to recognise the observed signal strength pattern within the radio map include support vector regression (Battiti et al. 2002), decision trees (Yim 2008) and neural networks (Fang et al. 2008).

Concluding remarks

The more information is stored in the database the more details can be explored from the RSSI uncertainty. This enables to extract more feature(s) that can be used for regression, allowing a more accurate recognition of that feature(s) and more better positioning. The amount of RSSI samples, that is the amount of information,

depends on the design of the system and on the labour invested to generate the radio map. The density of database reference positions is likewise important.

One system design issue influencing the positioning performance is the RSSI sample rate and the time available to observe the RSSI. The sample rate of standard WLAN transceiver chips is 1 s. For example a pedestrian moving with for example $1\,\mathrm{m\,s^{-1}}$ may only receive one observation at a position or may be up to five RSSI observation in the area of that position. Therefore in realistic scenarios it is unlikely that enough samples for a complete histogram can be collected. This loss of information make parametric approximations of the likelihood more useful even though they discard possibly useful information (Honkavirta et al. 2009; Mirowski, Whiting et al. 2012).

An other point to consider when designing a WPS is the average distance of the database reference points. In general, the closer the reference positions to each other the more exact the positioning system. But there is a limit that depends on the environment. In spaces with few obstructing objects the signal strength at close by positions differ little, hence it is useless in these areas to create fingerprints overly dense. In addition several authors studied how the radio map generation can be simplified. Three approaches could be found: Either by crowdsourcing as in (Park 2013), or by automation (Palaniappan et al. 2011), or by interpolation (Atia et al. 2012).

The number of access points received in survey phase and positioning phase is a further point to consider. Ideally, this number should coincide in both phases; but often it does not, due to a small time window receiving RSSI or may be moved or off-line access points. A frequent remedy is to choose only the strongest access points. Whereas Beder et al. (2012) and Kushki et al. (2007) and Mirowski, Steck et al. (2011) incorporated this problem in their algorithms.

All these considerations are relative and needed to be considered in the light of the overall system/WPS design, including the environment (variability, size of open spaces, etc.) and of course use cases.

Chapter 3

Gaussian process WLAN positioning system

This chapter describes a WLAN positioning system based on the concept of fingerprinting. Compared with classical fingerprinting, which infers the sought position from the spatially discrete fingerprints, we use the Gaussian process regression approach described in section 2.3 to interpolate the RSSIs. This brings two additional advantages: First, due to the interpolation of fingerprints the amount of fingerprints that is needed for a certain positioning accuracy can be reduced without loosing accuracy, thus facilitating largely the creation of radio maps; second, one can use this interpolated radio map to compute a RSSI likelihood function on a continuous state space. This facilitates the integration of RSSI into the data fusion framework (2.10), our main objective, but it also enables probabilistic fingerprinting localisation. Wherefore we use the ML estimator to infer a position from RSSIs readings.

Even though the overall objective is the hybrid localisation system, the WLAN-only position estimates are helpful to assess its accuracy and precision, providing a sense about the details about the fusion with GNSS. The following section presents the probabilistic WLAN RSSI model, that will also be used to fuse RSSI with pseudoranges. Section 3.3.2 describes the experimental setup used to create the radio map. Gaussian process regression model are very flexible methods, however this flexibility brings as well responsibility in choosing the 'right' model. Therefore, conduct a comparative analysis in finding the 'right' model. In section 3.4 we compare several Gaussian process models to find the one that suites WLAN RSSI best. Furthermore, we investigate if Gaussian process models

for indoor and outdoor environments can be distinguished. This is followed by results on Gaussian process regression based ML estimator derived from the best suited Gaussian process model.

3.1. WLAN RSSI model

The following sections describe the interpolation of RSSI with Gaussian process regression in order to form a likelihood function on continuous state space. The fundamentals of regression with Gaussian processes can be found in section 2.3.

Interpolating RSSI values corresponds to predicting RSSI in space. That was already studied before because this means an increase of resolution of the WLAN radio map. This idea was already studied by Elnahrawy et al. (2004) who applied Delaunay triangulation to interpolate between fingerprints, whereas Chai et al. (2007) interpolated linearly to obtain additional signal strength values. A large body of studies use some regression method as the very Gaussian processes regression models (Atia et al. 2012; Bekkali et al. 2011; Duvallet et al. 2008; Ferris et al. 2006; Li, Wang et al. 2005; Yoo et al. 2014). Benefits of this procedure are a continuous approximation of the spatial RSSI distribution – facilitating the integration into the Bayesian fusion framework – and largely reduced labour of fingerprint database creation. The disadvantages are the increase of complexity of the system and the additional computational burden.

Section 2.3 provides the fundamentals to understand the WLAN RSSI model that will be presented in section 3.1.2. In section 3.2 we outline the use of the RSSI likelihood function to estimate a location with the ML technique. As modelling of WLAN RSSI has been controversial we first state the different approaches of probabilistic RSSI modelling, justify our choice and then present the Gaussian process regression based RSSI likelihood function.

3.1.1. On the Gaussianity of WLAN RSSI

The RSSI model function is supposed to properly reflect the statistics of RSSI and to describe the relation of signal power and space. RSSIs are first of all governed by the environment and all the fluctuations and disturbances occurring in that environment – causing temporal and spatial variations of the statistical properties of RSSIs. In addition, the RSSI PDF varies dependent on receiver antenna orientation

and may even change completely for different receiving hardware. An accurate and universal model of the WLAN signal strength distribution is still an open issue.

Based on the log-distance path loss model (2.37), or one of its variations, it is commonly assumed that the logarithmic RSSI are normally distributed due to large-scale shadowing that affects them (Rappaport 2001). This assumption holds only now and then (Kaemarungsi et al. 2004b, 2012; Ladd et al. 2005; Le Dortz et al. 2012; Mirowski, Steck et al. 2011; Vaupel et al. 2010). RSSI PDFS are frequently left-skewed, sometimes they are also be bi- or multimodel, hence rendering the assumption of Gaussianity flawed.

Considering that RSSI are quantised, the normalised centralised histogram models the signal strength likelihood $p(s \mid p)$ quite natural. Chai et al. (2007), Mirowski, Steck et al. (2011) and Roos et al. (2002) chose this approach, but capturing a full histogram to completely approximate the RSSI distribution requires a relatively large amount of samples, implying enough time. This temporal constraint, an increased database size and increased computational burden, which additionally leads to higher device power consumption, makes this method inexpedient.

A different but also non-parametric approach is to approximate the RSSI likelihood by kernel density estimation. Kushki et al. (2007), Pan et al. (2006), Park et al. (2011) and Roos et al. (2002) did this, where the squared exponential kernel[1] is the kernel mostly used. This method add an additional step of complexity to the overall positioning system making it for the current work not appropriate.

In contrast to capture a histogram capturing a single RSSI is clearly not robust enough, nonetheless – the middle ground – capturing averages is a working approach as classical, deterministic fingerprint methods have shown. Capturing the mean and a variance is enough to model a Gaussian likelihood distribution. The majority of studies on this topic followed this approach, for example (Chen et al. 2007; Ferris et al. 2006; Li, Salter et al. 2006; Wallbaum et al. 2004; Youssef et al. 2003). More robust or more general distributions have been suggested too: For example a Gaussian mixture (Tseng et al. 2008), contaminated Gaussian with LOS and NLOS component (Yin et al. 2013), t-distribution (Elnahrawy et al. 2004), Weibull distribution (Chan et al. 2012), central χ^2-distribution (Kaemarungsi et al. 2004a) and a Gaussian enhanced with a Gibbs distribution accounting for missed access points (Beder et al. 2012).

1. Often also referred to as radial basis or Gaussian kernel.

All models suffer from the variability of the WLAN propagation channel. In practical systems RSSIs are recorded during short time periods, in fact in the survey and as well positioning phase. Different approaches are more robust than others, but that depends on the context and design of the WPS. Given that the RSSI are not stationary over medium and long time periods it is always uncertain if the observations made during the survey phase share the properties of the observations at a later time. This holds for the simple approaches as well as for the more complex ones. Measurement redundancy countervails modelling issues to a certain degree.

Albeit parametric models do not capture well the RSSI PDF, even less if the distribution is skewed or multimodel, several authors consider a Gaussian RSSI distribution a trade-off between complexity and feasibility and reported reasonable results Chen et al. (2007), Elnahrawy et al. (2004), Kaemarungsi et al. (2012), Mirowski, Whiting et al. (2012) and Youssef et al. (2003). We fall into line with that argumentation: We use the normal distribution to model the RSSI in the positioning context (eq. (2.10b)) and also for the interpolation of RSSIs with Gaussian processes (eq. (2.25)). Using a normally distributed Gaussian process likelihood function leads moreover to an analytically solvable regression. More details on this choice can be found in (Richter and Toledano-Ayala 2015).

For more insights about the statistical properties of WLAN RSSI in indoor environments we refer to (Kaemarungsi et al. 2012).

3.1.2. Interpolating the WLAN radio map

In section 2.5.2 we defined a reference point of the database $\bar{\mathbf{p}}^m$ on discrete subspace $\check{P}$ of $\mathbb{R}^3$. In our implementation of the WLAN positioning system we have only two-dimensional reference positions which is why we reduce the position space to $\bar{\mathbf{p}}^m \in \check{P} \subseteq \mathbb{R}^2$ as compared to section 2.5.2. (We recommend this section and section 2.3 because we use the same notations).

An analytic relationship between RSSI and space is not known though, but consider the following equation to represent this relation $s = f(p) + \epsilon$, where $f(\cdot)$ is the function describing this underlying relationship and which we are after because it allows us to predict further values. The noise, ϵ, is uncorrelated and stems from a normal distribution (see discussion above) with standard deviation σ_ϵ.

When creating a fingerprint database we sample at known locations from

this relation and get for each fingerprint a set of noisy RSSI measurements, $\mathbf{s}^m$, observed at some reference positions $\bar{\mathbf{p}}^m$. Thus, creating the whole radio map with n_m fingerprints is described by

$$\mathbf{s}^m = f(\bar{\mathbf{p}}^m) + \boldsymbol{\epsilon}, \qquad m = 1, \ldots, n_m.$$

Notice that $\mathbf{s}^m$ are the RSSIs of a single access point. That means that also the regression procedure is done for each access point separately. The index of access point, l, is therefore omitted here for convenience.

Since we average the signal strength over a few seconds we denote the RSSI that we observed from the same access point by $\bar{s}^m$. If we used the whole fingerprint database as training data we write the training data as $\mathcal{D} = \{\mathbf{p}^m, \bar{s}^m\}_{m=1}^{n_m}$, but in general one may only use $m \leq n_m$ data points, why we use the notation $\mathsf{P} \triangleq \{\mathbf{p}^m\}$ for the training input values and $\mathsf{s} \triangleq \{\bar{s}^m\}$ for target values.

The measurements $\{\bar{s}^m\}_{m=1}^{n_m}$ are spatially correlated (to a certain extend, as this depends on the distance of the corresponding reference points) which is expressed by their covariance function $\mathrm{cov}(\mathsf{s})$. The covariance needs to be specified by choosing a suited kernel, $K(\mathsf{P}, \mathsf{P})$. This is one of the crucial choices when determining a Gaussian process model. The covariance function of the noisy training targets becomes simply $\mathrm{cov}(\mathsf{s}) = K(\mathsf{P}, \mathsf{P}) + \sigma_\epsilon^2 I$. The mean function of that process is denoted $\mu(\mathsf{s})$. Hence, the radio map creation underlying Gaussian process follows a multivariate normal distribution $\mathsf{s} \sim \mathcal{N}\big(\mu(\mathsf{s}), \mathrm{cov}(\mathsf{s})\big)$.

Now we further restrict the noise to follow a zero mean normal distribution, $\epsilon \sim \mathcal{N}(0, \sigma_\epsilon)$, and assume we further sample RSSI, $\mathbf{s}^* \doteq f(P^*)$; this time at unknown locations P^* – speaking of our test data $\mathcal{D}^* = \{P^*, \mathbf{s}^*\}$. These measurements are the RSSI we want to predict. As both sets of observations origin from the same physical process they are jointly distributed and spatially correlated. (neglecting temporal variations between the times of measurement taking). The correlation between the two sets of RSSI is expressed by a covariance matrix that determines the correlation between training and test data pairs, given by $K(\mathsf{P}, P^*) = K(P^*, \mathsf{P})^T$, and the covariance matrices of the training target values $K(\mathsf{P}, \mathsf{P})$ and the same for test data $K(P^*, P^*)$ respectively. The joint RSSI distribution yields

$$\begin{bmatrix} \mathsf{s} \\ \mathbf{s}^* \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{m}(\mathsf{P}) \\ \mathbf{m}(P^*) \end{bmatrix}, \begin{bmatrix} K(\mathsf{P}, \mathsf{P}) + \sigma_\epsilon^2 I & K(\mathsf{P}, P^*) \\ K(P^*, \mathsf{P}) & K(P^*, P^*) \end{bmatrix} \right).$$

Based on the joint distribution, the posterior Gaussian process distribution is, as before, obtained by conditioning over the observations, resulting in a normal distribution

$$p(\mathbf{s}^* \mid \mathcal{D}, P^*) = \mathcal{N}\big(\mu(\mathbf{s}^*), \mathrm{cov}(\mathbf{s}^*)\big). \tag{3.1}$$

Its mean function reads,

$$\mu(\mathbf{s}^*) = \mathbf{m}(P^*) + K(P^*, \mathsf{P})[K(\mathsf{P}, \mathsf{P}) + \sigma_\epsilon^2 I]^{-1}(\mathsf{s} - \mathbf{m}(\mathsf{P})) \tag{3.2}$$

and covariance function

$$\mathrm{cov}(\mathbf{s}^*) = K(P^*, P^*) - K(P^*, \mathsf{P})[K(\mathsf{P}, \mathsf{P}) + \sigma_\epsilon^2 I]^{-1} K(\mathsf{P}, P^*), \tag{3.3}$$

respectively. To predict/interpolate further RSSIs at arbitrary locations one just uses equation (3.2) whereupon corresponding variances are provided by the covariance function. The prediction distribution models the randomness and the spatial correlation of RSSIs. The section that follows makes use of this distribution by constructing a ML estimator that is able to infer the location of the object.

For the particular choice of the mean and covariance function we refer to section 3.4.

3.2. Position estimation

Consider an object/user that wants to determine its position based on an observed RSSI average $\bar{s}_k$ from still only one access point. (The particular number of the signals strength values that is received at time k is of no interest as we average them, as we have done for the radio map as well.)

Since normal distributed noise was assumed, we also use a Gaussian distribution to denote the likelihood function (now of the positioning context):

$$p(\bar{s}_k \mid P^*) = \frac{1}{\sqrt{2\pi}\sigma_{\mathbf{s}^*}} \exp\left(-\frac{1}{2}\frac{\big(\bar{s}_k - \mu_{\mathbf{s}^*}\big)^2}{\sigma_{\mathbf{s}^*}^2}\right)$$

where we denote $\mu_{\mathbf{s}^*}$ and $\sigma_{\mathbf{s}^*}^2$ the value of the mean and covariance functions (eq. (3.1)) that corresponds to the test position under consideration. For each of the test inputs it determines the probability that $\bar{s}_k$ was observed at the particular

location.

The variance should actually reflect the uncertainty of the measurement at given location. When we captured the WLAN data we estimated the variance with the same samples that are used to calculate the RSSI mean. During the observation interval of a few seconds we unfortunately obtained quite often only a single RSSI from an access point, not allowing the estimation of a deviation measure. Therefore, we modelled the uncertainty with the predicted RSSI variances. With this approach the variance reflects the uncertainty of the measurement only if the measurement condition during radio map recording and during measurement reception are equal. This is an unjustified assumption which, nevertheless, works in practice.

The likelihood function is actually computed on a regular grid of locations which corresponds to the test positions. For this case only, the mean function equals the mean values ($\mu(\mathbf{s}^*) \equiv \mu_{\mathbf{s}^*}$) and the covariance function equals the variance ($\text{cov}(\mathbf{s}^*) \equiv \sigma^2_{\mathbf{s}^*}$) values of the likelihood function.

Consider now an object/user that wants to determine its position based on observed averaged RSSI $\bar{s}^l_k$ from $l = 1, \ldots, n_l$ different access points. For this situation, let the complete set of RSSI observation be $S_k = \{\bar{s}^l_k\}_{l=1}^{n_l}$. Measurements from access points that are not contained in the radio map are simply ignored and vice versa, so that the number of RSSI observed from different access points and the corresponding number of a fingerprints become equal $|S^*| = |S_k|$. The joint likelihood function becomes the product of the likelihood functions of each access point

$$\begin{aligned} p(S_k \mid P^*) &= \prod_{l=1}^{n_l} \frac{1}{\sqrt{2\pi}\sigma^l_{\mathbf{s}^*}} \exp\left(-\frac{1}{2}\frac{\left(\bar{s}^l_k - \mu^l_{\mathbf{s}^*}\right)^2}{(\sigma^l_{\mathbf{s}^*})^2}\right) \\ &= \mathcal{L}(\mathbf{p} \mid S_k). \end{aligned} \tag{3.4}$$

In the following experiments we establish the likelihood function when new observations arrive. The application of the ML method, equation (2.4), yields then the final position estimate.

3.3. Experiments

As follows, we describe tools and methods used in the experiments and the evaluation of the WPS performance. We, first, introduce an experiment to shed more light on the RSSI distribution. The other experiments are conducted to assess the fit of the Gaussian process model, which is used to interpolate the RSSI radio map. A comparative study of the fit of the different Gaussian process models narrows down the choice to a few models. These are then used to compute ML estimators, and based it on model fit and the yielded localisation performance, we choose the best fitting model.

WLAN packet capturing The hardware we use to listen to WLAN data is a laptop running Unix-like operating system (GNU)/Linux[2] with a wireless network interface card using the Atheros AR9271 chip set. This WLAN adapter is connected via universal serial bus (USB) to the laptop and equipped with a dipole antenna providing 4 dBi of additional gain.

The remainder of the WLAN data processing chain is in software. To identify the access points from which the RSSIs are emitted the MAC address that identifies uniquely the access point are parsed from he received packets and associated to the RSSIs. We only record *broadcast packets*, since they contain the RSSI of a packet and the MAC address of its source. In addition, broadcast packets are emitted periodically to announce the presence of the access point. Hence, WLAN packets can be captured passively, avoiding privacy issues as these packets do not contain any data of any WLAN user. We use the pcap library[3] to capture packets and to filter for only broadcast packets. To capture the packet passively, the *monitor mode* of the wireless network interface is activated. Also the channel, the network interface listens on, must be changed. Broadcast packets are usually sent every 100 ms; we switch the channel of the wireless network interface card every 200 ms. The channel switching of the WLAN adapter is done with help of the netlink library[4].

2. GNU/Linux, free Unix-like operating system: https://www.gnu.org/gnu/linux-and-gnu.en.html

3. libpcap, portable C/C++ library for network traffic capture: http://www.tcpdump.org/

4. Netlink Protocol Library Suite (libnl), collection of libraries providing application programming interfaces (APIS) to netlink protocol based Linux kernel interfaces: http://www.infradead.org/~tgr/libnl/

After a packet is received the sought data must be extracted. The RSSI can be found in the radiotap header[5], parser are readily available on the web. The MAC address is contained in the broadcast packet, for which an existing parser was adapted. The software so far described is written in C/C++.

Recall, WLAN fingerprinting consists of the survey phase and the localisation phase. For both phases the RSSIs and the MAC address of the access point must be recorded; for the survey phase, in addition, the position of RSSI recording the must be known.

To associate the position with the received WLAN data we adapted the program JMapViewer[6] for our needs. JMapViewer is a java software displaying OSM map data. We use it to integrate the different software blocks and control the overall system. On click on the map we retrieve the position in geodetic coordinates, we resume the thread capturing the WLAN packets and changing the channels. A second click (executed after a few seconds) forwards the position via the java native interface (JNI) to the C/C++ part described before. Then a few statistics are computed on the received RSSIs and then stored all together in a SQLite[7] database. After the storage process completes the thread pauses until the occurrence of a new click on the map, again instructing to record a fingerprint.

3.3.1. Static RSSI measurement from a single access point

The Gaussian process model requires to specify a likelihood function; but since the discussion about the RSSI distribution is not conclusive, an experiment to analyses of the distributions of raw RSSIs and averaged RSSIs is inserted, and described as follows. We refer to this experiment as Single-AP.

The positions of the emitting and the receiving antennae were fix during the experiment. Their distance was about 5 m, but in separate rooms which are connected by door.

Three series of RSSIs were measured in an indoor setting from a single access point, while the receiving antenna and the access point remained at fix positions.

5. radiotap header, packet header provides information additionally to the wireless LAN standards (IEEE 802.11) standard that is added to the packet at the time of capture by the driver controlling the capture device: http://www.radiotap.org/

6. JMapViewer, graphical user interface written in Java displaying open street map (OSM) data as a map: http://wiki.openstreetmap.org/wiki/JMapViewer

7. SQLite, software library implementing structured query language (SQL) database engine: http://www.sqlite.org/

Each of the three measurement series contains about 900 (raw) RSSI values. For the first series the RSSIs were stored unprocessed. For the second measurement series we computed the arithmetic mean and median of the RSSI for every second and stored them instead of the raw values. In the third take, the RSSIs are averaged over a time interval of five seconds.

3.3.2. Gaussian process regression modelling

The experiments have been conducted in our test bed at the faculty of engineering of the Universidad Autónoma de Querétaro (UAQ). The test area comprises two small buildings: an one storey building and a two storey building (both about 8×45 m); and the surrounding area covering about 80×75 m.

Both buildings (see grey areas in the centre of figure 3.1) are sectioned by brick walls with roughcast. Smaller sections are divided by soft partitions. The buildings count with a roofed outdoor passageway from which they can be entered. The one storey building (left building) has a central corridor and consist of small office spaces. The different sections of the two story building (right building) are accessed from the outdoor passageway. This building has a semi-open stairway giving access to the second floor. Size and properties of the buildings and the climatic conditions, that most of the time lead to opened (single-pane) windows and doors, provide rather low signal attenuation.

Radio map construction

Seamless localisation in this work relies on the availability of WLANs in indoor environments and in their surroundings. The characteristics of radio waves – including their signal strength – is considerably different in interiors than outdoors. Indoors the signal strength varies unpredictable, primarily, because of the different mediums they are passing, because of multiple reflections and possibly near field effects. To investigate whether the Gaussian process models, used here to interpolated radio maps, are also different indoors and outdoors, we constructed radio maps for indoor and outdoor environments separately.

We recorded the measurements in three parts. An indoor radio map, containing the fingerprints of the two buildings, and two outdoor radio maps, which contain the area between and behind the buildings. Figure 3.1 shows the three areas for which the radio maps were constructed: the fingerprints that constitute the

Figure 3.1.: Test area and fingerprint positions (·)[8]of one access point. Reference points within the yellow shaded area belong to the Indoor dataset; reference points lying in the blue shaded area form the Outdoor-2 dataset; and all other fingerprints belong to Outdoor-1.

indoor radio map are inside the yellow shaded area, one outdoor radio map covers the blue shaded area, denoted Outdoor-1, and the second outdoor radio maps contains the remaining fingerprints. We denote the second outdoor radio map by Outdoor-2.

A measurement of a fingerprint took about 3 s to 5 s during which the recording device stood still. The body of the experimenter, holding the recording device, attenuates WLAN signal and introduces a dependency to the RSSIs on the orientation of the experimenter. To account for that, we recorded four fingerprints at each reference position of the Indoor and Outdoor-1 radio map. One for every cardinal orientation, where the position of the fingerprint were virtually the same.

8. There is a mismatch between the overlaid reference positions and the background map image, rendered from OSM data. This mismatch constitutes in a slight correction of the OSM data after the WPS experiments were concluded (with the objective to better match GNSS and WLAN data). Regardless the mismatch, we use the updated map image here for reasons of consistency.

An example is shown in figure 3.1. The reference positions belong to an access point that covers the complete test bed, that is, its signals are strong enough to be received on almost every reference position. Dependent on the distant of the access point to the test bed, the signals of an access point are stronger or weaker and therefore were received more or less often, respectively. Signals from distant access points produce rather sparse radio maps and signals from nearby access points produce dense radio maps. The approximate access point position that corresponds to the data in figure 3.1 can be deduced from the interpolated RSSIs in figure 3.5a; it coincides with the peak of the graph in the centre of the test bed.

In addition to the three mentioned radio maps, we created a fourth one by joining the three radio maps into one: the Test-bed dataset. We use this radio map in addition and for the experiments evaluating the positioning performance.

Dense and sparse radio maps To analyse the influence of the fingerprint density on the model fit and on the positioning performance, experiments with a dense and a sparse radio map are required. The dense radio maps are the radio maps as described above, containing the complete set of data as obtained in the experiments. The sparse radio map has less fingerprints. However, the spatial distribution might be locally dense, because it is constructed by picking at random from the dense radio maps such that the amount of fingerprints equals the fourth part of that of the dense radio map. We expect with this procedure to obtain results that are applicable in general practice. Note, that the total amount of fingerprints for a given access point depends on its reception conditions at a certain position in space.

Training and assessment of the Gaussian process model

To interpolate the RSSIs over space we need a model that emulates the spatial structure of RSSIs. Gaussian process models are known to be flexible enough to model a wide range of structures. Gaussian process modelling is essentially the search for the structure in the data.

The principle structure of a Gaussian process is provided by its prior distribution, determined by a mean and covariance function. Each of these functions has a certain characteristic, which in combination account for certain structural properties or trends in the data. Note, that these two functions only assume properties on the underlying process. To find the best suited Gaussian process,

which fits the data being modelled, can be a difficult. It involves the choice of mean and covariance function, the search for the hyperparameters, which adapt the mean and covariance functions to the training data, and the comparison of the resulting models. We test a variety of mean and covariance functions, use them to interpolate the radio maps and evaluate the fit of the models based on reasonable measures and the model residuals.

Choice of mean and covariance function An overview of possible mean and covariance functions can be found in the literature (Rasmussen et al. 2006, e.g.). Global characteristics of these models are determined by their mean function. The covariance function models the local correlation between neighbouring RSSI. It accounts for local structures, but it can also model global structures.

In a homogeneous medium the power of radio frequency waves decays exponentially. We expect thus a two-dimensional exponential decay as the general form of the latent function, the global structure of our data. However, depending on position of the access point within the area of interest, which may only cover a part of the area that an access point covers, and superimposed effects caused by the wave's propagation environment, the underlying function could be a one-sided exponential, or have any other shape. In general we expect RSSIs to obey this two-dimensional exponential decay, or a clipping of it. More complex global global structures – for example periodicity – are not expected.

This globel structure is likely superimposed by local variations of diverse origin. In indoor environments, we expect additionally attenuation when the wave passes obstacles. We assume furthermore random fluctuation attributed to passing people or objects, due to adaptions of the transmit power of the access point (to reduce interference in the communication channels) and because of multipath propagation effects. The signals may as well be affected by uncorrelated noise.

The Zero mean function is the mostly used mean function[9]. In the experiments we use 1) the zero mean function, Zero, 2) the constant mean function, Const and 3) the linear mean function, Lin.

A zero mean function is zero everywhere and Gaussian processes with a zero mean function, which are not conditioned on data, are simply zero. A constant mean function assumes that the underlying process is everywhere constant

9. The zero mean function is the standard one, because it is simple and the covariance function can model its uncertainty by adding an additional term.

(determined by a single hyperparameter) and a linear mean function assumes a linear structure (determined by two hyperparameters).

In this work we use the following basis kernels to construct covariance functions

- squared exponential kernel, SE
- Matérn kernel with parameter $\nu = \frac{1}{2}, \frac{3}{2}, \frac{5}{2}$, $\text{MAT}_{\nu=\frac{1}{2}[,\frac{3}{2},\frac{5}{2}]}$
- rational quadratic kernel, RQ
- independent noise, kernel IN
- additive kernel, ADD

Additive kernels (ADD $\cdot$) have the particular property that their structure is governed by interactions across different dimensions. Additive kernels are composed by summing up all possible products of one-dimensional basis kernels (Duvenaud et al. 2011). Gaussian processes with such covariance function are called additive Gaussian processes. Additive Gaussian processes pose very general models, because in addition to the hyperparameters of each basis kernel they possess also parameters that determine the amount of interactions from each order. As all hyperparameters are adapted automatically, these processes have increased flexibility and are able to model a wide range of structures.

We evaluate these kernels directly or we combine them to form new covariance functions. Operations used to combine these basis kernels are the sum (SUM $\cdot$ & $\cdot$) the product (PROD $\cdot$ & $\cdot$).

Optimising the hyperparameters After the mean and covariance function are chosen, the optimal hyperparameters need to be found, recall 2.3.2. To do so, we use the gpml package[10] and GNU Octave[11]. This toolbox determinces the hyperparameters by minimising the negative logarithmic likelihood function of the training data with a gradient descent method, the limited-memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) algorithm.

10. gpml v3.5, http://www.gaussianprocess.org/gpml/code/matlab/doc/
11. GNU Octave, high-level interpreted language, primarily intended for numerical computations: http://www.gnu.org/software/octave/

Comparison of Gaussian process models The evaluation of the performance of the Gaussian process models is based, on the one hand, on the test error and, on the other hand, on the Bayesian information criterion (BIC) – both are obtained as an average from a 10-fold cross-validation. Therefore, we split the training data into ten folds. We use nine of them to fit the model and use the inputs of the remaining 10th-fold as test data for the prediction. The RMS of the difference between the target values of the 10th-fold and the predicted values yields the average test error. The second measure to assess the model is the BIC. (The BIC is used instead of the logarithmic marginal likelihood, because the latter favours models with more hyperparameters if already fitted Gaussian process models are compared under the same settings (Duvenaud 2014)). The BIC is based on the logarithmic marginal likelihood (see eq. (2.35)) of the data and contains an additive term that penalises larger numbers of model parameters: $\mathrm{BIC}(\mathsf{M}) = -2\log(\mathcal{D} \mid \mathsf{M}) + \frac{1}{2}|\mathsf{M}| \log N_*$; where M stands for the model, $\mathcal{D}$ are the data, $|\mathsf{M}|$ is the number of free model parameters and N_* the number of test points. Albeit that RSSIs do not meet all of the assumptions for applying the BIC, it yields a reliable model measure (Lloyd et al. 2014). Since a model is trained for each access point separately, the model selection criteria are averages over the access points.

The test error can be counted to the category of frequentist tools as it relies only on the data. The BIC incorporates a priori information, thus being part of Bayesian statistics. This choice of model selection criteria is expected to lead to unambiguous and reasonable conclusion.

Implementation details For each of the four radio maps, we trained a Gaussian process model for each access point. Since the optimisation procedure failed in cases when only very few fingerprints per access point were available, we excluded access points with less than 30 fingerprints in a dataset. For the Indoor dataset we trained models for 49 access points, we trained 91 models for the Outdoor-1 and 35 models for the Outdoor-2 radio maps. The Test-bed dataset, combining all three radio maps, consists of 100 access points. These numbers seem high in the first place, but all institutional access points are virtual access points; each of them emulates four different access points. Although the location of virtual access points is identical, we treated them as distinct access points as their MAC address is different.

Figure 3.2.: Test area with fingerprint positions (·) of one access point and trajectories (Trajectory-1 ----; Trajectory-2 -·-·-). The starting point of each trajectory is marked with a circle and the end point with an arrow. © OpenStreetMap contributors © IEEE 2015

Optimising the hyperparameters requires an initial guess. Before running the cross-validation experiments, we ran multiple tests on all datasets to find good initial hyperparameters. The number of maximum function evaluations of the l-bfgs algorithm was set to 70.

The grid on which we interpolated the rssis is 2 m × 2 m.

Maximum likelihood position estimation

To validate the Gaussian process model, we evaluated the localisation performance based on a ml estimator using these models. We recorded wlan data along two paths. These paths are presented in figure 3.2 and are denoted by Trajectory-1 and Trajectory-2. Both of them run through indoor and outdoor environments, but most of the time they are on the semi-open passageway.

For localisation only the rssis and mac addresses are needed. However, to analyse the performance additionally the reference coordinates on these

trajectories were recorded. To obtain this ground truth we used the software described before to record the radio maps. A trajectory dataset contains for each of its position the coordinates of the position that was clicked on the map, a set of RSSIs for all receivable access points and the MAC address associated with these access points. The capturing time at each position was about 2 s to 3 s. The mean and the variance of the RSSIs were finally stored in the database.

To compute a ML estimate, we precomputed the RSSI predictions for both Gaussian process model. At the arrival of new measurement the likelihood function, equation (3.4) was established and its maximum was determined.

As performance criteria we use is the RMS of the error. This accuracy measure computes first the error between each position of the ground truth trajectory, $\mathbf{p}_r$, and the estimated trajectory $\hat{\mathbf{p}}_r$. The root mean square error (RMSE) results as the square root of the squared and averaged errors. For a trajectory of n_r points we can write

$$e^{\text{RMSE}} = \sqrt{\frac{1}{n_r} \sum_{r=1}^{n_r} (\mathbf{p}_r - \hat{\mathbf{p}}_r)^2}. \tag{3.5}$$

3.4. Results

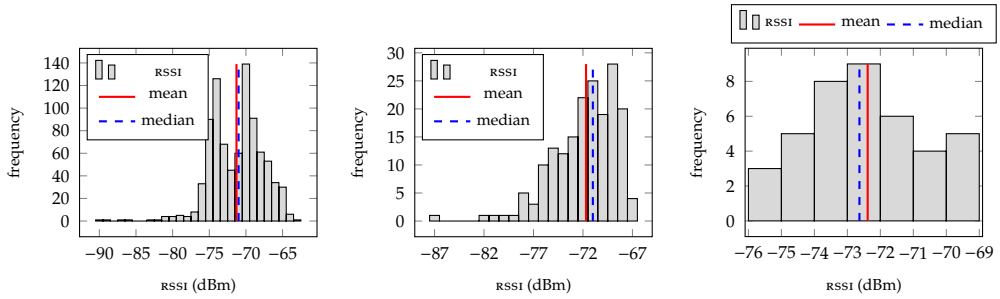
Before presenting the results for the Gaussian process model search, we show results from static indoor RSSI measurements to justify our choice of the WLAN RSSI distribution. Kaemarungsi et al. (2012) analysed the statistical properties of RSSIs and explored their impact on the localisation performance of WLAN location fingerprinting systems. Here, in addition, statistics of raw and averaged RSSIs are examined.

The second part of this section contains the results of the structure search in WLAN RSSI. A number of Gaussian process regression models are thoroughly assessed in order to spatially interpolate RSSIs. This assessment relies on two model criteria and the analysis of the regression residuals. The impact of some Gaussian process models on the localisation performance is evaluated in the last part of this section. It seeks to confirm if the models yielded by the model selection criteria are also the best suited models for localisation; or: if the evidence in form of model selection criteria transfers to practice.

3.4.1. Single position single access point RSSI measurements

In section 3.1.1 we argued to model WLAN RSSIs as normally distributed data as a compromise between practicality and complexity. Many WLAN location fingerprinting systems use the mean of RSSIs. This choice is supported by the central limit theorem; but only if the observations meet certain conditions, most importantly that they are independent and have finite variance. To corroborate the use of averaged RSSIs and to support the argumentation for normal distribution as a trade-off, the PDF of averaged RSSIs are compared to that of raw RSSIs and analysed with respect to normality. Recall the Single-AP experiment from section 3.3.1.

The RSSI histograms of raw RSSIs and averaged RSSIs are shown in figure 3.3. The left panel depicts the histogram of raw RSSI observations, the centre panel contains the histogram of RSSIs averaged over 1 s and the rightmost panel shows the histogram of arithmetic means of RSSI averaged over 5 s. All three histograms



(a) Histogram of unprocessed RSSI.

(b) Histogram of RSSI averaged over one second.

(c) Histogram of RSSI averaged over five seconds.

Figure 3.3.: RSSI histograms at a fix location. RSSIs are either single samples, samples averaged over one second or averaged over five seconds. Mean and median are marked by vertical lines.

are considerably skewed to the left and present multiple modes – that confirms that RSSI PDFs are not normal distributions.

The tendency of RSSI PDFs to have heavy tails to the left is simply caused by the nature of signal attenuation: Consider a fix emitter sending at constant power and fix receiver. On the one hand, signals may take the direct path, only affected by free space attenuation. These signal are the possibly strongest signals the

receiver observes, showing on the right from the mean in the histogram. These signals are bounded by the emitted power on the upper side, which prevents heavy right tails in the histogram, it even promotes light tails on the right hand side. On the other hand, signals sent with the same power may not take the direct path, but a quite complex one; for example passing walls, get reflected and scattered multiple times before they are received. These signals, albeit sent with the same power, show on the left side from the mean in the histogram. In indoor scenarios with complex paths between emitter and receiver, less signals are observed via the direct path, but more signals take in average a complex path and become more and more attenuated. These signals have no lower bound as their path can be arbitrarily complex – causing left-skewed PDFs. However, the wireless network adapter imposes a lower bound.

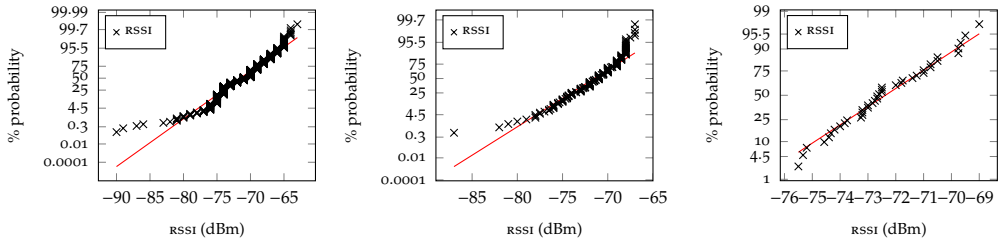
For RSSIs with a low average value the situation is slightly different. Signals that have low in power in average are not received by wireless network interfaces if they are below a certain threshold. This lower bound prevents the reception of RSSI far from the mean value. In this scenario the RSSIs have a lower and an upper bound that causes the histograms to be more symmetric. This was already pointed out in Kaemarungsi et al. 2012.

The normal probability plot presents the data against a theoretic normal distribution, see figure 3.4. We present again the raw data and the averaged RSSIs. The normal probability plot in the left panel (3.4a) and in the centre panel (3.4b show a similar non-linear trend, reflecting the heavy left tails and as well the light right tails of the distributions. However, around the mean all three plots are reasonable linear. (Especially the leftmost panel reveals the discrete nature of RSSIs.)

When considering the transition from raw RSSIs to RSSIs averaged over 1 s in the histograms and in the norm probability plots, a decrease of the left tail and an increase of the right tail is observable. The data is less spread out, the lower and upper bounds moved closer to the average; while the central tendency measures approximately remain.

Moving to the data averaged over 5 s this trend continues. The data concentrates more around the central tendency measures and its PDF becomes almost symmetric. In panel 3.4c most of the data points lie on the line indicating normality – thus, supporting the notion of a Gaussian model, albeit the PDF of RSSIs averaged over only a few seconds is, in a strict sense, not normal.

Table 3.1 shows the central tendency measure of the three data sets. The mean



(a) Norm probability plot of unprocessed RSSI.

(b) Norm probability plot of RSSI averaged over one second.

(c) Norm probability plot of RSSI averaged over five seconds.

Figure 3.4.: RSSI normal probability plot at a fix location. RSSIs are either single samples, samples averaged over one second or averaged over five seconds. The linear graph in the normal probability figures connects the first and the third quartile of the data and is then extrapolated towards the ends.

and the median are compared, to additionally examine if one or the other is eventually more consistent and therefore more robust, especially in the light of non-Gaussian RSSIs. The difference between mean and median is small, for

Table 3.1.: Mean and median of static RSSI measurements for different post-processing. (These values correspond to the mean and median shown in the panels 3.3a–3.3c.)

	raw RSSIs	1 s avg. RSSIs	5 s avg. RSSI
mean	-71.29	-71.69	-72.38
median	-71.00	-71.00	-72.63

all three data sets it is less than 1 dBm. We prefer therefore the mean over the median, because of its more convenient properties, primarily linearity.

To assess the dispersion of data, more robust estimators than the standard deviation exist, too. One example is the median absolute deviation. The time intervals of 1 sand5 s for collecting data and calculate the statistics are unfortunately too short. In many intervals only a single value was recorded, obviously not sufficient data to compute sample statistics. Because of too many intervals with invalid data the estimation of the average dispersion of the (averaged) RSSIs was not feasible.

Averaged RSSIs obey the central limit theorem, hence, for large averaging intervals they converge to a normal distribution. RSSIs averaged over one and three second are may be normally distributed. They are with higher probability symmetric and with lower probability heavy tailed, though. This is noteworthy, because according to Kaemarungsi et al. (2012) the logarithmic RSSIs can be approximated with a normal distribution.

3.4.2. Gaussian process model selection and validation

To determine Gaussian process model that is best suited to model RSSIs spatially, we adapted several models whose prior distributions are yielded by combining the mean and covariance functions listed in section 3.3.2. The results, in terms of the model selection criteria, are presented separately for the three radio maps: the indoor radio map, the two outdoor radio maps and the dataset that combines these radio maps. For models for those the model selection criteria do not allow a clear assessment, we examine additionally the predicted mean and covariance functions. Moreover, the model residuals for two selected access points are evaluated and presented in section 3.4.2.

We actually evaluated more covariance functions than we present here. Because some combinations of basis kernels resulted consistently in poorer models than a basis kernel alone (larger residuals and BIC), we withhold them from the evaluation. The additive combination of basis kernels with the independent noise kernel are such a cases.

For some covariance functions the optimisation of the hyperparameters failed because of singular covariance matrices. The use of the additive covariance function of orders higher than one resulted in that problem. Results for these models can hence not be presented.

Test error and Bayesian information criterion

The chosen model selection criteria for the predicted mean function from the Gaussian process regression are presented in the following tables. Each of the tables contains the results for the different radio maps and their combination within the Test-bed dataset. They show the test error and BIC for different prior distributions – different combinations of mean and covariance functions. In theory, the better fitting models yield lower test error and lower BIC than models

that fit the data poorly.

Table 3.2 compares the model selection criteria for the Indoor training data.

Table 3.2.: Test error and BIC of a Gaussian process regression for different combinations of mean and covariance functions. The Gaussian process models are trained with Indoor data. Model measures are averages over several access points and 10-fold cross-validation. The lowest values are highlighted.

	test error / dBm			BIC		
	Zero	Const	Lin	Zero	Const	Lin
Se	9.856	9.714	9.864	703.093	682.127	682.051
$\text{Mat}_{\nu=1/2}$	9.727	9.686	9.890	691.012	684.433	684.421
$\text{Mat}_{\nu=3/2}$	9.829	9.715	9.916	693.923	682.010	681.903
$\text{Mat}_{\nu=5/2}$	9.847	9.741	9.915	697.181	681.882	**681.730**
Rq	9.796	9.727	9.900	691.367	684.291	684.075
Sum Se & Se	9.860	9.709	9.922	707.760	686.931	687.104
Prod Se & Se	9.900	9.723	9.854	707.089	686.870	686.856
$\text{Add Se}_{o=1}$	9.765	**9.684**	9.762	704.530	693.851	694.259

If we compare first the test error with respect to the mean functions, one can observe, despite the small differences, that the constant mean function yields consistently the best results. If ranking the test error, the zero mean function yields except for the Sum Se & Se case the second lowest values. The linear mean function comes third. Repeating this comparison for the BIC, we find the linear mean function yielding the lowest BIC for all of covariance functions, except Sum Se & Se and $\text{Add Se}_{o=1}$. According to the BIC but in contrast to the test error, models using the zero mean function fit the data consistently worst. With respect to the mean function, the data from the Indoor radio map is ambiguous: The test error suggests the use of the constant mean function, whereas the BIC points to the linear mean function, albeit it resulted in the highest test errors. Noteworthy are the comparatively large differences of the BICs for the zero mean function with Se kernel the compared to the other two mean functions.

Ranking the test errors and comparing them points to the additive squared exponential and to the $\text{Matérn}_{\nu=1/2}$ covariance kernel of order one, but also the Matérn covariance function with $\nu = 1/2$ might be a good candidate. If we repeat this comparison procedure for the BIC, we may choose a kernel from the Matérn

function class, the $\text{Mat}_{\nu=5/2}$ kernel produced the lowest BICs. The hight BIC of the Add Se kernel is opposed to its low test error.

The BIC of the last three kernels, which are rather complex kernels with more hyperparameters, is distinctly higher than that of simpler kernel. This is in conformity with the definition of the BIC. A conspicuous detail can be observed from BICs for combination of the Zero mean function with the Se kernel: it yielded a BIC as high as that of the complex kernels, though it is a simple model with relatively few parameters.

An assessment of both model criteria for this Indoor data set indicates the use of a prior distribution with the constant mean function and the $\text{Matérn}_{\nu=1/2}$ or $\text{Matérn}_{\nu=5/2}$ kernel as covariance function. Nonetheless, as the model selection criteria of the $\text{Matérn}_{\nu=3/2}$ kernel are very close to the former ones it might be also a good option. In addition, the $\text{Matérn}_{\nu=1/2}$ function achieved low model measures only in combination with the zero mean function (which evoked some inconclusiveness); thus, the $\text{Matérn}_{\nu=1/2}$ function should be used with some care.

In Table 3.3 the test error and the BIC, for combinations of mean and covariance function, from the outdoor data sets are presented.

According to the test error for the Outdoor-1 datasets, the prior distribution with constant mean function yields for the majority of cases the lowest result. However, the difference to the test errors of the zero mean function are very small. Models from the Outdoor-2 dataset relying the zero mean function outperforms the models using the constant mean function. The linear mean function produces again the poorest models compared to the zero and constant mean function for both outdoor datasets. But only based on the test error. Given now consideration to the BIC of the Outdoor-1 data, the linear mean function constructed the best Gaussian process models and for the Outdoor-2 dataset the constant mean function yielded consistently the lowest values. As for the Indoor data, the BICs of the Zero mean function are the largest, contradicting the conclusions based on test error again. In particular the combination of Zero with Se yields a striking high BIC compared to the other mean functions. Contemplating the both model selection criteria the constant mean function seems to be the best compromise, as its test error and BICs are not conflicting.

A results obtained by different covariance functions are this time more coherent. Models using the additive squared exponential kernel produced distinctly the lowest test errors – for all mean functions and for both data sets. Considering the BIC, the Matérn covariance functions stand out. The $\text{Matérn}_{\nu=3/2}$ kernel shows

Table 3.3.: Test error and BIC for different Gaussian process models trained with the outdoor data. The values are averages over several access points and 10-fold cross-validation. The lowest test error and BIC are highlighted.

		test error / dBm			BIC		
		ZERO	CONST	LIN	ZERO	CONST	LIN
Outdoor-1	SE	8.982	9.025	9.085	1015.224	994.559	994.662
	$\text{MAT}_{\nu=1/2}$	9.075	9.041	9.090	999.759	993.034	992.891
	$\text{MAT}_{\nu=3/2}$	9.087	9.054	9.101	1004.740	991.610	**991.580**
	$\text{MAT}_{\nu=5/2}$	9.065	9.051	9.097	1009.041	992.243	992.330
	RQ	9.064	9.055	9.107	1002.027	995.405	995.051
	SUM SE & SE	8.984	9.034	9.068	1020.611	999.941	1000.402
	PROD SE & SE	9.008	9.026	9.082	1020.691	1000.822	1000.060
	$\text{ADD SE}_{o=1}$	8.971	**8.964**	9.017	1032.788	1021.243	1020.440
Outdoor-2	SE	7.315	7.348	7.401	372.575	356.170	357.956
	$\text{MAT}_{\nu=1/2}$	7.326	7.289	7.333	366.333	357.939	360.296
	$\text{MAT}_{\nu=3/2}$	7.338	7.322	7.382	368.586	356.720	358.833
	$\text{MAT}_{\nu=5/2}$	7.345	7.335	7.393	370.125	**356.221**	358.537
	RQ	7.354	7.361	7.418	367.016	357.939	359.611
	SUM SE & SE	7.317	7.337	7.401	376.122	359.774	361.995
	PROD SE & SE	7.296	7.339	7.386	376.388	359.774	362.188
	$\text{ADD SE}_{o=1}$	7.225	**7.207**	7.283	375.852	366.793	367.726

for both data set low BIC, however, $\text{Matérn}_{\nu=3/2}$ kernel is a candidate, too. These results shows again the discrepancy of the $\text{ADD SE}_{o=1}$ kernel: one the on hand the lowest test errors and on the other hand the highest BICs. The model combining ZERO function and SE kernel produced again a BIC as large as that of more complex models.

Because of the mentioned contradictions of test error and BIC a good choice for the outdoor radio maps is a prior distribution with a constant mean function and a Matérn covariance function with $\nu = 3/2$. We believe this combination balances well between the test error and the BIC.

The next table (3.4) shows the results yielded from the regression on the combined radio map, the data set that includes the previous three radio maps. These results are still more coherent and confirm the conclusions deduced from

the smaller data sets.

Table 3.4.: Test error and BIC for different Gaussian process models trained with the Test-bed data. Model measures are averages over several access points and 10-fold cross-validation. The lowest values of each category are typeset in bold. © IEEE 2015

	test error / dBm			BIC		
	ZERO	CONST	LIN	ZERO	CONST	LIN
SE	9.277	9.286	9.403	1498.148	1458.365	1460.108
$\text{MAT}_{\nu=1/2}$	9.351	9.318	9.422	1459.503	**1451.745**	1452.113
$\text{MAT}_{\nu=3/2}$	9.346	9.308	9.435	1472.590	1452.499	1453.363
$\text{MAT}_{\nu=5/2}$	9.347	9.305	9.433	1481.816	1454.384	1455.309
RQ	9.334	9.325	9.428	1462.458	1455.182	1456.116
SUM SE & SE	9.278	9.290	9.388	1504.146	1464.460	1466.045
PROD SE & SE	9.274	9.287	9.400	1504.203	1464.387	1466.445
$\text{ADD SE}_{o=1}$	9.059	**9.051**	9.141	1544.440	1531.548	1531.577

Models that rely upon the constant mean function yield the lowest test errors in combination with simple covariance functions, whereas models that rely on the zero mean function yield the lowest test errors in combination with rather complex mean function. Once again, the linear mean function produces consistently the highest test errors. The BIC favours clearly models with the constant mean function, yielding the lowest values. The second lowest BICS yield the linear mean function. The highest BICS result from zero mean function, particularly in combination with the SE and the complex kernels. The difference between the BIC of the zero mean function to the BICS of the other two mean functions is again noticeable large. It seems the constant mean function emerges as the best candidate from the data.

Consider the model measure for the different kernels: Choosing the covariance function solely on the test error would result again in additive SE kernel. In a ranking of test errors, followed by the PROD SE & SE and the SUM SE & SE kernel. As before, this is opposed to the results reported by the BIC. Regarding the BIC, models using the Matérn kernel with $\nu = 1/2$ and $\nu = 3/2$ fit the data best and models using the PROD SE & SE and the SUM SE & SE kernels show large BICS, hence, model the data inadequately. This inconclusiveness between test error and

BIC is illustrated best by the Add $\text{Se}_{\ell=1}$ kernel. The high BIC, (despite the relatively low test error) of the prior with zero mean function and squared exponential covariance function is again present and not reasonable regarding the amount of hyperparameters.

Based on this analysis of the four data sets, we narrow down the choices of prior distributions to a few candidates. These candidates will be further scrutinised, whereas other prior distributions will be eliminated. The insights obtained from the model selection criteria call for some conclusions that we summarise below:

- The differences of model measures between similar models are small, their informative value is rather limited. Furthermore, the examination of the mean and covariance function was done independently, without giving much consideration to specific combination. Nevertheless the model measures for a certain mean function in combination with certain kernel function may differ more than the difference to the other mean functions.

- The test error and the BIC provided partially contradictory conclusions:
  - The linear mean function yielded for all data sets high test errors, but low BICs. We exclude the Lin mean function in favour of the constant mean function from further evaluation, because the results of the constant mean function are consistent and their BICs are very close while the Lin mean function is more complex (possessing one hyperparameter more).
  - The simple kernels yielded low test errors, but high BICs. We exclude and complex kernels – except the Add Se kernel, as it yielded noticeable low test errors – from further analysis, because they more likely to overfit the data and its adaption to the data is more computationally demanding.

- The prior distribution based on the Zero and Se functions yielded low test errors; but also high BICs with magnitudes in the range of the complex models, while its number of hyperparameters is in the range of that of the simple models. We exclude the squared exponential kernel as an immediate choice. However, especially because it is the most common one and was successfully used in combination with the zero mean function as the 'default' Gaussian process prior, we select it for further evaluation.

- Models using the constant mean function outperform, in average over all data sets, the models relying on the zero or linear mean functions. It is therefore the top candidate mean function.

- The lowest BICs were achieved when the constant mean function was combined with the Matérn kernels with roughness parameter: $\nu = 5/2$ for the Indoor data, $\nu = 3/2$ and $\nu = 5/2$ for the two outdoor datasets and $\nu = 1/2$ for the joined radio map. For the Indoor and the Outdoor-2 data set these kernel yielded also the quite low test errors. For that reason and for being a trade-off between $\text{MAT}_{\nu=1/2}$ and $\text{MAT}_{\nu=5/2}$, we pick the $\text{Matérn}_{\nu=3/2}$ function as the (currently) best candidate.

- Convincing evidence was not found that the Gaussian process model for indoor and outdoor environments should use different prior distributions. The candidate models that followed from the comparison of the model selection criteria are not clearly distinguishable.

  Rather the contrary is the case. For both environments the constant mean function yielded the best results and also the Matérn kernel is suggested by two data sets with consistency: the Indoor and Outdoor-2 data. Only the test error from the Outdoor-2 data set shows unfavourable values for the Matérn kernels – probably because of the distribution of the fingerprints in one dimension, along the paths see figure 3.1.

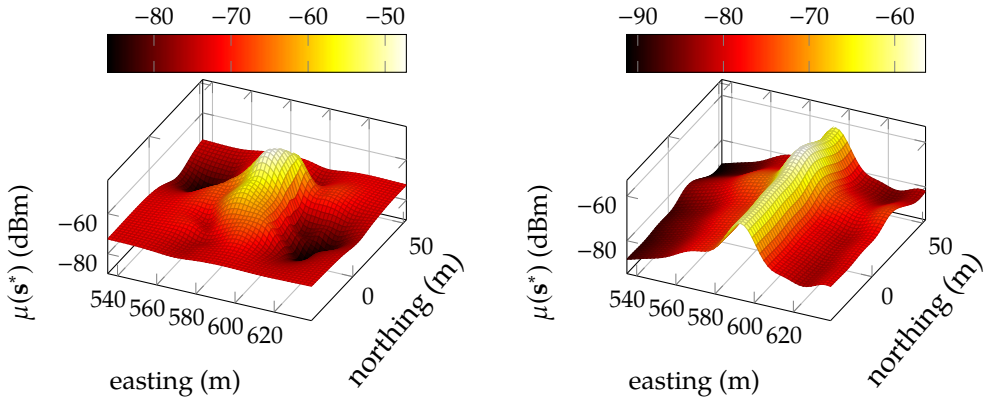
  An analysis of the hyperparameters for the same mean and covariance function did not disclose more insights. We compared the hyperparameters yielded from indoor data with the hyperparameters yielded from outdoor data for three prior distributions, certain patterns were not found. The reader may find that data in appendix C.

Up to this point, we consider a Gaussian process model with a prior distribution based on the $\text{Matérn}_{\nu=3/2}$ covariance function as the best suited to model RSSIs. As follows, we use this model as a reference model to examine the other candidate: the additive square exponential kernel. Both are combined with the constant mean function.

Because no differences between the indoor and outdoor models was found, the Test-bed radio map is used from here on.

Two spatial RSSI distributions, predicted from models relying on the constant mean function and using the $\text{Matérn}_{\nu=3/2}$ kernel and the additive SE kernel are

shown in the figures 3.5 and 3.6. The function in panel 3.5a presents a global



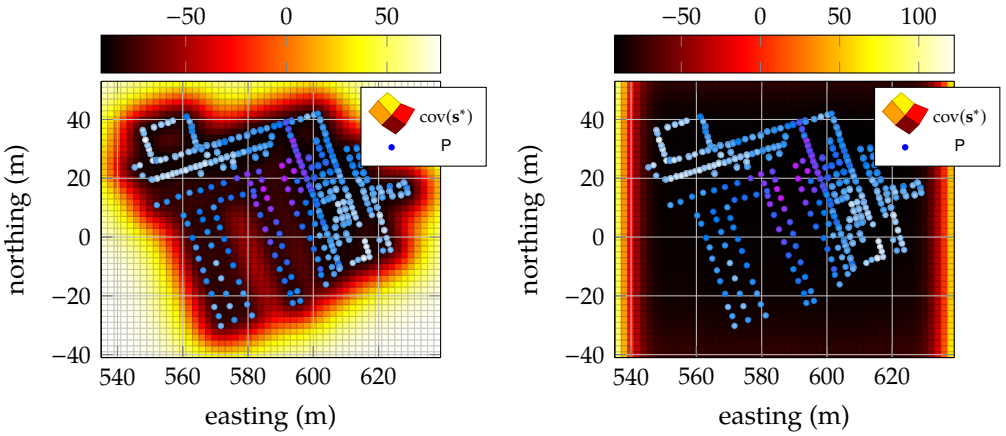
(a) Predicted RSSI mean function based on a prior with Matérn$_{\nu=3/2}$ kernel.

(b) Predicted RSSI mean function based on a prior with additive SE kernel of order one.

Figure 3.5.: Predicted RSSI mean functions of two Gaussian processes with different prior covariance function for one access point. The two Gaussian process models have both a constant mean function combined with either Matérn$_{\nu=3/2}$ or additive SE kernel, respectively. The magnitude of the mean and covariance function are indicated by the colour bars.

maximum, indicated the position of the access point, in the centre of the test area. To the borders of the test area this function decays, this decay is governed by local minima and maxima. Panel 3.6a present the corresponding covariance function and the training points that where used to regress the model. It shows low uncertainty in regions where a lot of information (in form of training data) is available. That is reasonable and expected. In areas where few or no training data is available the variance is high, representing the uncertainty about the predictions.

In contrast, the predictions from the ADD SE prior covariance function in the panels 3.5b and 3.6b are not comprehensible. Panel 3.5b depicts a mean function that predicts almost constantly high RSSIs along the y-axis, around 590 m, with a global RSSI-peak coinciding with that in panel 3.5a. The covariance function in panel 3.6b is almost constant over a large area and increases towards the edges of the test area, in particular east and west. It does not logically follow the training data. Neither the structure of the RSSI nor the covariance function are reasonable.



(a) Predicted RSSI covariance function based on a prior with Matérn$_{\nu=3/2}$ kernel.

(b) Predicted RSSI covariance function based on a prior with additive SE kernel of order one.

Figure 3.6.: Predicted RSSI covariance functions of two Gaussian processes of one access point and corresponding training data. The two Gaussian process models have a constant mean function combined with either Matérn$_{\nu=3/2}$ or additive SE kernel. The magnitude of the mean and covariance function are indicated by the colour bars. Large training data are coloured purple–dark blue and low RSSI light blue–white.

This prediction is not explainable by physics or the existing environment.

Albeit Gaussian processes with a prior distribution that uses the ADD SE$_{o=1}$ kernel yielded very low test errors, the posterior process shows that they fail to fit the data – and that the small test error is misleading. A large bias or variance of the test error may cause these misleading outcomes. We thus exclude the ADD SE kernel from the list of candidates and establish the Matérn kernel based covariance function as the kernel of choice for RSSI interpolation.

Spatial density of fingerprints A critical and much investigated issue of WLAN location fingerprinting is the required effort to create the radio maps. Although interpolating the radio map reduces this labour, a radio map is still required.

The effect of the amount of fingerprints on the interpolated radio maps was studied by Bekkali et al. (2011). However, the applicability and generality of their

results appears restricted, as they based their study on simulations, idealising the problem: Ray launching is used to generate the RSSIs; the test environment is indoors only, constituting a single, symmetric room; the fingerprints are on a regular grid; realistic disturbances of the signal strength are not modelled; different mean functions than the zero mean function are not regarded.

This part of the analysis of Gaussian process models examines the influence of the fingerprint density on the quality of predictions. It is based on data that was obtained in a realistic indoor/outdoor environment and extends the findings of (Bekkali et al. 2011). We share some conclusions of their study, though, we want to look into one of their results: That a model that uses the SE kernel fits RSSIs better than a model relying on the MAT kernel. The importance of that point rises when considering that the SE covariance function in combination with the ZERO mean function is used as 'default' model in a majority of works.

Before getting to the results though, we have to put our experiment in perspective. First, the fingerprint density in our study and in Bekkali et al. (2011)'s are different; a dense radio map for our scenario would still be sparse in their environment. Second, some parts of our Test-bed radio map contains up to four fingerprints on virtually the same position. Third, the models we use rely on the constant mean function, not on the zero mean function – recall, based on that choice, in particular the difference in BIC can be large.

Table 3.5 presents the BIC for the models using constant mean function and either the squared exponential or the Matérn kernel. We compare the fit of the models when trained with either a dense or a sparse radio map. The dense radio map is the complete Test-bed radio map and the sparse radio map was obtained by sampling the Test-bed radio map, according to section 3.3.2. In addition, we did this for two access points. One that provided many fingerprints to the radio map, AP_D, and an other access point whose signal were not captured that often and therefore did not contribute to many fingerprints, AP_S. The complete set of fingerprints of the access point AP_D can be found in figure 3.6a and the complete set of fingerprints of the access point AP_D is shown in figure 3.7a. For three out of four scenarios, the models employing the Matérn$_{\nu=3/2}$ kernel achieve a better model fit. In the case of the undersampled radio map, from an access point that already has a low amount of fingerprints in this radio map, the squared exponential kernel fits the data marginally better. The BIC-difference is very small and the result may not be significant. Based on our data the statement of (Bekkali et al. 2011) can not be refuted, but neither confirmed with certainty. Only if the

Table 3.5.: Test error and BIC obtained from Gaussian process model with constant mean function and either SE- or MAT-kernel. For each of the two kernels the model was fitted either using the complete training data (complete) or using only a subset of the training data (undersampled) for both access points. The lowest BIC is typeset in bold.

	BIC - AP_D		BIC - AP_S	
	dense	sparse	dense	sparse
SE	3559.841	939.622	1077.977	**288.091**
$MAT_{\nu=3/2}$	**3542.781**	**936.257**	**1073.987**	288.178

fingerprint distribution is very sparse the SE kernel seems to perform better.

One may conclude that for spatially dense fingerprint distributions, the Matérn kernel should be used, and for sparse fingerprint distributions, the squared exponential kernel has potentially advantages. Considering the properties of the two kernel functions and the data that used for this examination, this result is reasonable. Due to its parametrisation the Matérn function is very flexible, based on ν this function can be very rough but also very smooth, it even converges to the SE kernel for $\nu \to \infty$. Generally speaking, the Matérn kernel is rougher than the squared exponential kernel and is therefore able to adapt to large RSSI variations within small distances. As this occurs more likely when the fingerprint distribution is dense, the Matérn function is beneficial in these cases. In scenarios where the fingerprints are sparse, the training RSSI are smooth (either because the latent function is smooth, or because not enough information about the latent function is available), which can also be fitted by processes with a prior SE covariance function.

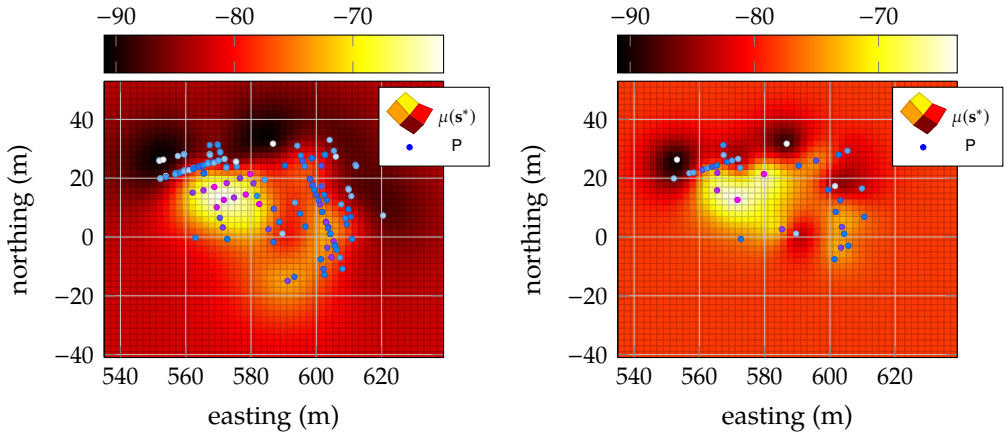
The decision for one or an other covariance function based on the spatial fingerprint density is very difficult, since the environment affects this density. First of all, in a certain area, the fingerprint density for different access points usually varies, because of the reception condition of each access point. And for a single access point the reception condition in turn depends on the position of the user/object, the changes in the environment and the wireless network interface of the user/object. In practice these factors are uncontrollable. Of course, in general the fingerprint density decreases with distance from the access point; the spatial distribution of the corresponding fingerprints is still non-uniform.

Without further provision for the interpolation of RSSIs, one would need to predict the fingerprint density for each access point – may be as function of space – or include it as a-priori information.

For these reasons we reinforce our recommendation to use of the Matérn$_{\nu=3/2}$ covariance function (over the squared exponential covariance function).

The amount of fingerprints used to analyse the model fitting for the different prior distributions is admittedly high. If considered from a practical point of view one would like to avoid such a fingerprint density. To assure practicability of the suggested model, even if very few training data is available, we examine the predicted RSSI mean functions for an access point with a low density of fingerprints (AP$_5$). The corresponding results are shown in figure 3.7.

The left hand side panel 3.7a depicts the predictions obtained from a process trained with the complete radio map and the right hand side panel 3.7b depicts the predictions obtained from a process trained with a undersampled radio map. Although for the process in panel 3.7b only the fourth part was used to adapt it



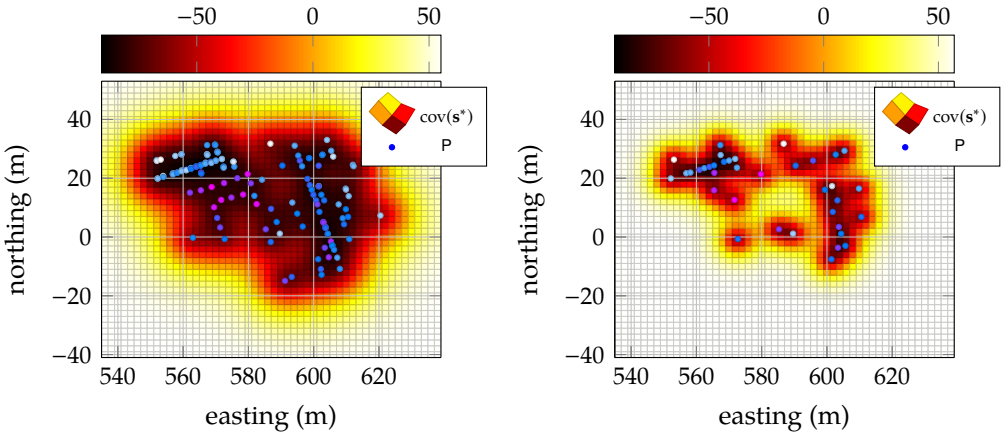
(a) Predicted RSSI mean of a model that was trained with the full set of its fingerprints.

(b) Predicted RSSI mean of a model trained with the fourth part of its fingerprints.

Figure 3.7.: RSSI predictions from two Gaussian process regression models trained with a different amount of training data for a single access point. Both models are based on the same prior distribution with constant mean and Matérn$_{\nu=3/2}$ function. The magnitude of the mean function is indicated by the colour bars. Large training data are coloured purple–dark-blue and low RSSI light-blue–white.

to the data, the principle features are present. They are less pronounced, though. Both mean functions show the peak RSSI at [560, 580] m × [0, 20] m and also the areas of low RSSI within and behind the two buildings (see figure 3.1 for comparison) are present. Major differences are visible in areas where no training data exist: about [580, 600] m × [0, −20] m and in the surrounding of (620, 7) m. It causes the local maximum visible in panel 3.7a about (592, −14) m and parts of the local minimum behind the right building to be smoothed over. The magnitude of RSSI towards the edges of the test bed is also slightly lower.

However, information about this degradation of accuracy is contained in the covariance function, shown in figure 3.8. Comparing the covariance function of



(a) Predicted RSSI variance of a model trained with all its fingerprints.

(b) Predicted RSSI variance of a model that was trained with the fourth part of its fingerprints.

Figure 3.8.: RSSI covariance function from two Gaussian process regression models trained with a different amount of training data for a single access point. Both models are based on the same prior distribution with constant mean and Matérn$_{\nu=3/2}$ function. The magnitude of the covariance function is indicated by the colour bars. Large training data are coloured purple–dark-blue and low RSSI light-blue–white.

panel 3.8a and panel 3.8b discloses the corresponding increase of variance in the areas affected by that reduction. The information that the covariance function provides is essential for the main objective of this work, the hybrid positioning system, because it allows an automatic weighting of the different systems that are

to be fused.

The figures 3.7 and 3.8 also show the main advantage of RSSI interpolation compared to basic fingerprinting. Namely that despite reducing the number of fingerprints of the radio maps, an accurate radio map can still be provided. The effect of the decrease of accuracy of the radio map on the localisation performance is investigated in section 3.4.2.

Residuals

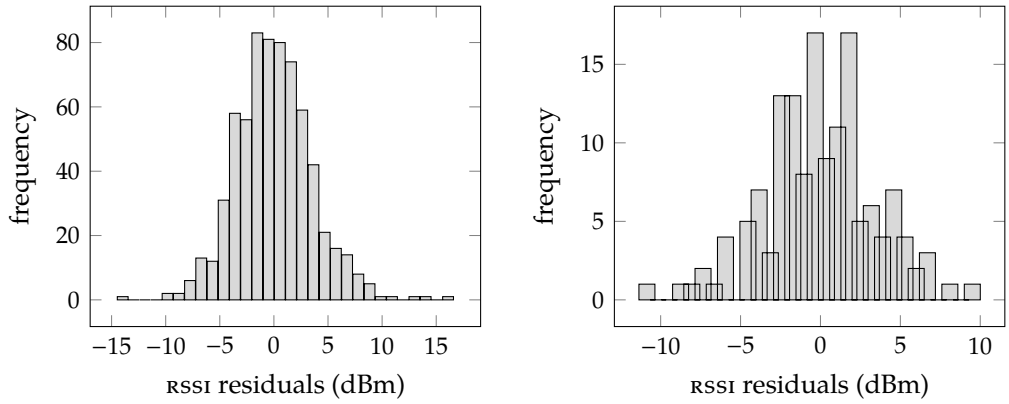
The chosen Gaussian process prior distribution yielded reasonable results, as shown in the previous section. Nevertheless, the model we used is based on assumptions that may be disputable. In this section we like to assess qualitatively the modelling errors and how much the data deviates from the made assumptions.

We explore the residuals of the Test-bed data, for the preferred prior distribution (Const, $\text{Mat}_{\nu=2/3}$). The data we present as follows belongs to two access points we used as well in the previous section to illustrate the model selection: one access point that contributes a lot fingerprints to the radio map, and an other access point that contributes few fingerprints to the radio map. The training data for these two access points, AP_D and AP_S, are shown in figure 3.6a and figure 3.7a.

Recall equation (2.25), if the RSSI observations are modelled by a Gaussian likelihood function, as we did, the Gaussian process regression model residuals ought to be normally distributed, too. We evaluate the histograms and the norm probability plots of the residuals to see if the assumption of normally distributed RSSI is valid.

The histograms for the two access points are shown in figure 3.9. The residuals are expected to be about normal distributed with mean zero. An important observation is that in both cases the histograms are symmetric and the residuals have a mean around zero. But, the distributions are to heavy tailed for normal distributions. The histogram on the right hand side (3.9b) presents even multiple modes, possibly caused by the small sample size.

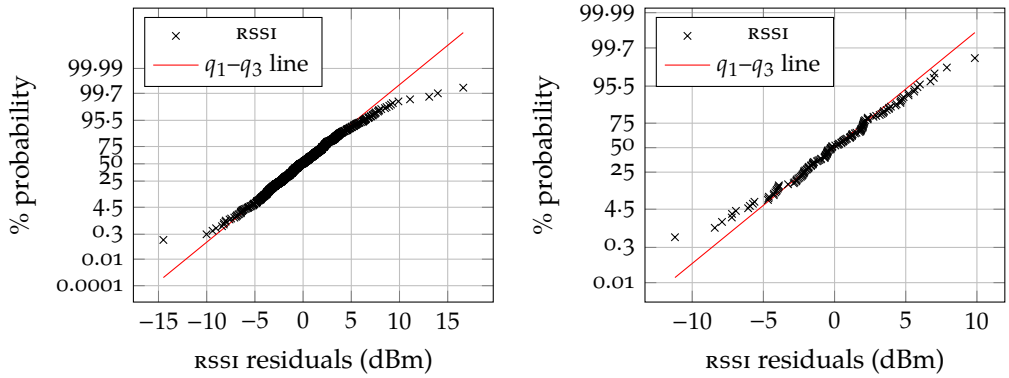
More details can be observed in the norm probability plot in figure 3.10. (Perfect normal distributed data would lie on the red line.) The statements based on the histograms can be confirmed: The distribution of the residuals is symmetric around zero, with heavy tails. In panel 3.10a the majority of the residuals, the residuals between -5 dBm to 5 dBm, coincide with the line indicating normality, thus, they follow a normal distribution. The remaining residuals smaller than



(a) Histogram of residuals corresponding to an access point with many fingerprints.

(b) Histogram of residuals corresponding to an access point with few fingerprints.

Figure 3.9.: Histogram of RSSI residuals of two Gaussian process regression models, each for a different access point.



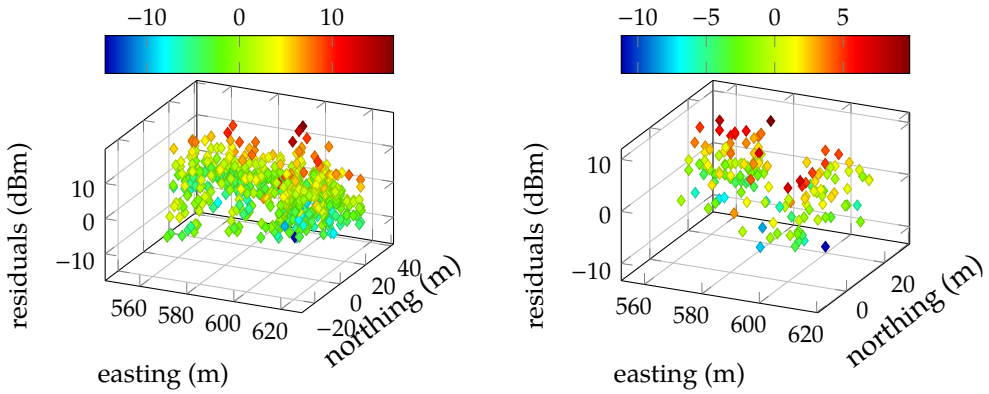
(a) Norm probability plot of the residuals which correspond to an access point with many fingerprints.

(b) Norm probability plot of residuals which correspond to an access point with few fingerprints.

Figure 3.10.: Norm probability plot of the RSSI model residuals from Gaussian process regression models for two different access points.

±10 dBm deviate slightly from that line. The number of outliers is relatively small, about 1.5 % of the sample. One could describe the model residuals as approximately normal distributed. The residuals of APS share the same properties, only the maximum values are smaller.

The distribution of the residuals is one issue, an other one is their spatial independence. In figure 3.11 the residuals of the two access points are graphed as function of space.



(a) Residuals corresponding to an access point with many fingerprints.

(b) Residuals corresponding to an access point with few fingerprints.

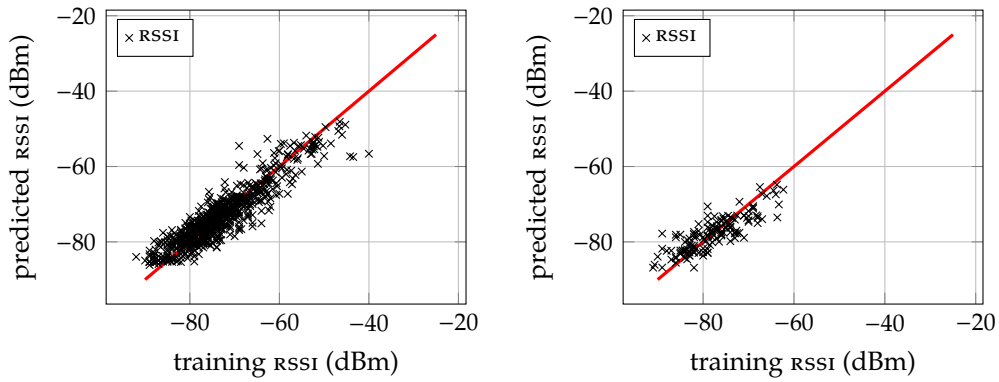
Figure 3.11.: Rssi model residuals over space of the test area for two access points. The magnitude of the residuals are indicated by the colour bars.

Several of these figures were visually examined and compared. However, a definite statement, whether the residuals are spatially independent or not, is difficult to provide, as no perspective allows an appropriate assessment. Additionally, the unknown locations of the access points, which affect the spatial distribution of the residuals, complicate such a inspection. Nonetheless, the spatial residual distributions that were examined and compared did not suggest that the residuals possess spatial patterns.

What in addition can be seen in the example figures 3.6a and 3.7a, is that the residuals from APD present a larger variance than the residuals from APS. The reason is the larger variance of the RSSI if the power of WLAN signals is high, compared to the variance of low power RSSIs.

Figure 3.12 presents the predicted RSSI as function of th RSSIs used to train

the model. If the predictions matched the training target values perfectly, all points would lie on the red line. A random distribution around the line of ideal



(a) RSSI measurements versus predictions for an access point with many fingerprints.

(b) RSSI measurements versus predictions for an access point with few fingerprints.

Figure 3.12.: Training RSSI versus predicted RSSI for two access points. The red line indicates under and overfitting of the training data.

predictions indicates an appropriate adaption to the training data. Nevertheless, towards the ends of the function becomes asymmetric for both access points.

Just below −40 dBm in panel 3.12a and below −60 dBm in panel 3.12b), the RSSI lean towards the training data. That is, the predictions are smaller the training RSSI, they do not 'reach' the training data. This suggests that the model underestimates the RSSI observations around the maximum RSSI nearby the access points. This effect is explainable by the way the radio maps were recorded, namely that eventually four fingerprints at the same position was captured. If these RSSIs varied too much while being very close, the process would not be able to adapt. Notice that not all access points present these outliers.

The opposite effect is visible at the lower end of RSSI. Here, the points lean towards the predictions, meaning that the predicted RSSI are smaller than the training data. It indicates that the training data is underestimated by the model; note, this time at the lower RSSI bound. Only, a few access points showed that effect.

Positioning accuracy

The previous findings favour the Gaussian process model based on the Matérn kernel, but these findings rely on the model selection criteria only and do not allow conclusion about the positioning accuracy or precision. This issue is addressed in this subsection. Through the positioning performance of the ML estimator we seek confirmation about choice of the Gaussian process model.

To see whether or not the positioning performance reflects the results of the previous sections, we derive ML estimators and compare the resulting RMSEs. Each estimator is derived according to equation (3.4): For a certain Gaussian process model we compute first the RSSI value on a regular grid of the test area. Then, at each time instance of the trajectory, we use the corresponding RSSIs measurement to compute the likelihood function. The likelihood functions are computed based on Gaussian processes with the constant mean function and the eight different covariance functions we examined in the previous section. The position at the maximum of the likelihood function is the final position estimate of that time instance.

Table 3.6 compares the RMSE that resulted from that procedure for Trajectory-1 and Trajectory-2, see figure 3.2. The unequivocally highest accuracy achieved the

Table 3.6.: RMS position error for Trajectory-1 and Trajectory-2. RMSEs are from ML estimators based on Gaussian process models with different prior covariance function, using the Test-bed radio map.

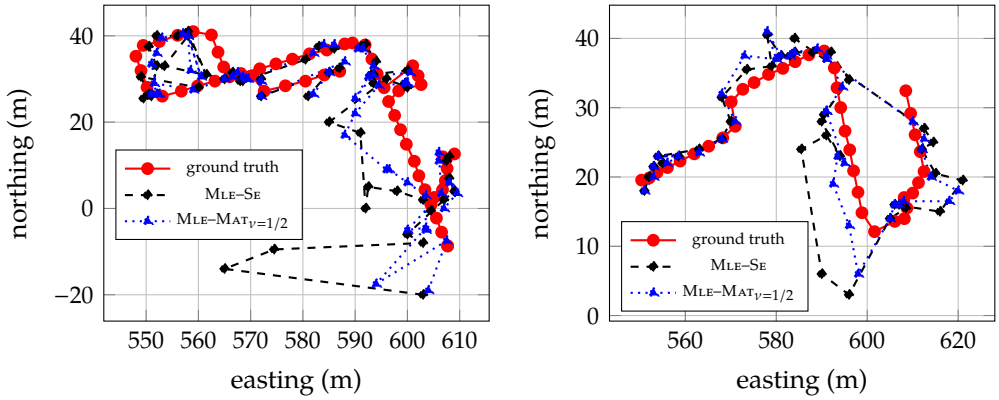
ML estimator	Position error (m)	
	Trajectory-1	Trajectory-2
SE	10.21	6.46
$\text{MAT}_{\nu=1/2}$	**5.90**	**3.97**
$\text{MAT}_{\nu=3/2}$	8.90	5.68
$\text{MAT}_{\nu=5/2}$	8.98	5.92
RQ	6.18	5.30
SUM SE & SE	10.48	6.46
PROD SE & SE	10.46	6.35
ADD SE	14.42	11.45

estimator that uses the $\text{Matérn}_{\nu=1/2}$ covariance function. The second best result

achieved the rational quadratic covariance function; a covariance function which we excluded quite early as a promising candidate to model RSSIs. Also the other two Matérn functions perform reasonably well. The more complex functions and the squared exponential covariance function yielded the lowest accuracies.

These results reflect well the outcomes from the model selection criteria, see table 3.4. In particular, that the error of the estimator using the squared exponential covariance function is comparable to the errors of the more complex covariance functions. The error of the estimator using the squared exponential covariance function is about 40 % higher than that of the most accurate ML estimator. This corresponds to improvements of 1.3 m, in case of Trajectory-1, and 0.78 m for Trajectory-2.

The estimated trajectories are presented in figure 3.13. They show the northing over easting for two estimators. One ML estimator uses the Matérn$_{\nu=1/2}$ covariance function the other estimator the squared exponential covariance function. Both estimators have again the constant mean function in common. The ground truth is depicted as reference. The indoor/outdoor trajectories are in general estimated



(a) Estimates of Trajectory-1.

(b) Estimates of Trajectory-2.

Figure 3.13.: Estimated trajectories from two Gaussian process based ML estimators, using different prior covariance function, for two trajectories. The ML estimators are based on a constant mean (CONST) and 1) squared exponential (SE), or 2) Matérn (Mat$_{\nu=1/2}$) kernel. The Gaussian processes are trained with the complete radio map. The ground truth is depicted as reference.

well. The typical problems of localisation methods based on fingerprinting is

visible. In open areas fingerprints are not well distinguishable due to insufficient signal attenuation. Neighbouring fingerprints and even more distant fingerprints are alike. A result are the outliers visible for both trajectories, occurring in the vicinity of open spaces. In Trajectory-1 they are visible at the beginning about (607, −9) m, along the pathway of the right building [585, 600] m × [0, 25] m (which is open towards the large open space) and around the small lawn area [550, 565] m × [25, 41] m. The estimates for Trajectory-2 present these outliers also along the pathway of the right building [585, 600] m × [5, 30] m and behind the right building [610, 620] m × [15, 20] m. Especially the estimate of the last position deviates, it jumped about 25 m to (585, 40) m. In these cases the estimates are pulled towards the open areas.

The indoor sections of the trajectories are well estimated. This is observable in panel 3.13a at the entrances, which are located about (605, 2) m, (595, 27) m and (570, 30) m. Trajectory-2 constitutes only one indoor section, which is also estimated well, panel 3.13b illustrates that, see [555, 30] m × [572, 27] m.

These two panels also reveal that in sections of the trajectories where the accuracy is high, both estimators perform equally well. But the large position errors produced by the Matérn function become even larger when the squared exponential covariance function is used.

These results confirm the insights from the previous section. 1) The model differences are large enough propagate via the likelihood function and are reflected in the localisation error. The choice of the Gaussian process model influences the accuracy of a WPS. 2) The Matérn covariance function is a very good choice to interpolate RSSIs, though, based on the model selection criteria, we favoured the Matérn covariance function with $\nu = 3/2$ instead of $\nu = 1/2$. This suggests additionally that RSSI variations are quite large and therefore to use rather rougher kernels to model RSSIs.

3.4.3. Positioning performance of WPS

In this section, we report more general about the positioning performance of the Gaussian process based WPS. This section shows the usefulness of interpolating the radio maps to decrease the labour of its construction. We contrast the positioning accuracy and precision for the dense and the sparse radio map of the Test-bed. (It is the radio map used for the Gaussian process model selection experiments.) These findings provide an idea about the (surprising) effect of the

positioning performance, if one reduces the amount of fingerprints.

The positioning performance is evaluated by means of localisation accuracy and precision. We present the results for the ML estimator based on our suggested model with constant mean and Matérn$_{\nu=3/2}$ covariance function. To set the outcomes in context, it is compared with the ML estimator that relies on the zero mean and squared exponential covariance function and also with the k-NN algorithm. We chose these two algorithms as benchmark because they are used commonly for WLAN fingerprinting. The two ML estimators are denoted by MLE–CONST & MAT and MLE–ZERO & SE. We show results again for the two trajectories, Trajectory-1 and Trajectory-2.

The accuracy in terms of the RMSE is summarised in table 3.7. It contrasts RMS position error yielded for the dense and the sparse radio map. For the dense radio

Table 3.7.: RMS position error of three different position estimators employing the dense or the sparse radio map for Trajectory-1 and Trajectory-2. © IEEE 2015

	error Trajectory-1 (m)		error Trajectory-2 (m)	
	dense	sparse	dense	sparse
k-NN	7.06	13.41	4.63	12.79
MLE–ZERO & SE	9.92	5.57	6.45	4.65
MLE–CONST & MAT$_{\nu=3/2}$	8.90	5.10	5.68	4.19

map the k-NN estimator outperforms the two ML estimators. Comparing the ML methods, the Maximum likelihood estimator (MLE)–CONST & MAT yields lower position error. For the sparse radio map the situation is different. Now, both ML estimator outperform the k-NN algorithm. The MLE–CONST & MAT yields again higher accuracy than the MLE–ZERO & SE. These outcomes as well corroborate the results of the model selection section.

The good performance of the k-NN method indicates that the fingerprint density of the dense radio map is relatively high, though, it is a little unexpected. The degradation of the k-NN's accuracy with the reduction of fingerprints is because it works directly on the discrete fingerprints. In the case of the dense radio map enough fingerprints are available to attain a high accuracy. In the case of the sparse radio map the fingerprints are too separated and too sparse. This leaves not enough candidate positions for the solution space to achieve a similar accuracy.

It is noteworthy that MLES's performance improve if the amount of fingerprints is decreased. For Trajectory-1 the error of both ML estimators is reduced by factor 1.8 and for Trajectory-2 the accuracy improvement is factor 1.4. This leads to the interpretation of the accuracy results of the dense radio map that not the k-NN algorithm performs exceptionally good, but that the MLES perform for some reason poor. A possible explanation for that requires further analysis which follows in the next sections.

Dense radio map

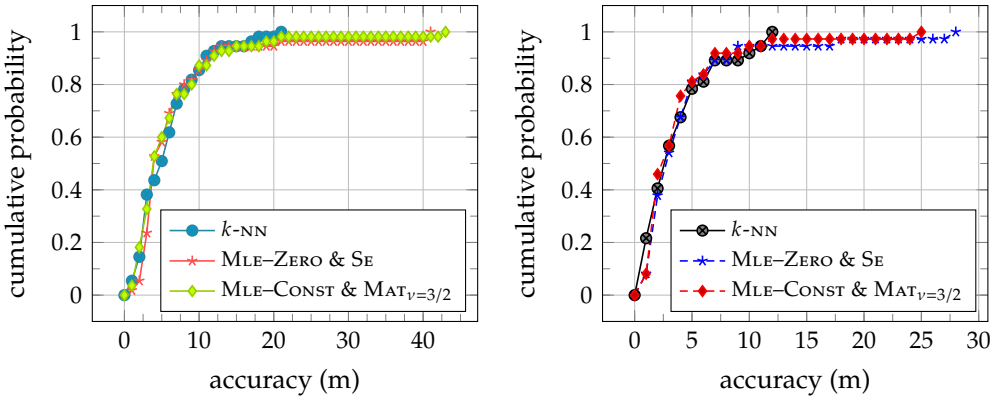
The findings we present in this subsection rely all on the dense radio map. It contains the full set of fingerprints as they were recorded. Panel 3.6a and 3.7a exemplify the density of fingerprints for one access point. The first panel shows an access point contributing many fingerprints to the database, because its signals are received at almost any place of the test area. The latter panel shows an access point whose signals where not captured often during radio map construction.

The positioning performance of the three estimators are represented in form of the empirical CDF of the error. Figure 3.14a depicts the accuracy for the dense radio map. Panel 3.14 shows it for Trajectory-1 and panel 3.14b shows it for Trajectory-2.

The first interesting observation is that the three estimators perform up to 90 % nearly identical; and this can be seen for both trajectories. The estimators differ mainly in that they have larger or smaller maximum errors. However, for Trajectory-2 Mle–Const & Mat shows occasionally higher accuracy of 0.5 m–1 m below 75 %. The median accuracy for Trajectory-1 is about 4 m, for Trajectory-2 is is just below 3 m. The error at 90 % is 12 m for Trajectory-1 and 7 m for Trajectory-2.

More significant differences between the estimators can be observed above 90 % localisation accuracy. One can observe that the small RMSE of the k-NN is attributed to the smallest maximum errors. They are about twice as small than that of the ML estimators: 21 m in case of Trajectory-1, whereas the MLES exhibit maximum errors about 43 m; and 12 m for Trajectory-2, whereas the ML estimators yielded about 25 m.

Consider the CDF in panel 3.14b. The estimator based upon the 'default' Gaussian process model (Mle–Zero & Se) presents about 3 m smaller maximum error than Mle–Const & Mat, albeit in table 3.7 its average error exceeds that of Mle–Const & Mat by 1 m. It suggests that Mle–Const & Mat's errors are



(a) Cdf of the rms error of three positioning estimators for Trajectory-1.

(b) Cdf of the rms error of three positioning estimators for Trajectory-2.

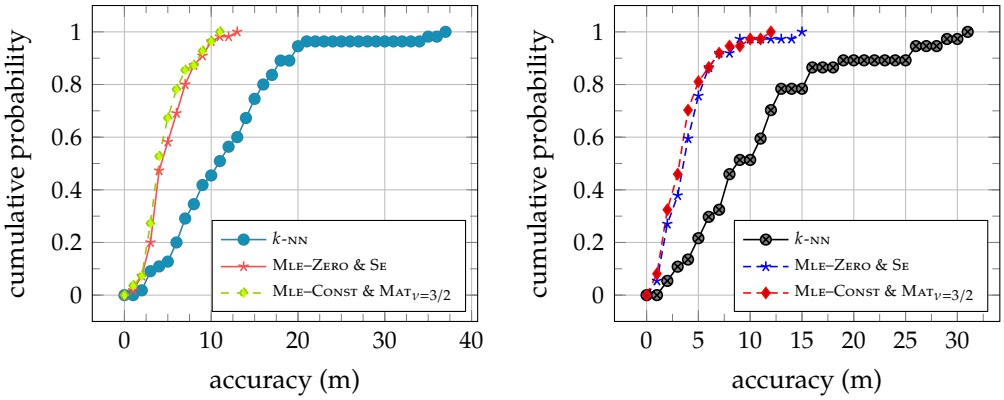
Figure 3.14.: Empirical cdf of the rmse of two trajectories examined for three different positioning algorithms relying on the dense radio map. The ml estimators are based on a constant mean (Const) and 1) squared exponential (Se), or 2) Matérn (Mat$_{\nu=3/2}$) kernel and used estimators used the dense radio map. © ieee 2015

in general smaller. In the case of Trajectory-2, Mle–Const & Mat outperforms Mle–Zero & Se consistently.

Sparse radio map

Positioning performance for the undersampled radio map, the radio map obtained by randomly removing the fourth part of the fingerprints, is described as follows. The data we present here, is much more relevant for a practical, eventually large-area wps, since the fingerprint density of the dense radio map is unrealistic for such cases (Of course, the actual fingerprint density depends on the concrete application.). A one access point sample of fingerprints from the sparse radio map can be found in figure 3.7b.

The cdf that shows performance of the three estimators for the sparse radio map is shown in figure 3.15. The performances of the estimators for Trajectory-1 and Trajectory-2 are very similar. The k-nn algorithm presents now very low accuracy and the two mles perform almost the same. The error of Mle–Const & Mat is generally slightly smaller than that of Mle–Zero & Se, but in particular



(a) Cdf of the rms error of three positioning estimators for Trajectory-1.

(b) Cdf of the rms error of three positioning estimators for Trajectory-2.

Figure 3.15.: Empirical cdf of the rmse of two trajectories examined for three different positioning algorithms relying on the sparse radio map. The ml estimators are based on a constant mean (Const) and 1) squared exponential (Se), or 2) Matérn ($\text{Mat}_{\nu=3/2}$) kernel. © ieee 2015

the difference in maximum error demonstrates again its superiority.

The accuracy of the k-nn estimator is not competitive any more, its median error is about twice and its error at 100 % is trice as large than that of the mles. Whereas the median error of the mles are roughly 3 m to 4 m for both trajectories. Their accuracy at 90 % is 9 m for Trajectory-1 and 7 m for Trajectory-2, respectively. Their largest deviations from the ground truth are 12 m for both trajectories.

A comparison of the outcomes for the two radio maps provides more insights:

Table 3.8 contrasts the accuracy of the k-nn and the mles for the dense and the sparse radio map.

First of all, the results for the dense radio map are significantly different to that of the sparse radio map.

One point is the accuracy decrease of the k-nn performance. It is expected and already discussed.

The most striking point is that a reduction of the number of fingerprints by factor four, did not or almost not change the errors of the mles below 80 % probability. The maximum errors even decrease because of the reduction of fingerprints. They decrease considerable: by factor 3.3 in case Trajectory-1 and

Table 3.8.: Statistics on the position error of three different position estimators employing the dense or the sparse radio map for Trajectory-1 and Trajectory-2.

		error Trajectory-1 (m)		error Trajectory-2 (m)	
		dense	sparse	dense	sparse
median	k-NN	5	11	3	9
	MLE	4	4	3	3
90 %	k-NN	11	19	10	25
	MLE	12	9	7	7
100 %	k-NN	21	37	12	31
	MLE	43	12	25	12

factor 2.3 for Trajectory-2, respectively. The fingerprint density does not affect the mean error, but influences substantially the robustness of the ML estimators. The decrease of their average RMSE, shown in table 3.7, is principally caused by that increased robustness.

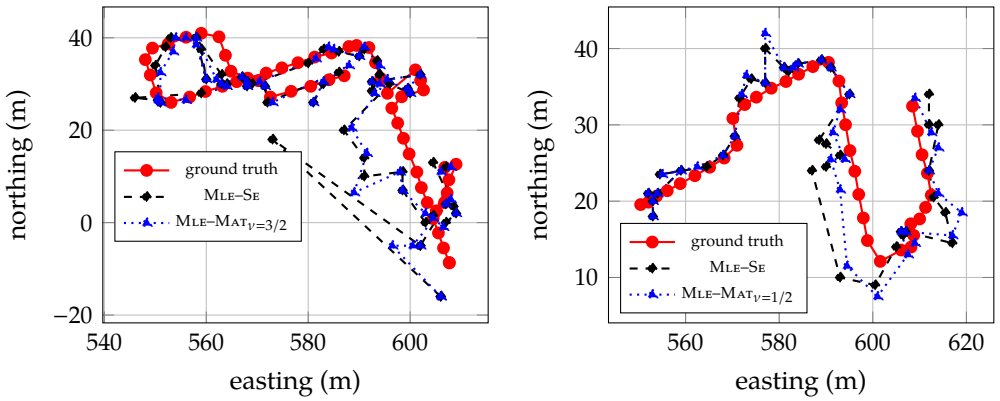
This increase of robustness is attributed to the ambiguity problem fingerprinting methods have. Very large errors in fingerprinting systems are caused by this issue, in fact, it occurs when similar RSSI patterns are recognised at distant locations. We verified this by examining several likelihood functions for the dense radio map. They showed repeatedly maxima distant from the actual location. Removing fingerprints from the radio map potentially also removes ambiguous fingerprints, while the amount of fingerprints is still high enough so that the Gaussian process interpolation can provide sufficiently exact RSSIs at these locations – therefore the position accuracy can be maintained and even improved. Thus, Gaussian process ML estimators based upon the sparse radio map limit the influence of ambiguous RSSI on the position estimate better than the ML estimators based on the dense radio map. However, it is expected that below a certain amount of fingerprints the quality of RSSIs predictions declines drastically, such that the accuracy deteriorates until the position error is only confined by the borders of the radio map.

Last but not least, one may have noticed that the k-NN algorithm does not suffer much from the ambiguity issue. The reason is in the distance penalisation term we introduced. It adds a second feature to the algorithm, namely the number

of access points common in the observations and in the database. This second feature helps to distinguish fingerprints similar in their rssi and diminishes erroneous recognition of fingerprints. [12]

Note, in realistic scenarios the actual density of fingerprints is nearly impossible to control, because the signal reception conditions change randomly in time and space. Thus, the robustness of the estimators to a varying fingerprint density that is a further advantage and reason to build a wps on Gaussian process predictions.

We illustrate concluding the trajectories in easting and northing. Figures 3.16 shows the estimates from the two ml estimators, where the Gaussian processes are trained with the sparse radio map. In panel 3.16a depicts Trajectory-1 and panel 3.16b shows Trajectory-2. Comparing these results with the ones derived



(a) Estimates of Trajectory-1.

(b) Estimates of Trajectory-2.

Figure 3.16.: Estimated trajectories from two Gaussian process based ml estimators trained with the sparse radio map and using different prior covariance function. The ml estimators are based on a constant mean (Const) and 1) squared exponential (Se), or 2) Matérn (Mat$_{\nu=1/2}$) kernel. The Gaussian processes are trained with the sparse radio map. The ground truth is depicted as reference.

from the dense radio map in figure 3.13, a reduction of extreme position outlier is observable, in particular for Mle–Const & Mat (even though ν is chosen to $3/2$).

In addition, one can see in both trajectories that Mle–Zero & Se shows larger

12. Beder et al. (2012) introduced such an approach into the likelihood function for their probabilistic fingerprinting system. They model the number of common rssi with a Gibbs distribution and integrated it into their likelihood function.

outliers than Mle–Const & Mat. We like to stress again the superiority of the ml estimator derived from a Gaussian process with constant mean and Matérn covariance function over the estimator relying on the 'default' Gaussian process model. Mle–Const & Mat constitutes lower maximum error than Mle–Zero & Se, an exception though is the case Trajectory-1 with dense radio map. That means that it can deal better with larger rssi variations and is hence more robust. This is coherent, considering that the Matérn kernel is composed of an exponential and a polynomial term. It makes it more flexible, hence also wider applicable than the squared exponential kernel. The greater flexibility facilitates the adaption to training data with large rssi differences within small distances. The ability to model rssi well translates to the positioning performance and confines localisation errors. In comparison, the zero mean function of Mle–Zero & Se additionally deteriorates the fit and contributes to the described effect. Moreover, using the constant mean function makes an additional estimation steps as described in (Ferris et al. 2006) and (Atia et al. 2013) unnecessary.

The insights gained from the analysis of the localisation performance indeed reflect the findings from the Gaussian process model selection – with respect to the choice of the prior distribution but also with respect to the density of fingerprints.

The presented indoor/outdoor wlan fingerprinting positioning system yields a median accuracy of only 4 m. We consider this an acceptable, even competitive result.

3.5. Conclusion

Gaussian processes model accurately the spatial structure of wlan rssis. They provide a robust and versatile method to predict/interpolate radio maps for location fingerprinting systems, notwithstanding the flawed assumption of normally distributed rssis.

3.5.1. Distribution of rssi

As a conclusion from the Single-ap experiment, we see it justified to approximate averaged rssis with a Gaussian distribution. Using these averaged rssis to parametrise the Gaussian process likelihood function, results in more robust predictions as if raw rssis were used.

However, it is indicated that more robust Gaussian process likelihood functions may improve the RSSI models. The Student's t-distribution or the Laplace distribution, for example, seem promising and have been applied in Gaussian process regression. A non-Gaussian likelihood function will increase the computational burden of fitting the model, because approximations are needed to infer the hyperparameters.

3.5.2. Structure search

The spatial structure of RSSIs is relatively simple, since simple covariance functions, such as the SE kernel, are able to fit the data better than more complex covariance functions. This is reasonable in the light of the exponential decay of RSSI and the indistinguishability of spatial RSSI distributions of indoor and outdoor environments.

The search for the most appropriate Gaussian process prior distribution resulted in the choice of the constant mean and the Matérn$_{\nu=3/2}$ covariance function.

Choosing the constant mean function over the zero- and linear mean function is also theoretically justified. The zero mean function is obviously questionable, because the access point's maximum RSSIs is commonly at least 20 dBm below zero and power decreases further with increasing distance from the access point. Thus, a convergence to zero (in the absence of training data) is not being appropriate for RSSI. The linear mean function may model the RSSIs well if the access point is located at the borders of the test area. In these cases the linear decay approximates the one-sided exponential decay. But as the model selection criteria of the linear mean function were quite close to that of the constant mean function and its the complexity is larger, we favour the constant mean function.

The choice of the parameter ν is also a compromise. Based on our data, we believe that a Matérn covariance function with $\nu = 3/2$ is more general applicable than the rougher ($\nu = 1/2$) or smoother ($\nu = 5/2$) version. The property of the Matérn functions to model a certain roughness is also the reason to not recommend the squared exponential kernel, despite its very common use. Only for very sparse radio maps, the squared exponential kernel is an appropriate choice. In that case the training data points are distant from each other and are therefore sufficiently smooth to be modelled by that kernel.

We conclude that the 'default' Gaussian process model, employing zero mean function and squared exponential covariance function, is usable, but not the best

choice. The reasons are already mentioned: the undesired convergence to zero of the zero mean function and the inability of the squared exponential kernel to model large RSSI variations. Moreover, the positioning performance confirmed the superiority of using a Gaussian process prior with constant mean and Matérn covariance function.

Finally, we did not find any indication that the RSSIs of indoor or outdoor data sets have different structure. Neither the test error nor the BIC presented patterns that suggest the use of different prior distributions; an examination of the hyperparameters did not change that conclusion. The possible reasons that the spatial RSSI distributions of indoor and outdoor environments are indistinguishable are: 1) the difference in RSSI we expect, is not present in our environment; 2) the RSSI training data did not capture it; or 3) methodological errors bury it. Causes for each possibility are listed subsequently:

Difference not existing in that environment The building's geometry and size, its structure or materials (single brick outer walls and mainly soft partitions indoors), or the contained objects lead to indifferent spatial RSSI distribution for indoor and outdoor environments.

Difference not contained in data The spatially discrete distribution of fingerprints is to scarce or geometrically disadvantageous to capture the difference between indoor/outdoor RSSI distribution.

Possible as well is that the chosen Gaussian process models are too insensitive to model the existing differences between indoor/outdoor RSSI distribution. They basically smooth over the these differences.

Methodological errors To compute the model selection criteria, we averaged over all available access points. Note, the spatial fingerprint distribution for each access point differs and also dependents on the position of the access point. The effect we expect is perhaps contained in the data of some access points, but not in these of others. Therefore, averaging over all access points may remove the spatial RSSI differences.

Thus, this calls for an analysis for each access point, possibly on a regular and finer grid of fingerprints. (The latter point is yet hard to achieve in realistic scenarios.)

A combination of these reasons is conceivable, too.

Having said that, the indiscernible of spatial indoor/outdoor RSSI distribution bears the advantage that a common model for both environments is sufficient, which (in addition to the resource savings due to the interpolation) saves time when deploying a WPS.

3.5.3. Localisation performance

The positioning performance of the ML based fingerprinting system was analysed, where the ML estimators were derived from Gaussian processes. The positioning system achieved 4 m median error and 6 m mean error for the indoor/outdoor trajectories.

The evaluation of the positioning performance confirmed the previous conclusions: The recommendation for a Gaussian process prior with constant mean and Matérn covariance function. It demonstrated additionally that the MLE using the Matérn covariance function outperforms the ML estimator relying on the 'default' Gaussian process model, because of its greater robustness. The Gaussian process model with zero mean and squared exponential covariance function is unfavourable for RSSI modelling.

Furthermore, a low radio map density can improve localisation performance compared to a dense one; in particular if the RSSI are modelled adequately. The reduction of fingerprints diminishes the ambiguity issue due to very similar fingerprints in a radio map. This affects positively the robustness of the WPS, increases in average its accuracy and precision.

Chapter 4

Particle filter fusing WLAN RSSI and GNSS pseudoranges

In the previous chapter, we explained the WLAN based positioning system using interploated RSSIs and the ML method for self-localisation. In this chapter we present the proposed information fusion algorithm to blend GNSS pseudoranges with RSSIs based upon the RSSI likelihood function of chapter 3. The proposed method combines pseudoranges and RSSIs by employing a particle filter that fuses the likelihood functions of RSSIs and pseudoranges in its measurement update step.

The following section describes subsequently the probabilistic models used in the particle filter; then in section 4.2 we finally derive the particle filter. Section 4.3 presents the experiments made to evaluate the fusion filter and section 4.4 delivers results on the filter performance – compared to the performance of the filter in the GPS-only or WLAN-only setting.

4.1. State-space model

The state space for our self-localisation is naturally three-dimensional space. To model the dynamic of the object that is to be located we incorporate its position coordinates, $\mathbf{p} = (x, y, z)^T$, in a local level coordinate frame and its velocity, $\dot{\mathbf{p}} = (\dot{x}, \dot{y}, \dot{z})^T$ respectively. Due to the differences between receiver and satellite clocks the receiver clock bias must estimated too. Hence, the state vector is augmented with receiver clock offset itself, t_r, and its time derivative, the clock

drift $\dot{t}_r$. The state vector becomes

$$\mathbf{x}_k = (\mathbf{p}_k, \dot{\mathbf{p}}_k, t_k, \dot{t}_k)^T. \tag{4.1}$$

This 8-state vector is a typical choice for a PVT of a GPS receiver (Kaplan et al. 2006). Note, this state vector is defined on continuous space, in particular $\mathbf{p} \in \mathbb{R}^3$ instead of the discrete subspace $\check{P}$ on which the fingerprints are defined (see sec. 2.5.2). The next section presents the process model, followed by the measurement model.

4.1.1. Process model

So far we did not specify what we want to localise and spoke generally about an object, robot or a person. There is no practical general model that models all possible dynamics. In the field of robotics, motion models have usually an additional input variable which is the result of a motion command and as such connected to some sensory data such as odometer or accelerometer data. Hence, except of some uncertainty the robot already knows what motion occurred between the previous and the current time step. As well the motion of a pedestrian can be inferred from sensors carried by him. A set of inertial sensors enables dead reckoning based on detected strides, their length and the relative direction of the user. Also models based on human behaviour have been developed. They require additional information about the user and its state/condition that must be known to the system at the outset; examples for these parameters are age, disorientation and weather, weekday or ground steepness. Another class of approaches rely on stochastic models. Additional sensors and user information is not required for these models. These stochastic models are general enough to capture a great variety of dynamics and can be adapted to the application with few parameters.

To simplify the experimental phase we have chosen a pedestrian to carry the positioning system, consequentially we need to model low dynamics as performed by pedestrians. Such a dynamic can be described linearly with additive, normal distributed noise. Thus, we have chosen a Langevin process as in (Vermaak et al. 2001) to propagate the position and velocity in time.

It is a stochastic differential equation to describe the velocity of a Brownian

particle, for a single dimension its discrete version reads

$$\begin{aligned}\dot{\mathsf{x}}_k &= a_\mathsf{x}\dot{\mathsf{x}}_{k-1} + b_\mathsf{x} v_{\mathsf{x}_k},\\ \mathsf{x}_k &= \mathsf{x}_{k-1} + \Delta T\dot{\mathsf{x}}_k,\end{aligned}$$

with

$$a_\mathsf{x} = \exp(-\beta_\mathsf{x}\Delta T), \quad b_\mathsf{x} = \bar{v}_\mathsf{x}\sqrt{1-a_\mathsf{x}^2}.$$

The excitation process is white and normally distributed $v_{\mathsf{x}_k} \sim \mathcal{N}(0,1)$ and ΔT is the discretization step. The parameters are $\bar{v}_\mathsf{x}$, the steady state RMS velocity (the velocity the pedestrian moves at most) and β_x the process' rate constant. Experiments showed that $\beta_{\mathsf{x},\mathsf{y}} = 10\,\mathrm{s}^{-1}$, $\beta_\mathsf{z} = 100\,\mathrm{s}^{-1}$ and $\bar{v}_{\mathsf{x},\mathsf{y}} = 1.5\,\mathrm{m\,s}^{-1}$, $\bar{v}_\mathsf{z} = 0.5\,\mathrm{m\,s}^{-1}$ provide good positioning performance.

This equation can be interpreted as a walker moving with a certain step length. The zero mean of the noise term implies that all direction are equally likely. This model has an other advantage: If no observations are available the uncertainty of the objects position increases because the particle filter only predicts the position. During pseudorange and RSSI outages this model accounts for that fact by increasing the variance over time.

The process model for the position and velocity can be denoted by

$$\begin{pmatrix}\mathbf{p}_k\\ \dot{\mathbf{p}}_k\end{pmatrix} = F_p\begin{pmatrix}\mathbf{p}_{k-1}\\ \dot{\mathbf{p}}_{k-1}\end{pmatrix} + G_p\mathbf{v}_{p,k}, \tag{4.2}$$

where the noise is Gaussian and uncorrelated $\mathbf{v}_{p,k} \sim \mathcal{N}(0, I)$. The state transition matrix becomes

$$F_p = \begin{pmatrix} 1 & 0 & 0 & a_\mathsf{x}\Delta T & 0 & 0\\ 0 & 1 & 0 & 0 & a_\mathsf{y}\Delta T & 0\\ 0 & 0 & 1 & 0 & 0 & a_\mathsf{z}\Delta T\\ & & & a_\mathsf{x} & 0 & 0\\ & 0_{3\times3} & & 0 & a_\mathsf{y} & 0\\ & & & 0 & 0 & a_\mathsf{z}\end{pmatrix}$$

and the noise coupling matrix

$$G_p = \begin{pmatrix} b_x & 0 & 0 \\ 0 & b_y & 0 \\ 0 & 0 & b_z \\ b_x \Delta T & 0 & 0 \\ 0 & b_y \Delta T & 0 \\ 0 & 0 & b_z \Delta T \end{pmatrix}.$$

As well the receiver clock offset obeys linear laws, also with normal distributed noise. We use a standard clock model where clock phase and clock frequency obey a random walk.

$$\begin{pmatrix} t_k \\ \dot{t}_k \end{pmatrix} = F_c \begin{pmatrix} t_{k-1} \\ \dot{t}_{k-1} \end{pmatrix} + G_c \mathbf{v}_{c,k},$$

where

$$F_c = \begin{pmatrix} 1 & \Delta T \\ 0 & 0 \end{pmatrix}$$

and the process noise component for the clock model is again $\mathbf{v}_{c,k} \sim \mathcal{N}(0, I)$. For the noise coupling term we follow Brown et al. (1997, pp. 428–432), who states a correlation of the noise terms for the clock phase and clock frequency. The noise covariance matrix Q_c is a 2×2 matrix and its elements are given by the parameters of the clock

$$q_{11} = \tfrac{h_0}{2}\Delta T + 2h_{-1}\Delta T^2 + \tfrac{2}{3}\pi^2 h_{-2}\Delta^3 \qquad q_{12} = h_{-1}\Delta T + \pi^2 h_{-2}\Delta^2$$
$$q_{21} = q_{12} \qquad q_{22} = \tfrac{h_0}{2\Delta T} + 4h_{-1} + \tfrac{8}{3}\pi^2 h_{-2}\Delta.$$

For the used receiver (u-blox 2013) we assume a temperature compensated crystal clock, resulting in $h_0 = 2 \times 10^{-19}$, $h_{-1} = 7 \times 10^{-21}$ and $h_{-2} = 2 \times 10^{-20}$.

To denote the clock model (4.1.1) with the white, standard normal distributed driving noise and to reflecting the correlation, its noise coupling matrix must be specified accordingly. Correlated noise can be generated by mapping (scaling, rotating) uncorrelated noise. A scale and rotation matrix are determined by the eigenvectors, V, and eigenvalues, D, of the covariance matrix, thus the noise coupling matrix is given by $G_c = V\sqrt{D}Q_c$.

Stacking together the transition and the coupling matrices

$$F = \left(\begin{array}{c|c} F_p & 0_{6\times 2} \\ \hline F_c & 0_{2\times 6} \end{array}\right) \quad \text{and} \quad G = \left(\begin{array}{c} G_p \\ \hline 0_{2\times 1} \mid G_c \end{array}\right)$$

respectively, we can denote the complete process model by

$$\mathbf{x}_k = F\mathbf{x}_{k-1} + G\mathbf{v}_k. \tag{4.3}$$

The process noise of the complete model is simply $\mathbf{v}_k = (\mathbf{v}_{p,k}, \mathbf{v}_{c,k})^T$.

We can also describe the process model probabilistically. We have zero mean normal distributed noise that disturbes the state vector. The transition PDF describing the temporal evolution results in

$$\begin{aligned} p(\mathbf{x}_k \mid \mathbf{x}_{k-1}) &= p_{\mathbf{v}_k}(\mathbf{x}_k - F\mathbf{x}_{k-1}) \\ &= \frac{1}{(2\pi)^{n_x/2}(G^TQG)^{1/2}} \exp\Big(-\frac{1}{2}(\mathbf{x}_k - F\mathbf{x}_{k-1})^T(G^TQG)^{-1}(\mathbf{x}_k - F\mathbf{x}_{k-1})\Big) \end{aligned}$$

A few more motion models have been evaluated, such as the nearly constant velocity model (Bar-Shalom et al. 2001) and a model that models a person's motion randomly, including random velocity and heading (Widyawan et al. 2007); nonetheless the Langevin process based model fits best to our experiments and outperformed the other models. Before getting to the experiments we explain the measurement models that will be used in the particle filter correction step.

4.1.2. Measurement models

This subsection derives the probabilistic observation models. It is divided in two: The first part deals with the pseudorange model and the second part describes the signal strength model.

Pseudorange likelihood function

To derive the pseudorange likelihood function we commence with the sensor model, in accordance with the general observation model (2.9),

$$\varrho_k = h_{k,\varrho}(D\mathbf{x}_k) + w_{\eta,k}.$$

This model relates the state vector to the observations, where we apply a boolean matrix, $D = \left(\begin{array}{c|c|c} I_3 & 0_{3\times3} & 0_{3\times2} \\ \hline 0_{2\times3} & 0_{2\times3} & \begin{smallmatrix} 1 & 0 \\ 0 & 0 \end{smallmatrix} \end{array}\right)$ to feed only the needed elements of the state vector. These are the receiver's position, $\mathbf{p}_k$, and the offset of its clock to those of the satellites, $t_{r,k}$. We further assume that this model is time-invariant between two consecutive time instances.

An analytic model for a pseudorange was already presented in Background section 2.4. For the sake of convenience we restate the non-linear model that describes a pseudorange:

$$\varrho_k = ||\mathbf{p}_{s,k} - \mathbf{p}_k|| + c(t_{r,k} - t_{s,k}) + I_k + T_k + w_{\eta,k}.$$

This model is time-variant, because the receiver and satellites are moving and the properties of the space between satellite and receiver vary too. It consists of the geometric range between the satellite, $\mathbf{p}_{s,k}$, and the receiver, $\mathbf{p}_k$, the receiver, $t_{r,k}$, and the satellite clock offset, $t_{s,k}$ and the delay due to the ionosphere, I_k, and troposphere, T_k, and eventually other error terms. The experimental setup corrects the pseudoranges already for the code bias and the satellite clock offset with respect to system time, nonetheless some residual errors of the satellite clock offset may remain. As well the ionospheric and tropospheric delays are corrected, based on the 'Troposphere model' and the 'Broadcast ionosphere model' from (Takasu 2013, pp. 149). Hence, the remaining errors, which we collect in $w_{\eta,k}$, are caused by multipath propagation and shadowing effects, prediction errors of the ephemerides, relativistic effects, residuals of corrections terms and of course noise (Kaplan et al. 2006, sec. 7.2). In this work we use

$$\varrho_k = ||\mathbf{p}_{s,k} - \mathbf{p}_k|| + ct_{r,k} + w_{\eta,k} \tag{4.4}$$

to estimate a pseudorange.

It is usually considered that these remaining errors are distributed normally with zero mean, $w_\eta \sim \mathcal{N}(0, \sigma^2_{w_\eta})$. We introduce the pseudorange likelihood function for a single satellite

$$p(\varrho_k \mid \mathbf{x}_k) = p_{w_\eta}\big(\varrho_k - h_\varrho(D\mathbf{x}_k)\big).$$

Assuming statistical independence of pseudorange measurements from n_ϱ satel-

lites and denoting these pseudoranges $\boldsymbol{\varrho}_k = \{\varrho_k^j\}_{j=1}^{n_\varrho}$, the joint likelihood function of the pseudoranges is given by

$$\begin{aligned} p(\boldsymbol{\varrho}_k \mid \mathbf{x}_k) &= \prod_{j=1}^{n_\varrho} p_{w_\eta}\big(\varrho_k^j - h_\varrho^j(D\mathbf{x}_k)\big) \\ &= \prod_{j=1}^{n_\varrho} p_{w_\eta}(\varrho_k^j - \hat{\varrho}_k^j) \\ &= \prod_{j=1}^{n_\varrho} \frac{1}{\sqrt{2\pi}\hat{\sigma}_{w_\eta,k}^j} \exp\left(-\frac{1}{2}\frac{(\varrho_k^j - \hat{\varrho}_k^j)^2}{(\hat{\sigma}_{w_\eta,k}^j)^2}\right) \\ &\triangleq \mathcal{L}(\mathbf{p}_k \mid \boldsymbol{\varrho}_k). \end{aligned} \tag{4.5}$$

Pseudoranges estimated through (4.4) and its standard deviation are denoted with $\hat{\varrho}_k^j$ and $\hat{\sigma}_{w_\eta,k}^j$, respectively. To estimate the standard deviation RTKLIB's default values were used.

Rssı likelihood function

We mentioned before (sec. 2.5.2) that models that map space to signal strength, and vice versa, are too inaccurate for arbitrary areas. There is no agreement of the scientific community on a unifying model for that relationship. The common 'workaround' is the fingerprinting method – that is creating this missing model empirically. How one can obtain such a model on continuous space is explained in section 3.1 and herein used.

In the state space context RSSI observations take the general form

$$s_k = h_{k,s}(C\mathbf{x}_k) + w_{\epsilon,k},$$

where $h_{k,s}(\cdot)$ is a non-linear function. This model is also time-variant since the object/user may move and because of the changes of the environment that influence RSSI readings. These changes include primarily appearing, disappearing or moved access points and of larger objects. Such modifications in the fingerprint environment requires updates of the radio maps, however, the resulting temporal variations are prevalently slow which requires only infrequent updates. The

design of the fingerprinting system and the conditions of the environment determine the update rate. For the duration of a discretization interval, we consider this observation model time-invariant.

In this study the radio map and the wlan positioning system is defined on two-dimensional space of locations, $\mathbf{p} \in \mathbb{R}^2$. Therefore we use matrix $C = (I_2 | 0_{2\times 6})$. Nonetheless the extension to $\mathbb{R}^3$ is straight forward and for notational simplicity we denote a position by $\mathbf{p}$ regardless of its z-coordinate.

Based on the general model and relying again on the assumption of normal distributed noise, the likelihood function for the rssi for one access point can be denoted as

$$p(\bar{s}_k \mid \mathbf{x}_k) = p_{w_\epsilon}\big(\bar{s}_k - h_s(C\mathbf{x}_k)\big).$$

The notation $\bar{s}_k$ is used again to indicate the use of the mean instead of the collection of several rssi.

To incorporate n_l access points we multiply likelihood functions for each access point; and instead of relying only on fingerprints, as in section 2.5.2, we use interpolated rssis yielded by the Gaussian process regression. When the Gaussian process model is once established, predictions are obtained by equations (3.2) and (3.3), from which the likelihood function can be established as observations are received. Recall section 3.2, the likelihood function becomes

$$\begin{aligned} p(S_k \mid \mathbf{x}_k) &= \prod_{l=1}^{n_l} p_{w_\epsilon}\big(\bar{s}_k^l - h_s^l(C\mathbf{x}_k)\big) \\ &= \prod_{l=1}^{n_l} p_{w_\epsilon}\big(\bar{s}_k^l - \mu^l_{\mathbf{s}^*,k}\big) \\ &= \prod_{l=1}^{n_l} \frac{1}{\sqrt{2\pi}\sigma^l_{\mathbf{s}^*,k}} \exp\left(-\frac{1}{2}\frac{\big(\bar{s}_k^l - \mu^l_{\mathbf{s}^*,k}\big)^2}{(\sigma^l_{\mathbf{s}^*,k})^2}\right) \\ &\triangleq \mathcal{L}(\mathbf{p}_k \mid S_k). \end{aligned} \tag{4.6}$$

One difference to equation (3.4) exists though: The likelihood function within the update step of the filter (2.10b) needs to consider the object's/user's motion, thus, the change of the state vector over time. The test inputs need to be determined dynamically according to the system's state. The model accounts for that with the time index added to the parameters of the likelihood function.

Equation (4.5) and (4.6) are the key elements to fuse signal strength and pseudoranges. Both functions represent the measurements as function of space with the position as 'fixed' parameter. Varying the position provides values of likeliness that the measurement was received at the currently chosen position.

Based on these likelihoods the balance between GPS and WLAN is controlled. Within the particle filter the likelihood functions determine the weights of each particle, whose combination determines the smoothness of seamless localization. The integration of the likelihood functions into the particle filter is discussed next.

4.2. Particle filter – A WLAN aided PVT

The recursive Bayesian filter is conceptually described in equation (2.10). We discussed the two step procedure consisting of the process and measurement update. As both parts of the estimator involve integrals over non-trivial PDFs they need to be approximated.

In section 2.2.5 we described the fundamentals of the particle filter as numerical approximation to the sequential state estimation problem. Based on the state space models of the previous section we explain in this section the particular details of the particle filter. For simplicity we resort to the bootstrap filter. We commence with the process update followed by the measurement update, where the data fusion takes place.

4.2.1. Process update

The state transition probability describes the propagation of the state from the last time instance to the current one. We provided a probabilistic description based on the Langevin process that models this evolution. In addition, assume a particle approximation of the prior distribution of the last time step is known. We could sample from the state transition density and compute the particle approximation to equation (2.10a) to obtain the PDF that predicts the current state.

Instead, since an analytic description of the state transition exists, we pass each particle from the prior, $p(\mathbf{x}_{k-1}^{(i)} \mid \mathbf{y}_{1:k-1})$, through equation (4.3) to yield

$$\mathbf{x}_k^{(i)} \mid \mathbf{y}_{1:k-1} = F\mathbf{x}_{k-1}^{(i)} + G\mathbf{v}_{k-1}^{(i)} \tag{4.7}$$

the approximative prediction density, $p(\mathbf{x}_k \mid \mathbf{y}_{1:k-1})$. Where $\{\mathbf{v}_{k-1}^{(i)}\}_{i=1}^{N}$ are samples

from the process noise PDF. Simulating equation (4.3) is more efficient and intuitive than the way via the state transition PDF.

At the initialisation stage the prior PDF is usually not known. In such cases of high uncertainty a uniform prior, $p(\mathbf{x}_0) \triangleq p(\mathbf{x}_0 \mid \mathbf{y}_{-1})$, over a plausible region is a common choice.

4.2.2. Measurement update

The measurement update, or often called correction, follows formally equation (2.10b). To solve this equation the PDFs are approximated with particles and their weights, where the weights compensate for the deviations of the importance function from the posterior PDF. We approximate the filtering distribution with particles and their associated weights

$$p(\mathbf{x}_k \mid \mathbf{y}_{1:k}) \approx \sum_{i=1}^{N} \underbrace{\frac{1}{\sum_{j=1}^{N} \tilde{\omega}_k^{(j)}} \frac{p(\mathbf{x}_k^{(i)})}{q(\mathbf{x}_k^{(i)})}}_{\omega_k^{(i)}} \delta\big(\mathbf{x}_k - \mathbf{x}_k^{(i)}\big). \tag{4.8}$$

The particles that are distributed according to the predictive distribution $\mathbf{x}_k^{(i)} \sim p(\mathbf{x}_k^{(i)} \mid \mathbf{y}_{1:k-1})$ and the weights are initially uniform distributed. The key is to adapt the weights such that the predicted distribution is updated according to the obtained measurements.

The Bayesian bootstrap filter employs the state transition PDF as importance function. The weight update is given by the product of weights of the previous time instance and the likelihood function, see equation (2.24).

The likelihood functions are given by equation (4.5) and (4.6) and their particle approximations can be obtained directly by drawing samples from these two Gaussian distributions, whose mean and variance are known since they are deduced from measurements.

For the pseudorange likelihood function we obtain

$$\begin{aligned} p(\varrho_k \mid \mathbf{x}_k^{(i)}) &= \prod_{j=1}^{n_\varrho} p_{w_{\eta,k}}\big(\varrho_k^j - h_\varrho^j(D\mathbf{x}_k^{(i)})\big) \\ &= \prod_{j=1}^{n_\varrho} \frac{1}{\sqrt{2\pi}\hat{\sigma}^j_{w_\eta,k}} \exp\left(-\frac{1}{2}\frac{(\varrho_k^j - \hat{\varrho}_k^{j,(i)})^2}{(\hat{\sigma}^j_{w_\eta,k})^2}\right) \\ &= \mathcal{L}(\mathbf{p}_k^{(i)} \mid \varrho_k). \end{aligned}$$

To predict the pseudoranges, $\hat{\varrho}_k^{(i)}$, model (4.4) is calculated for each particle. The variance of each pseudorange is estimated by RTKLIB and equal for all particles.

Similarly, the approximative RSSI likelihood function becomes

$$\begin{aligned} p(S_k \mid \mathbf{x}_k^{(i)}) &= \prod_{l=1}^{n_l} p_{\bar{w}_{\epsilon,k}}\big(\bar{s}_k^l - h_s^l(C\mathbf{x}_k^{(i)})\big) \\ &= \prod_{l=1}^{n_l} \frac{1}{\sqrt{2\pi}\sigma^{l,(i)}_{\mathbf{s}^*,k}} \exp\left(-\frac{1}{2}\frac{\big(\bar{s}_k^l - \mu^{l,(i)}_{\mathbf{s}^*,k}\big)^2}{(\sigma^{l,(i)}_{\mathbf{s}^*,k})^2}\right) \\ &= \mathcal{L}(\mathbf{p}_k^{(i)} \mid S_k). \end{aligned} \tag{4.9}$$

The RSSI predictions, $\{\mu^{(i)}_{\mathbf{s}^*,k}\}_{i=1}^N$ are yielded by choosing the particle's position as test inputs, $P_k^* = \{C\mathbf{x}_k^{(i)}\}_{i=1}^N$ and employing (3.2).

The interaction of the true location with the observations and their variances in the likelihood functions modifies the impact of a particle. If the measurement coincides well with true location the impact of a particle increases, and the other way around for measurements far from the correct location. Likewise, if the variance is small particles receive more impact as if the variance of the likelihood function is large.

The next step is to combine the two measurement likelihood functions. The independence of RSSIs and pseudoranges allows us to multiply their likelihood functions in order to obtain a global likelihood function

$$p(\mathbf{y}_k \mid \mathbf{x}_k^{(i)}) = \mathcal{L}(\mathbf{p}_k^{(i)} \mid S_k) \cdot \mathcal{L}(\mathbf{p}_k^{(i)} \mid \varrho_k).$$

This part of the algorithm essentially fuses the observation data. The allocation of weights to the particles based on their positions and the observations fulfils an automatic shifting between GPS and WLAN.

By substituting the latter equation into (4.8) we can finally formulate the particle approximation to the posterior PDF:

$$p(\mathbf{x}_k \mid \mathbf{y}_{1:k}) \approx \sum_{i=1}^{N} \mathcal{L}(\mathbf{p}_k^{(i)} \mid S_k)\mathcal{L}(\mathbf{p}_k^{(i)} \mid \varrho_k)\omega_{k-1}^{(i)}\delta(\mathbf{x}_k - \mathbf{x}_k^{(i)}). \tag{4.10}$$

Likelihood functions provide information on the parameters only up to a multiplicative constant, therefore we normalise the updated weights so that they sum to unity. This ensures that the resulting posterior also sums to unity and thus yields a proper PDF that provides a valid prior PDF for the following recursion step.

Updating the weights with the multiplied likelihood functions assigns high weights to particles with locations that correspond best with both observations. During the resampling particles with high weights are multiplied whereas particles with low weight are forwarded or may even not survive.

Furthermore, they control the influence of GNSS and WLAN observations, because both systems have contrasting error mechanisms. GNSS performance degrades in areas with obstructed satellite to receiver link, where scattering and multipath effects delay the signal propagation time. The same obstructions cause the WLAN signal strength to distribute heterogeneously, and not smoothly, exponentially decaying as predicted by path loss models. The larger RSSI deviations between neighbouring locations facilitates fingerprinting. For example, variances of pseudoranges received indoors are generally higher that the pseudorange variances received in open sky scenarios. Consequently, indoors the weights determined by the pseudorange likelihood are lower than outdoors which decreases confidence in the GNSS measurements.

The other way around in open sky scenarios, where GNSS achieves high accuracies, the RSSI are too ambiguous. Small differences in signal space can result in very large deviations in space, making fingerprinting methods perform poor. However, the good GNSS signal reception conditions result in low pseudorange variances. That again shifts weight to the GNSS, likely improving the overall localisation accuracy.

The choice of the RSSI variance is derived from the fingerprint database and therefore not directly related to the observation made in the localisation phase. The weighting between the two sensor data is (unfortunately) merely governed by the variances of the pseudoranges.

4.3. Experiments

We explain the experiments, to assess the performance of the particle filter fusing GNSS pseudoranges and WLAN RSSI, in this section. They build upon the experiments for the WPS described in chapter 3. For information about the test area, recording and pre-processing of RSSIs we refer to section 3.3.

This section describes the extension of the data logging hard and software to store in addition to the RSSIs the GPS raw data, it explains the measurement process(es) and comprises some noteworthy details of the particle filter implementation.

Measurement capturing The WLAN data capturing and processing chain used to evaluate the WPS is described in section 3.3 and extended by multiple points:

Record GNSS data The used receiver is the ublox LEA-6T-0 (u-blox 2013) single frequency GPS receiver with raw pseudorange data output. It is connected via USB to a laptop. The antenna is a standard patch antenna that was delivered with the device. During the experiments the experimenter hold the laptop in front of him, while the antennta was mounted on his head, to avoid blocking because of the experimenter's body and the laptop. In addition to the motion of the experimenter's body also his head movements, such as pitch and yaw motions, influence the GPS antenna position.

The obtained raw GPS data are parsed and preprocessed. Preprocessing comprises calculation of the satellite positions and satellite clock at transmission time and estimation of correction data, such as ionospheric and tropospheric delays, and corresponding variances. Thereafter the data are associated with a timestamp. To parse the UBX protocol and preprocess the data we integrated functions from RTKLIB (Takasu 2013)[1]. Addition-

1. RTKLIB, program package for standard and precise positioning with GNSS: http://www.rtklib.com/

ally the GPS receiver NMEA output is read, parsed and associated with a timestamp too. The software library nmealib[2] is used to parse the NMEA messages. Either data are subsequently buffered and stored in a SQLite database. Reading from the buffer and writing to the database is executed in a separate thread.

The whole GPS data processing runs concurrently to the WLAN data processing, where GPS data processing runs on its own, without user intervention. Starting and stopping of threads is controlled by the already described graphical user interface.

Capture WLAN data continuously So far WLAN data are only recorded after the experimenter reached a reference position, clicked on the map application and started data capturing and storing for a few seconds. For the combined GPS and WLAN experiments we implemented an optional recording mode which allows to capture WLAN data in a temporal continuous fashion. We used PCAP's dump feature that stores WLAN packets as they arrive, without preprocessing, in a pcap-file. This mode fits better typical scenarios in which the object/user moves uninterrupted.

The drawback of this mode is that no reference points can be associated to the observations. However, both modes, the temporally discrete and continuous mode, may be used simultaneously.

Synchronisation of GPS and WLAN data Above we mentioned the association of timestamps to the observations. We add a Julian date timestamp after parsing of GPS and discrete WLAN data. When doing a continuous WLAN data capture a unix epoch timestamp is used.

Notes on dealing with geographic data Fusing the GPS and WLAN observations premises a common spatial reference system. GNSS receiver use overwhelmingly a system based on an ellipsoidal approximation of the earth's shape. The spatial reference system of most digital maps available on the internet (also OSM) is based on a spherical approximation of the earth's shape. This discrepancy – if left uncompensated – leads to differences in northing (larger differences at higher latitudes) while easting is unaffected. To account for this difference and

2. nmealib, software library to parse and generate NMEA sentences: https://github.com/AHR-Project/nmealib

to minimise potential inaccuracies due to coordinate conversions, we employ the proj.4 library[3] for all transformations and projections between geographic coordinates.

Transformations from geographic Cartesian coordinates (earth-centred, earth-fixed) to local coordinates are done in GNU Octave. The databases used to store GPS and WLAN data contains the unaltered geographic coordinate data as provided by the GPS receiver and OSM. The algorithm eventually processes local East, North and Up coordinates.

Furthermore, the OSM data itself might be erroneous, in particular because it is generated by its users. However, this allowed us to correct and update important landmarks of the map data of the test bed. It was done with help of a geodetic GNSS receiver in relative positioning mode, claimed to be accurate up to a few centimetre.

To obtain the trajectory data (RSSIs and associated MAC addresses) and the ground truth positions the experimenter used the software to record the radio map. (The same approach was used as well to generate the radio map, see section 3.3.) Ground truth data was generated by marking the equipment's position on the map. The approach of marking locations on the map introduces further location inaccuracies: The experimenter has to, first, visually estimate his own location (by identifying nearby landmarks); then second, find this position correctly in the map; and third, mark it on the map using a pointer device. To facilitate the last two tasks, we set up a OSM tile server cloning parts of the OSM data, configured it to render two more additional zoom-levels and enabled the JMapViewer software to depict these additional zoom-levels.

All the mentioned error sources are unmodelled and affect the outcomes of the particle filter. However, we believe that particularly the human introduced uncertainties are partially averaged out and that in average the error of each ground truth position is less than 0.5 metre.

Filter implementation details Assessing the filter performance revealed several issues in the design presented so far. In particular during the initial convergence of the filter, when the first estimates are still far from the true location. Furthermore, the task of self-localisation and the filter design constitute some issues that need attention. We could overcome most of the drawbacks by modifying the likelihood

3. proj.4, standard library for cartographic conversions: http://proj4.org/

functions.

The adaptions of the pseudorange likelihood function are discussed as follows:

1. The order of magnitude of pseudorange residuals can be about 10^6 m. Such magnitudes as arguments to the exponential kernel of a normal distribution exceeds the numerical precision and results virtually in zero. Therefore

2. In this initialisation phase of the filter the pseudorange likelihood function often yields only one single non-zero value. This concentrates the particles rapidly in a very small area from where they may never leave again. In such cases we set the likelihood function invalid and proceed with RSSI observations and subsequently with the next recursion step. If no RSSI measurement is obtained, it follows prediction, which each time increases the variance, spreading the particles farther. If the particles get into the area of the true location the pseudorange likelihood possibly becomes more informative again, the RSSI observations suit the predicted location and the algorithm continues as described above. A quite slow convergence rate is the consequence, because while only predicting repeatedly the particles obey a random process.

3. Additionally, as a result of multiplying several pseudorange likelihoods, one for each satellite, the joint pseudorange likelihood is often very peaked. To mitigate this effect we smooth the joint likelihood function as in (Ferris et al. 2006); it is given by the geometric mean of the individual likelihoods:

$$\mathcal{L}(\mathbf{p}_k \mid \varrho_k) = \Big(\prod_{j=1}^{n_\varrho} p(\varrho_k^j \mid \mathbf{x}_k^{(i)})\Big)^{1/n_\varrho}.$$

4. The last issue regarding the pseudorange likelihood we need to discuss is the estimation of the receiver clock offset. The pseudorange likelihood function does discriminates poorly between the position and the receiver clock offset. As Hejc, Seitz and Vaupel (2014), we also use a least squares method to estimate the clock offset

The presented RSSI likelihood function needs only a modification to avoid the case when the set of observations contains no RSSI which is also contained in

the radio map. In this situation we invalidate the likelihood function as if no measurement was received.

An other issue, related to WLAN observations, occurs if the object/user moved beyond the radio map area, but WLAN signals are still received. This constitutes a problem because of possibly ambiguous RSSI in the radio map which may be similar as the ones received outside the radio map area. Knowing that the estimate will be inaccurate, we compute the RSSI likelihood function only if the distance to the nearest reference position is smaller than 10 m, thus bounding the error. This value must of course be balanced between the average distance of fingerprint positions and the desired error bound.

In commercial receivers GPS observations are usually postprocessed to improve the positioning solution. Common provisions are the Hatch filter to smooth pseudoranges, advanced measurement integrity methods to detect faulty observations and multipath detection methods to eventually mitigate errors due to multipath propagation effects. Our implementation – beside what is already implemented in the receiver's before delivering pseudoranges – does not rely on such methods to show the potential of the fusion strategy. Nevertheless, we are aware that the used GPS receiver is of decent quality, most likely comprising the mentioned features and providing relatively low noise measurements. We included the position estimates of the receiver as a benchmark where it is appropriate.

The correction step of the particle filter (4.10) employs both measurements simultaneously, but not always both observation are received within in the same discretization interval. Two measurement sources yield four cases: 1) none measurement was received, 2) both measurements were observed, 3) only WLAN RSSI were received and, last but not least, 4) only pseudoranges were obtained. According to the availability of measurements the likelihood functions are calculated and the particle filter corrects the predicted PDF. In the first case the correction is skipped and the filter proceeds with the prediction.

[May be include pseudocode (listing) of complete pf. Include all the above implementation details? Or may be in the filter section.]

4.3.1. Temporally discrete trajectory

We continue with the description of the experiments to evaluate the proposed algorithm. We have chosen two different trajectries: First, Trajectory-3, consisting of mostly indoor sections, and, Trajectory-4, a path consisting of mostly outdoor

sections. Both trajectories take course along the semi-open passageways of both buildings of the test bed. We use the discrete WLAN measurement mode in order to be able to generate a reference trajectory; GPS measurements are obtained continuously.

The ground truth of Trajectory-3 and Trajectory-4 are depicted in figure 4.1[4].

Figure 4.1.: Ground truth trajectory within the test area (Trajectory-3 —•—Trajectory-4 —•—). The trajectory starts at the circle and finished in the lower left part of the image. © OpenStreetMap contributors

Trajectory-3 begins in the left side of the left building, proceeds to the building's access where it leaves on the roofed passageway. At the corner with the right building it follows its passageway almost until the end of the building where it

4. These paths consists of the actual ground truth measurement data. While the experimenter followed relatively accurate the existing ways (when they were part of the trajectory); light-red dashed lines in the OSM map, the deviations from these lines are essentially introduced by the issues when dealing with geographic data and difficulties in creating the ground truth data.

enters a room, then leaves that room to proceed for 18 m outdoors on the lawn area.

Trajectory-4 starts at a open space, turns sharp right, passes below a few trees and then takes a sharp left turn towards the right building. The next section of Trajectory-4 is largely along the roofed passageway of the right building, with a short deviation inside the right building from where it proceeds to the northernmost corner of the trajectory. After entering the semi-open passageway of the left building Trajectory-4 deviates north west to go on around the small lawn area — as on the roofed passageways, as well in this part of the path GPS LOS condition is unlikely, attributable to trees. Then the trajectory takes the passageway of the left building back to the right building, continues on its passageway until it turns right in the middle of the right building where it continues to the large open space where it ends.

Most sections of both trajectories run through a multipath rich environment, additionally, while on the semi-open passageways and around the small lawn area most parts of the sky view is blocked. Only on the open space area, where Trajectory-4 starts and ends a relatively open view to the sky is available; relatively because about 30 m south-west of the starting point a four storey building is located and parts of the open space are covered with trees.

In case of Trajectory-4, the experimenter stayed initially static at the start point until the first raw GNSS observations were obtained. A meaningful initial position estimates is thereby ensured, because this trajectory begins outdoors in an area were WLAN fingerprinting is expected to perform poorly. The initial delay until the first raw GPS observations arrived is usually 20 s to 40 s. While recording the trajectory data the experimenter moved from reference position to reference position with approximately constant speed. The WLAN package capture time at each reference position was again 2 s to 3 s.

In each cycle of the particle filter the EAP is estimated according to equation (2.3). These estimates and the ground truth are used to calculate RMSE, but since the number of ground truth positions and the number of position estimates differs we interpolate the ground truth data linearly (relying on the assumption of constant speed) between the ground truth positions.

Results on the accuracy of the particle filter are assessed by means of the RMSE, (3.5). Due to the randomness of the algorithm the presented results are averages over 50 repetitions.

A reader comparing the results of chapter 3 with the subsequently presented

results may notice that the coordinates suggest a shift of the test bed, this is caused simply by a different reference coordinate used for the transformation to local coordinates.

4.3.2. Temporally continuous trajectory

In this dynamic scenario the experimenter moves with roughly constant speed along Trajectory-4, see figure 4.1. We will refer to it by Trajectory-4a. The GPS data and WLAN data are recorded from the beginning to the end, continuously without breaks.

The first measurements not containing GPS raw data are removed as before.

The estimates of the state vector are also obtained in each filter cycle by the EAP from the set of particles.

4.4. Results

In this section we presents the evaluation of the localisation performance, exemplified with two trajectories: Trajectory-3 and Trajectory-4. Trajectory-3 begins indoors and contains more indoor sections and a transition from indoor area to an outdoor area with a relatively open sky view. Whereas Trajectory-4 begins outdoors in relatively open space, contains a short indoor section and terminates close by the starting point with good satellite visibility. We first analyse the performance of the proposed particle filter for both trajectories with the temporally discrete recorded trajectory, because it allows us to compute reasonable error measures and to pinpoint its strength and weakness. We compare the filter's accuracy and precision for three cases: the hybrid solution using GPS and WLAN sensor data (GPS + WLAN), the GPS-only solution using solely GPS observations and the WLAN-only solution that employs only observations from the WLAN.

The stop-and-go scenario when the trajectory is recorded temporally discrete is probably not the most realistic one, therefore we show also results of the particle filter for a path as in Trajectory-4, but recorded continuously. Unfortunately, a detailed evaluation is not possible in that case. We present it nevertheless to demonstrate the filter's practicability.

To produce reliable results and to compensate the random variations of a Monte Carlo method we repeat each simulation 50 times. The following results are averages over these 50 runs.

4.4.1. Indoor-like trajectory – Trajectory-3

Figure 4.2 depicts the trajectory and the ground truth data. The position estimates

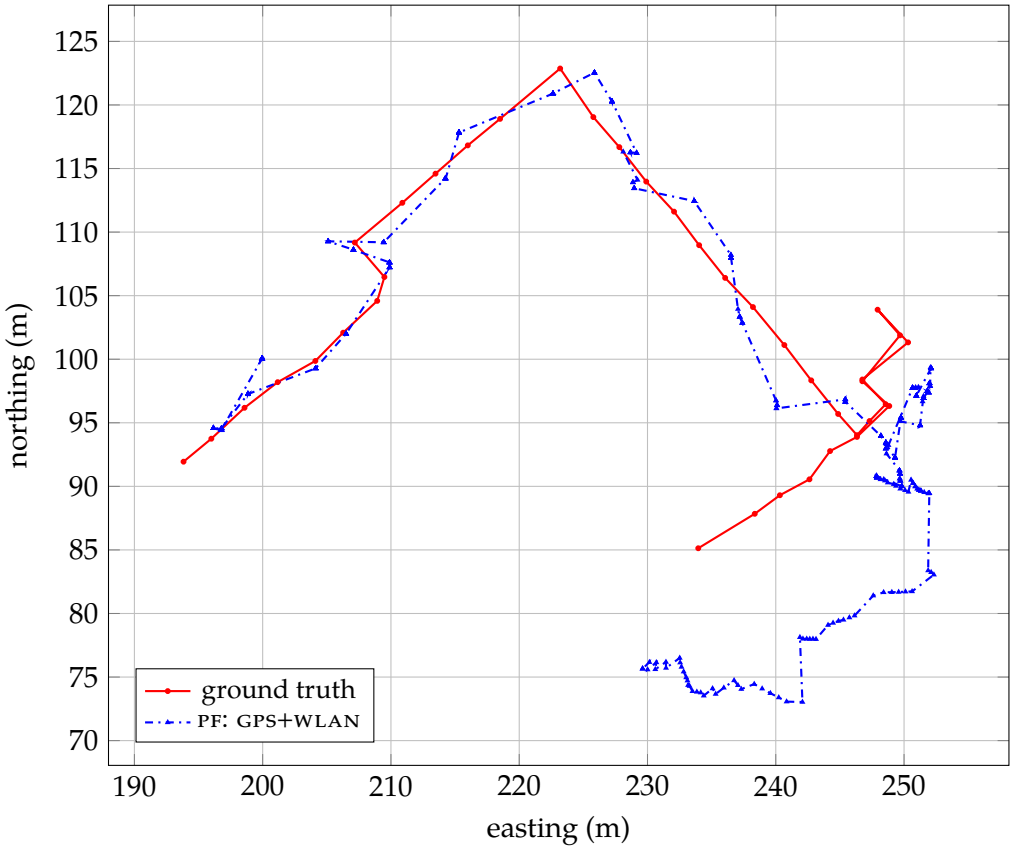


Figure 4.2.: Particle filter estimates and ground truth of Trajectory-3. The particle filter employed both measurements (GPS+WLAN).

are relatively sparse, because RSSIs are only obtained every few seconds. In the intermediate intervals the particle filter only runs the prediction step and when a new RSSI measurement is observed the position jumps to the new estimate.

Initially no observation is available and the estimated position results as the centroid of the initial particle distribution ($[120, 280]\,\mathrm{m} \times [50, 150]\,\mathrm{m}$). After WLAN observations are received the estimated trajectory coincides well with the ground truth, both indoors and while on the roofed passageway. Towards the

end of Trajectory-3 the trajectory proceeds indoors again, but the estimates show an offset of about 3 m. When leaving the building towards the open space, GPS measurements are received the first time. Because of the poor quality of these observations, the particle filter estimates the position inaccurately. However, the WLAN observations are apparently worse than the GPS observations, so that the filter trusts the GPS observations more than the RSSIs and estimated trajectory deviates further from the ground truth. A consequence of this behaviour is the high overall RMS error, shown in the first column of table 4.1.

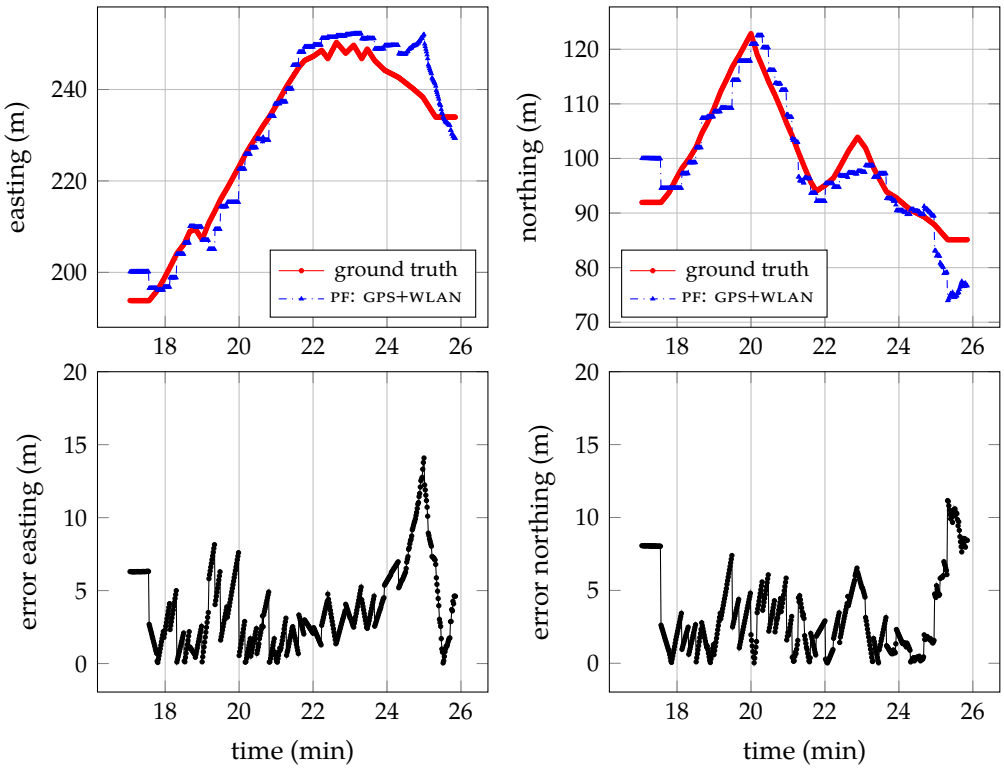
Figure 4.3 shows the estimated trajectory and the absolute value of the error. Errors larger than 7 m are observable at the beginning of the trajectory, about 19 s where the left building is left and when the right building is entered (22 s to 24 s) . Thus, larger inaccuracies occur during and after the transitions between indoors and outdoors. However, also the changes in direction that occur during these transitions may contribute to the error. The by far largest error can be observed at the end of the trajectory, where the measurements of both sensors are of low quality. In this section of the path the error grows above 15 m.

It is worth mentioning that if the first and the last section of the trajectory are neglected the error reduces to magnitudes as reported for the WPS in chapter 3.

Figure 4.4, 4.5 and table 4.1 compare the results of the particle filter using with both measurements, or only GPS- or only WLAN measurements.

The magnitude error of these setups is depicted in figure 4.4. The majority of time the error of WLAN-only is par with the error when GPS and WLAN are used. This is clear, because only during the last two minutes of the trajectory GPS signals were received and therefore no weighting between the both sensors takes place. During the last seconds when the path takes course outdoors and GPS measurements are available, the algorithm considers both measurements. It weights the GPS data much more than the WLAN data; which is reasonable, as the accuracy of the WLAN-only solution in this section of the trajectory is worse than that of GPS-only solution. The error when only GPS observations are used is largest, because the filter can only predict the position. While no measurement is received the position estimate based on solely the prediction is essentially static and the rising and falling of the error is determined by the trajectory itself. At the end of Trajectory-3 the error decreases, although the observations are not very accurate. It decreases even below the error of the WLAN-only solution, because outdoors the RSSIs suffer ambiguities that cause large errors.

The empirical cumulative distribution function is shown in figure 4.5. One



(a) East component of estimated trajectory and of the absolute value of the position error.

(b) North component of estimated trajectory and of the absolute value of the position error.

Figure 4.3.: Trajectory-3 estimated by the particle filter in easting and northing with ground truth and the resulting position error. Both, GPS and WLAN measurements are fed to the filter.

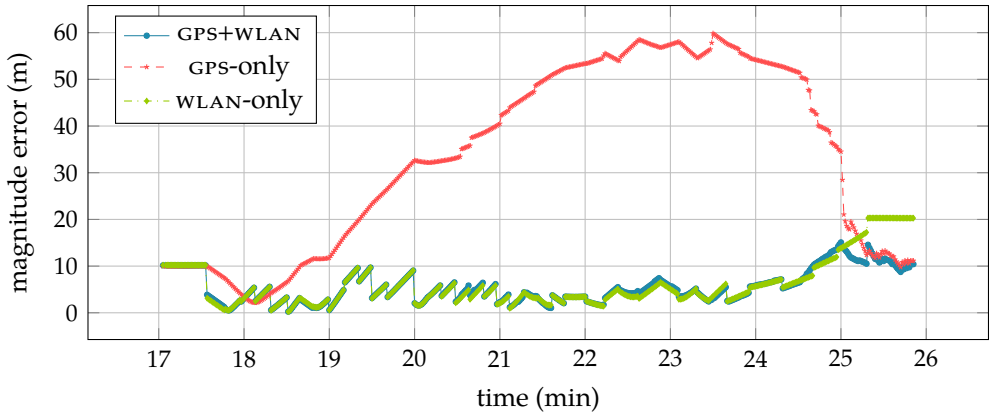


Figure 4.4.: Error of GPS+WLAN, GPS-only and WLAN-only solutions.

can see that the hybrid and the WLAN-only solution have similar distributions, whereas the distribution of the hybrid solution is less fat. The distribution of the GPS-only solution is much wider with very heavy tails. The GPS+WLAN and the WLAN-only solution yield a median error of less than 5 m. At 95 % the error of the GPS+WLAN solution is about 12 m, but for the WLAN-only solution it is above 20 m. These large errors of the WLAN-only solution is caused by the ambiguity issue, mentioned above. The heavier tail of the WLAN-only solution compared with the hybrid solution indicates that the precision of the hybrid solution is smaller. Since no pseudoranges are received for most of the time, the median error of the GPS-only solution is about 35 m and the error at 95 % is about 58 m.

Table 4.1 compares the RMSE and the standard deviation (SD) of the particle cloud, σ_p, averaged over the whole trajectory. These averages are little above the median

Table 4.1.: RMS error and SD of Trajectory-3 using GPS+WLAN, GPS-only and WLAN-only.

	GPS+WLAN	GPS-only	WLAN-only
RMSE (m)	6.21	37.33	7.80
avg. SD (m)	5.81	48.27	6.02

errors, an expected outcome. The particle filter that employs pseuodranges and RSSIs yields the lowest error and standard deviation. The WLAN-only solution's

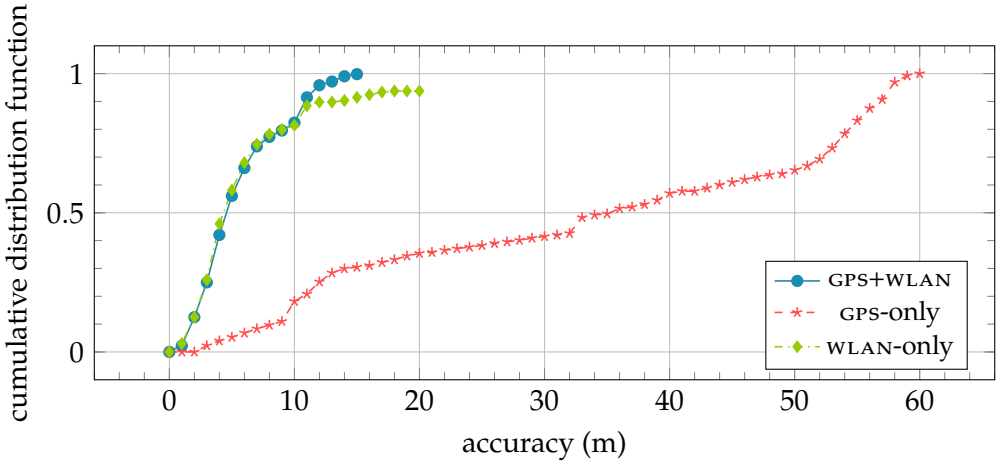


Figure 4.5.: Empirical error distribution function of the GPS+WLAN, GPS-only and WLAN-only solutions for Trajectory-3.

error and standard deviation are very close to those of the hybrid solution. Because of the lack of pseudoranges for the major part of the trajectory the hybrid filter uses principally WLAN data. For this trajectory we conclude that the hybrid solution outperforms the other two solution in accuracy and precision.

4.4.2. Outdoor-like trajectory – Trajectory-4

Temporally discrete trajectory

Figure 4.6 shows the estimates of Trajectory-4 of the proposed algorithm and gives an impression of the overall performance. The ground truth data is the one already shown in figure 4.1. The particle filter solution is first of all noisy and contains jumps that seem unrealistic. However, considering the process of trajectory recording explains this behaviour. Pseudoranges are received continuously, also during the static phases of RSSI capturing. In these intervals the filter continuous to predict and correct based on the GPS measurements, but the standstill is not estimated well. Suddenly observations from possibly both sensors indicate that the equipment is already at a location farther ahead and the filter adjusts to them, which results in position jumps.

In general, the filter outcomes follow the trajectory. Smaller deviations can be

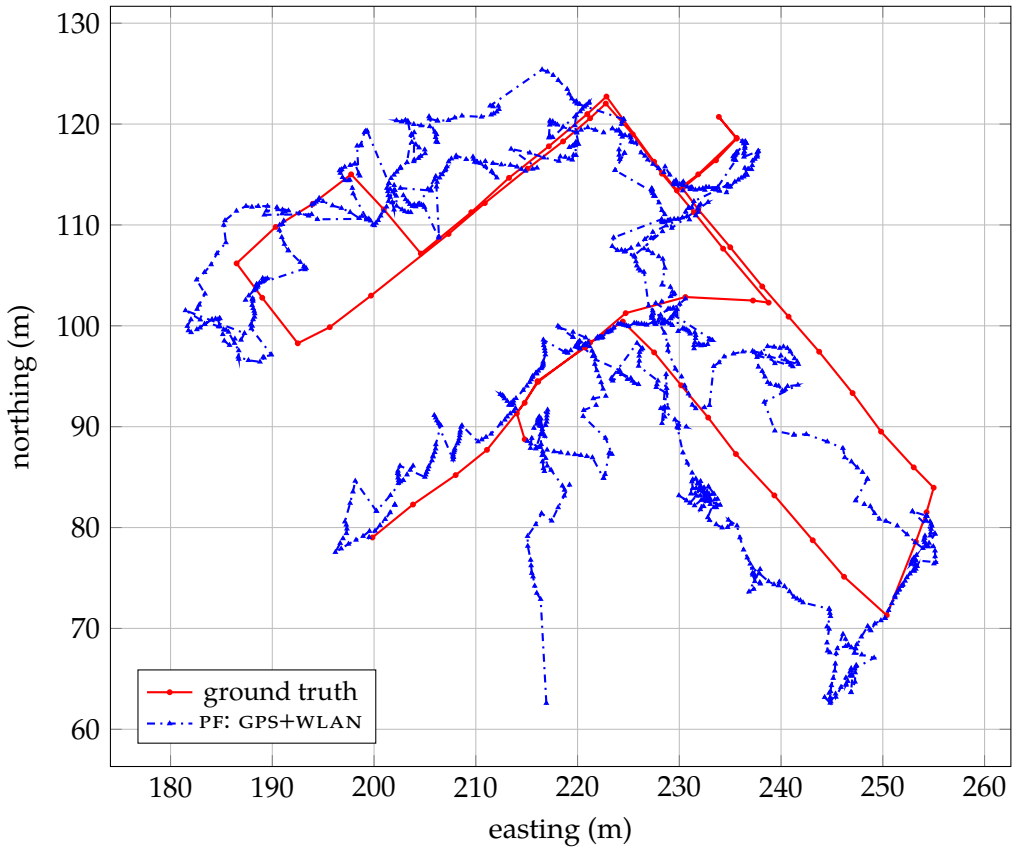


Figure 4.6.: Particle filter estimates and ground truth of Trajectory-4. The particle filter employed both measurements (GPS+WLAN).

seen in the large open area and along the semi-open passageways. Larger errors can be found along the right building about (240, 100) m, around the small lawn area about [185, 205] m × [98, 115] m and on the way back to the northernmost corner at about (225, 123) m where it gets back on track. The performance along the passageway of the right building and around the small lawn area can be explained by the poor WPS accuracy while the GPS also delivers uncertain estimates. As discussed in section 3.4.2, the ambiguous RSSI observations cause the poor WPS accuracy. The trajectory section on the way back through the passageway

of the left building is not covered by experiments of the WPS, but the decreased accuracy origins most likely from the same effect. Also the indoor section helps the algorithm to get back on track, due to more distinctive RSSI measurements. The indoor sections were also well estimated by the WPS.

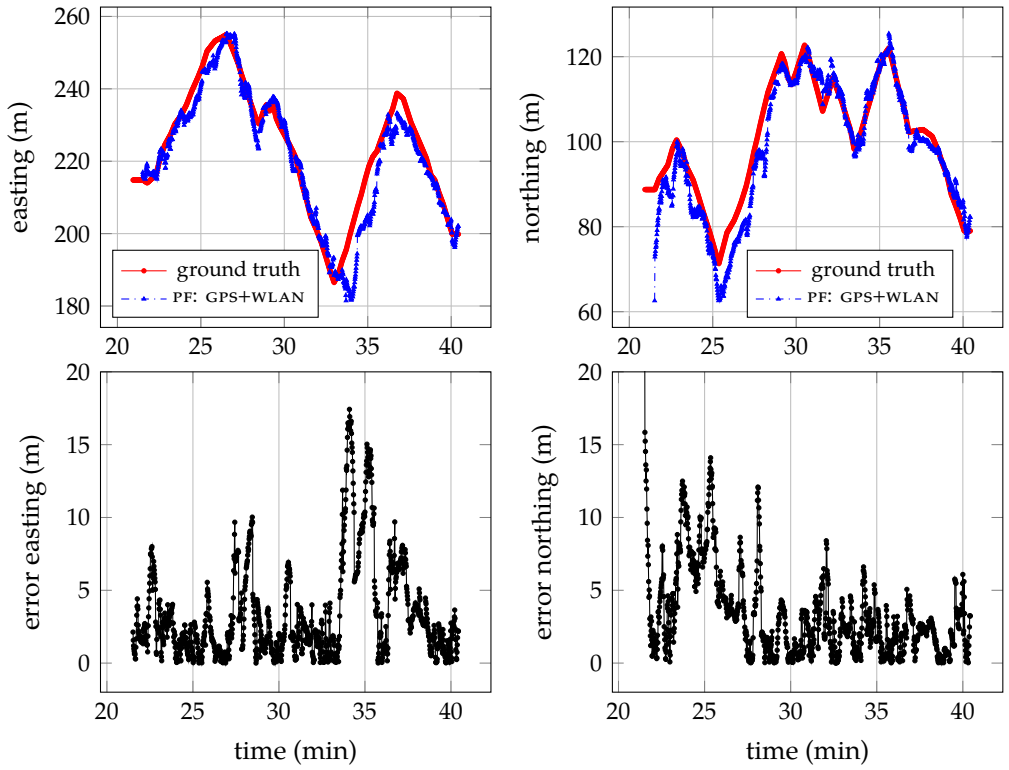
It is not surprising that the localisation error grows in particular where the GPS signal reception is poor – almost everywhere except in the open space area where the trajectory begins and ends – and where the WLAN data are less accurate, or ambiguous: in the middle division of the passageway of the left building and around the small lawn area (see results for Trajectory-1 in section 3.4.2).

Although, the initial estimates (from a uniform distribution on the test area) are more than 20 m away from the true initial location, the filter converges after some time as reliable measurements are obtained. These initial errors are evidently the largest errors and contribute significantly to the error statistics.

A more detailed analysis allows figure 4.7, showing the estimated trajectory over time. Ground truth data, linearly interpolated between every actual ground truth position, is depicted as reference. Below we show their difference over time, the positioning error. One can draw the same qualitative conclusion of the estimates as for the precedent figure. The estimates coincide largely with the ground truth, the position jumps can be seen over the whole trajectory. Errors about 5 m and smaller are most of the time observable, nevertheless two larger peaks, of about 10 m and more, at 34 min to 35 min and 37 min in East and at 28 min in North direction can be observed; they belong to the middle section along the passageway of the right building and to the parts of the passageway of the left building.

Figure 4.8 compares the position accuracy of the filter, using either GPS *and* WLAN, or only GPS, or only WLAN sensor data. The WLAN stand-alone solution present the largest errors, but for some time periods its errors are also the lowest. The error of the GPS-only solution appears smaller than the outcomes based on only WLAN, and equal or larger than the estimates derived from the hybrid solution, GPS+WLAN. The accuracy of the filter using data of both sensors outperform in average the single sensor solutions. In large parts of the trajectory it is in between the accuracies of the GPS-only or WLAN-only solution; during a few short periods the accuracy is even the highest, as about 27 min and 31.5 min.

The next figure (fig. 4.9) shows the empirical CDF of the position error for the hybrid, the GPS stand-alone and WLAN stand-alone solution. It becomes unequivocally that the GPS+WLAN solution outperforms the single sensor solutions.



(a) East component of estimated trajectory and the absolute value of the position error.

(b) North component of estimated trajectory and the absolute value of the position error.

Figure 4.7.: Trajectory-4 estimated by the particle filter in easting and northing with ground truth and the resulting position error. GPS and WLAN measurements are employed.

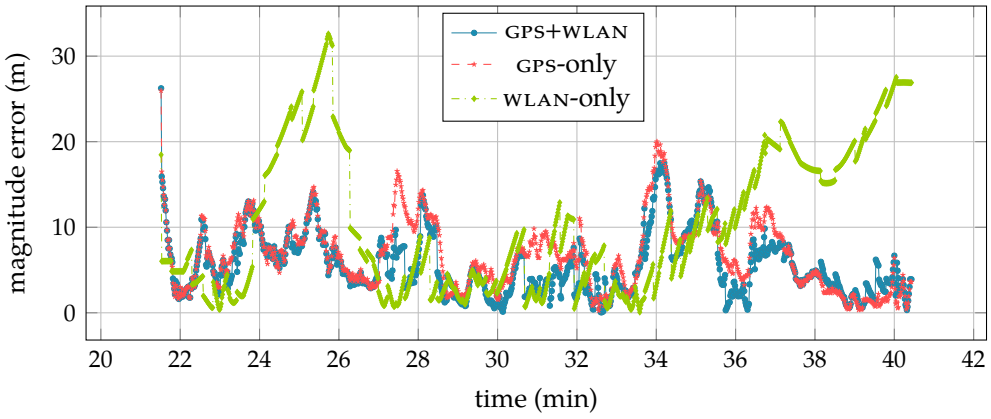


Figure 4.8.: Error of GPS+WLAN, GPS-only and WLAN-only solutions.

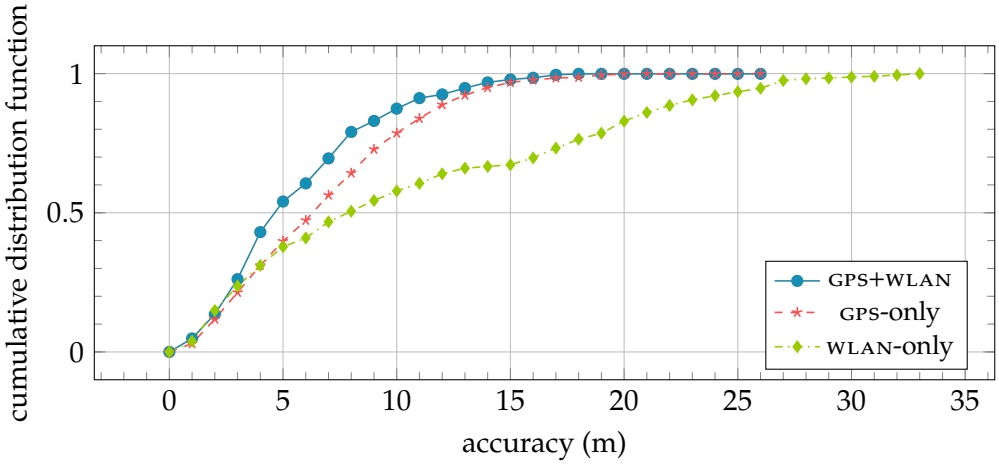


Figure 4.9.: Empirical error distribution function of the GPS+WLAN, GPS-only and WLAN-only solutions for Trajectory-4.

Its outcomes are not only more accurate, but as well more precise, in particular when large errors are considered. The error of the gps+wlan solution is almost bounded by the error of one of the other two solutions. For small percentiles it is bounded by the error of the wlan-only solution and for large percentiles the error is bounded by the gps-only solution.

For the combined solution, the median accuracy is about 4.2 m, where the median of the gps-only solution is about 2 m larger and the median of the wlan-only solution is only about 1 m larger. At 90 %, the estimates have 9.5 m accuracy when gps+wlan are used, 13.3 m accuracy when only gps is used and 18.4 m accuracy when only wlan is used.

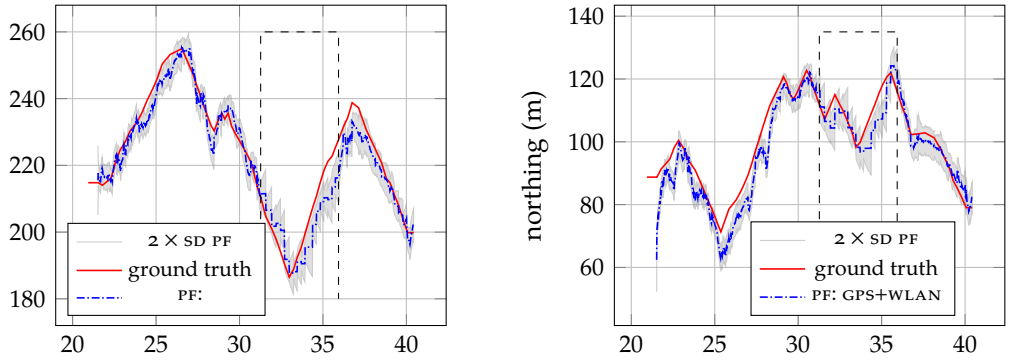
The summary in terms of the rmse and sd over the complete trajectory confirms this analysis. Table 4.2 compares the rmse and the sd of the particle cloud of the same three cases. The particle filter, employing gps and wlan data, has higher

Table 4.2.: Rms error and sd of Trajectory-4 using gps+wlan, gps-only and wlan-only.

	gps+wlan	gps-only	wlan-only
rmse (m)	6.68	7.88	13.36
avg. sd (m)	2.70	2.99	4.81

accuracy and precision than the particle filter using only one of the sensory data. The accuracy is improves 1.2 m compared with the gps-only solution, and 13 m compared with wlan-only solution.

Gps outages The following results show the localisation performance during gps outages. The outage was simulated in the interval 31.3 min to 35.9 min by simply removing all gps observations. In figure 4.10 the components in northing and easting of the estimated trajectory are shown. The gps outage is indicated by the vertical dashed lines. During the outage, the trajectory is still well estimated because of the wlan measurements. Between two consecutive wlan observations the filter only predicts the filtering density. Therefore the position stays almost constant, but the variance increases until the arrival of the next wlan measurement. The increase of variance is depicted by the grey area. Clearly, the accuracy during the outage depends on the quality of the wlan measurements; in the shown case the accuracy even improves as the outage interval was chosen



(a) East component of estimated trajectory and ground truth.

(b) North component of estimated trajectory and ground truth.

Figure 4.10.: Trajectory-4 estimated by the particle filter in easting and northing with ground truth and two times the standard deviation of the particle cloud. During the indicated interval GPS outages are introduced.

during poor GPS signal reception conditions. The filter can deal with abrupt GPS signal loss, though, the quality of the estimates during an outage depends on the WLAN data.

Comparison with u-blox GPS stand-alone solution We want to make use of the additional reference data we recorded: the positions estimates of the u-blox GPS receiver. Therefore we compare the filter outcomes from the hybrid solution with the estimates from the u-blox GPS receiver. Figure 4.11 presents the estimates for Trajectory-3. The estimates from the u-blox receiver are much smoother than that of the hybrid GPS+WLAN particle filter solution, the stop-and-go motion is not recognisable any more. It performs quite well, it follows the ground truth in very large sections through the large open space and also on the passageway of the right building – but only on the forward run. In continuation the estimates become worse and the estimated trajectory proceeds about 10 m parallel of the ground truth, until the small lawn area, where the estimates improve again. From the northern corner (225, 123) m on the errors increase again, the estimates pass through the right building. When moving to the open space the estimated trajectory coincides again well with the ground truth trajectory.

It is, first of all, remarkable that over the whole data recording period the

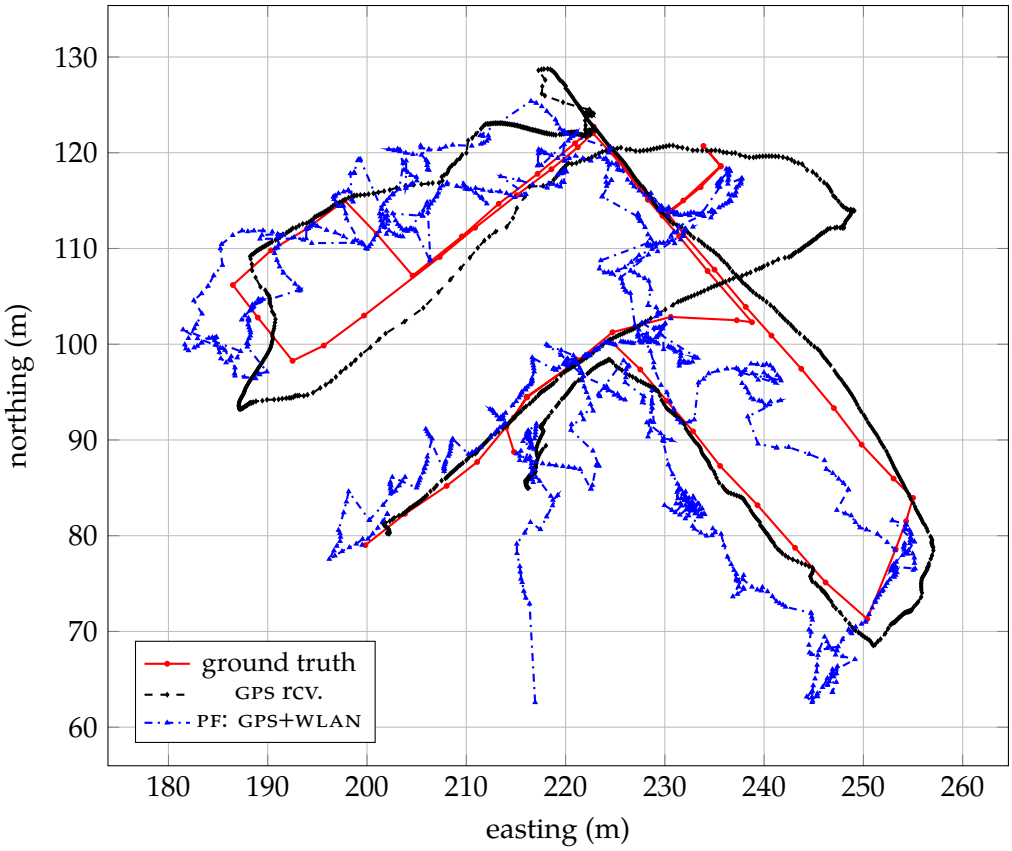
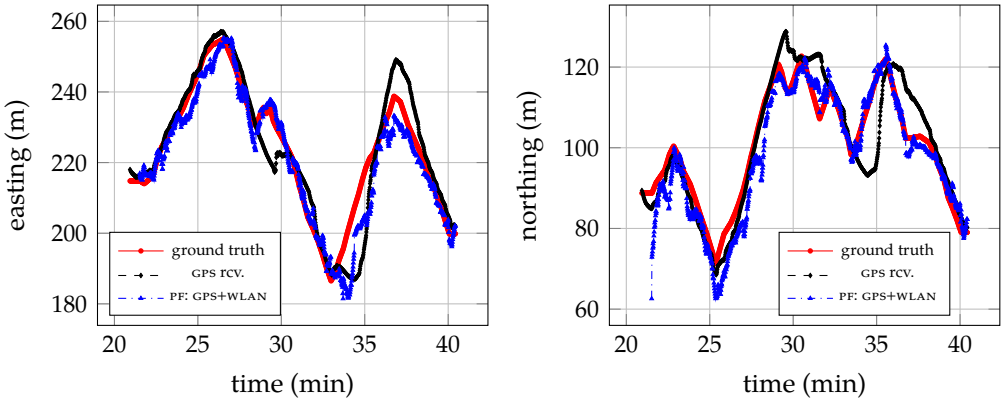


Figure 4.11.: Particle filter- GPS receiver estimates and ground truth of Trajectory-4. The particle filter employed both measurements (GPS+WLAN).

receiver receives signals of four or more satellites. Noteworthy is as well the capability to smooth the trajectory. In the stop-and-go scenario, this behaviour may on one hand improve the user experience, but on the other hand it falsifies the data.

An advantage of the hybrid solution is clearly that due to WLAN data locations within buildings are estimated more reliable and the system distinguishes better between indoors and outdoors. Therefore, in contrast to the u-blox GPS receiver stand-alone estimates, the estimates by the particle filter do not cross buildings.

The estimates in easting and northing separately, show that the particle filter hybrid solution follows more truthfully the ground truth trajectory. It presents less sections which are for a considerable time more than 10 m off. In the interval where the GPS+WLAN solution presents its largest deviations, 34 min to 35 min, it appears that it trusts the GPS data too much.



(a) East component of estimated trajectory. **(b)** North component of estimated trajectory.

Figure 4.12.: Trajectory-4 estimated by the particle filter and by the GPS receiver in easting and northing with ground truth and the resulting position error. GPS and WLAN measurements are employed.

Table 4.3 compares the RMS error of the hybrid- with the GPS stand-alone solution. The combination of pseudoranges with WLAN RSSI improves the localisation accuracy by factor 1.5.

Figure 4.13 compares again the error of the hybrid- with the GPS stand-alone solution, this time over time. In particular during the first four minutes, while the trajectory is in open space, the GPS stand-alone solution outperforms the particle filter. Afterwards the particle filter provides more accurate estimates, except around minute 28. One can also observe four major error peaks of the GPS receiver; and that while the error is increasing the HDOP is varying relatively fast. This indicates that the poor accuracy of the GPS receiver in these sections is caused by appearing and disappearing satellites that in turn are caused by a fast changing sky view, attributed to obstructions of the satellite links. The average HDOP during this trajectory is 4, whereas in good conditions the HDOP is between 1 and 2. As well the HDOP values about 20 at minute 35 confirms the harsh GPS

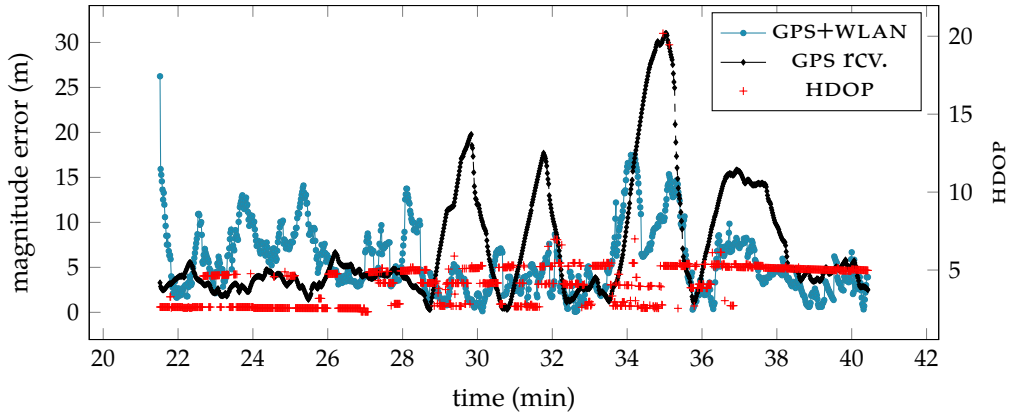


Figure 4.13.: Magnitude error of GPS receiver solution compared with hybrid particle filter solution. Both are related to HDOP of the GPS receiver solution.

RMSE GPS rcv. (m)	RMSE PF (m)
9.79	6.68

Table 4.3.: RMSE of GPS receiver and particle filter using GPS and WLAN sensor data.

signal reception conditions. (For epochs at which less than three satellites were in view the least squares solution is underdetermined and the HDOP is complex or yields magnitudes about 10^8. Therefore we removed the HDOP values of the corresponding epochs.)

Temporally continuous trajectory

The following results are derived from data recorded during continuous motion of the experimenter. Albeit the experimenter moved along the same trajectory, no ground truth data are available, because the timestamps of the measurements are uncorrelated. However, to put the results into context, we compare them with the linearly interpolated ground truth of the discrete data recording, used in the precedent section.

Figure 4.14 depicts the estimates from the hybrid particle filter. Since the RSSI measurements were obtained continuously, we average them every 4 s before

feeding them to the algorithm. Four seconds were chosen experimentally and provide good results. The estimated trajectory follows the ground truth well.

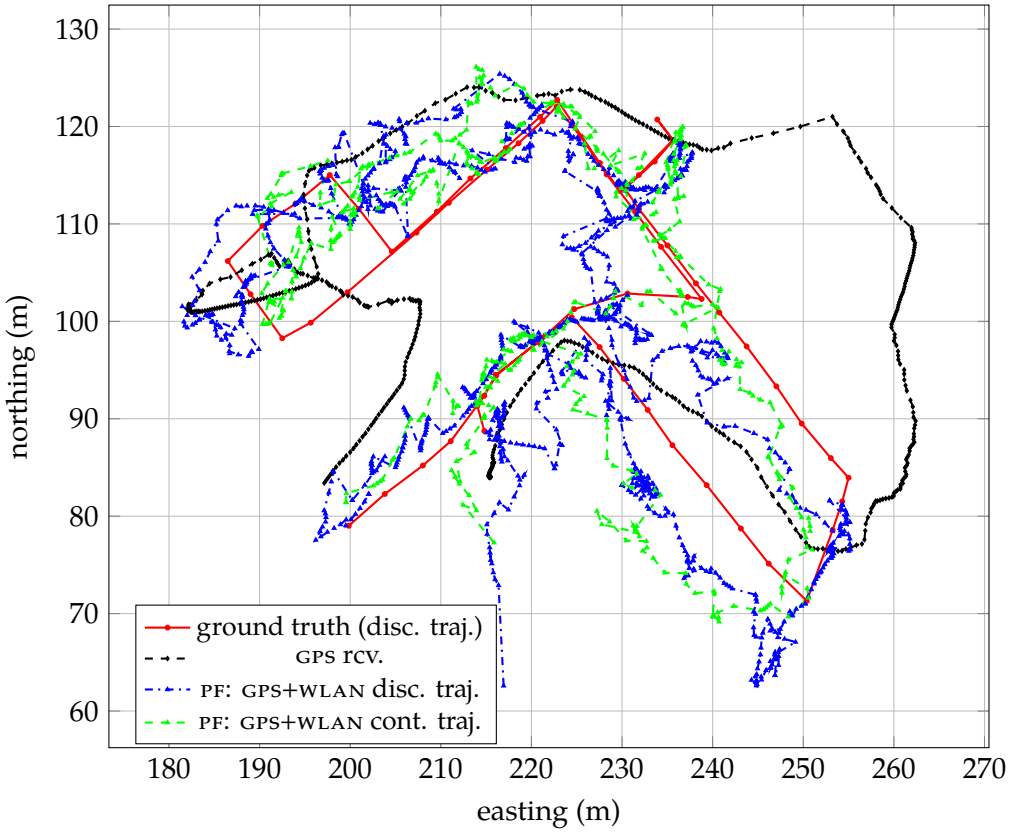


Figure 4.14.: Particle filter estimates and ground truth of Trajectory-4a using GPS and WLAN measurements. WLAN data are averaged over 4 seconds.

In comparison to the GPS+WLAN solution of the temporally discrete trajectory experiment, the estimates appear even more accurate. Part of that impression is certainly the reduction of position jumps, due to the continuous recording of the data.

An other reason for that improvement may lie in the poor GPS signal reception, due to which the filter weighs more the WLAN data, yielding a better overall performance.

The effects of the poor GPS signal reception scenario can be observed in the trajectory estimates of the u-blox GPS receiver: Only during the open area section, at the beginning of the trajectory, the receiver estimates the trajectory reasonable accurate. The performance of the receiver stays low until the loop way around the small lawn; afterwards the receiver looses its position fix completely. In contrast, the particle filter still provides good estimates, also because of its ability to make advantage of less than four pseudoranges – already discussed by Hejc, Seitz and Vaupel (2014) and Khider et al. (2013). Hence, the particle filter is also more robust and, naturally – because of the availability of RSSI indoors – yields a better coverage.

The above outcomes demonstrate additionally that the chosen motion model works also for continuous motion, these results show that it is indeed very general.

4.5. Conclusion

We presented a particle filter that fuses pseudoranges and WLAN RSSI. The algorithm is designed as a PVT as used in GNSS receivers, having access to pseudoranges and WLAN RSSI simultaneously.

Integrating GNSS pseudoranges and WLAN RSSI has already been accomplished. However, instead of excluding multipath affected GNSS signals, this work focuses on a balanced weighting of the two sensor data. Our method achieves smooth seamless transitions between indoors and outdoors by automatically weighting the observations based upon probabilities deduced from their likelihood functions; in which the presented RSSI likelihood function plays a key role. As well in areas that can not clearly described as indoor or outdoor, the particle filter's performance is accurate and reliable. The proposed algorithm yielded a median error of 5 m and a mean error of about 7 m in a realistic, challenging indoor/outdoor environments. A drawback is certainly its slow initial convergence after the first measurements are received.

Because of the complementary performance of location fingerprinting and GNSS in contrastive environments, the presented hybrid particle filter demonstrated improved accuracy and precision compared with the particle filter solutions using solely GPS or WLAN. Moreover, the functioning of WLAN location fingerprinting in GPS denied environments increases the availability of the positioning system;

and besides, the filter's capability to make use of only two pseudoranges makes it more robust, too.

We found that the hybrid solution also shows smaller errors compared with the used ('professional'-grade) GPS receiver. What is more, our results suggest that if the motion of the mobile platform is smoother and uninterrupted the performance can further improved.

Chapter 5

Conclusive remarks

The objective of this study is to develop a multi-sensor data fusion method able to estimate a position in medium- to large-scale indoor/outdoor environment with consumer grade sensors. This includes of GNSS denied areas and also areas where no WLAN may be available.

Therefore we first investigated and evaluated an appropriate method to interpolate RSSIs on space, in order to have a common state space for the data fusion task. We revisit Gaussian process regression modelling for RSSI and found an appropriate model that outperforms the previously suggested Gaussian process models.

This interpolation approach is used to develop a WLAN location estimator based on the ML principle. The accuracy of this WPS is about 5 m mean error and it is in particular robust against the ambiguity problem that fingerprinting methods suffer. Beside reducing the burden of constructing the fingerprint database, interpolating RSSIs additionally makes the location estimation more robust.

To combine RSSI with pseudoranges the Gaussian process based likelihood function was integrated in the framework of the sequential Bayesian estimator by computing the likelihood for the RSSI measurement for each particle. The resulting hybrid positioning system fuses WLAN RSSI and GNSS pseudoranges in a natural way. As well for the pseudorange observations, the likelihood for each particle is computed.

These functions essentially realise the automatic weighting and integration into the location estimation process, yielding smooth indoor/outdoor transitions, in environments difficult for WLAN fingerprinting and GNSS. In these harsh environments the hybrid positioning system yields an average error of about 7 m.

The ability of using information of only two satellites and the robustness due to the RSSI interpolation makes the overall system more robust and increases the coverage of the system.

We anticipate that incorporating further signal processing – as commonly used by commercial receivers – such as pseudorange smoothing, integrity monitoring or multipath mitigation, will further improve our method. As well the use multi-constellation- and multi-frequency receiver will improve the accuracy, precision, robustness and coverage.

For specific applications one needs to adapt the motion model. It potentially further reduces the accuracy and precision of the positioning system.

And of course, further improvement can be achieved by incorporating additional sensors, in particular when the sensor data is fused tightly. Especially the use of inertial sensors demonstrated good results in previous works. Inertial sensor data can be fed into an adequate motion model and thus reduces the positioning errors.

There is also space for improvement on the WLAN side. A first point is the variance used for the RSSI likelihood function. The RSSI likelihood function employs variances of the predicted RSSI distribution instead of a deviation that actually describes the uncertainty of the observed RSSI. Integrating a uncertainty measure of the actual observation in the positioning phase would most likely improve the positioning performance.

The detection of a indoor or outdoor area is an other issue worth investigating. Using for example the WLAN-only mode indoors, could limit battery drainage of mobile devices, though pseudoranges may, if received, improve the localisation accuracy indoors. Annotating the radio map with the respective environment is one approach. It relies on the correct recognition of the RSSIS, if, for example, the best matching RSSIS correspond to a fingerprint of the indoor radio map, the user/object is likely indoors.

These benefits achieved but also the potential benefits mentioned above come for an increase in computational costs. In particular, predicting the RSSIS at the particle's position is computationally demanding. The prediction requires the inversion of the (square) covariance matrix, whose dimension corresponds to the number of particles. Solutions to this problem is left for further research. The amount of computational resources that the particle filter consumes is in comparison a minor problem. Nevertheless, it can be reduced by applying parametric filters, such as Kalman filters, to the conditionally linear-Gaussian structure of

the state space. As the designated area of the positioning system becomes very large, searches through large radio map databases are also computationally costly. Clustering methods and hierarchical database designs can alleviate that problem.

Scalability is always an issue for positioning systems. In this study it depends basically on the scalability of the WPS, as GNSS has world-wide coverage. Scalability of WPS is limited by the construction of radio maps. Interpolation techniques, as for example Gaussian process regression, facilitate this step, but further automation by robots or crowd-sourcing are certainly beneficial and needed. An other point of concern are the sizes of databases of large-area WPS.

If the area covered by the WPS is distributed over different administrative units (states, countries and so on), the administration of the resulting vast databases would probably be decentralised, or even community managed as OSM data. For such a system, one would have to find a consent about the minimum set of database entries (including coordinate reference system, units, etc.), a strategy to keep the databases up to date and mechanisms to manage the vast databases. Clearly, all these points need to be designed with different applications and users in mind.

Appendix A

The Bayes recursion

A.1. Derivation of sequential update of the filtering density

The sequential Bayesian estimator is a unifying conceptual framework for many estimation and control problems. It describes the marginal *a posteriori* PDF, $p(x_k \mid y_{1:k})$, of the Bayesian formalism, as the solution to these classes of problems, where x_k is the hidden state and y_k the observation. The index k indicates that these values are the latest of the corresponding time sequence of states and observations, and $x_{0:k} = \{x_0, \ldots, x_k\}$ denotes the complete sequence and $y_{1:k}$ respectively for the observations. Additionally, we assume that

- an initial state pdf is available $p(x_0)$ (prior density), that
- states follow a first-order Markov process: $p(x_k \mid x_{0:k-1}) = p(x_k \mid x_{k-1})$, and that
- the observations of the current time step are only dependent upon the current state: $p(y_k \mid y_{1:k-1}, x_{0:k}) = p(y_k \mid x_k)$.

Sought is the state most recent state x_k as a point estimate of the posterior $p(x_k \mid y_{1:k})$ given the set of observations $y_{1:k}$. The above mentioned assumptions allow us to derive the sequential Bayesian estimator:

$$\begin{aligned} p(x_k \mid y_{1:k}) &= \frac{p(y_{1:k}, x_k)}{p(y_{1:k})} = \frac{p(y_{1:k} \mid x_k)p(x_k)}{p(y_{1:k})} \\ &= \frac{p(y_k, y_{1:k-1} \mid x_k)p(x_k)}{p(y_k, y_{1:k-1})} \end{aligned}$$

$$
\begin{aligned}
&= \frac{p(y_k \mid y_{1:k-1}, x_k) p(y_{1:k-1} \mid x_k) p(x_k)}{p(y_k \mid y_{1:k-1}) p(y_{1:k-1})} \\
&= \frac{p(y_k \mid y_{1:k-1}, x_k) p(x_k \mid y_{1:k-1}) \cancel{p(y_{1:k-1})} \cancel{p(x_k)}}{p(y_k \mid y_{1:k-1}) \cancel{p(y_{1:k-1})} \cancel{p(x_k)}} \\
&= \frac{p(y_k \mid x_k) p(x_k \mid y_{1:k-1})}{p(y_k \mid y_{1:k-1})}. \qquad \text{(A.1)}
\end{aligned}
$$

Equation (A.1) determines the posterior by means of the likelihood function, $p(y_k \mid x_k)$, the prediction of the posterior, $p(x_k \mid y_{1:k-1})$, and the evidence, $p(y_k \mid y_{1:k-1})$.

The next step is to obtain the prediction of the posterior PDF, the second term of the nominator. The Chapman-Kolmogorv equation provides an representation in terms of the transitional PDF and the posterior PDF calculated at the previous time step. Again the Markov property and total probability theorem are applied

$$
\begin{aligned}
p(x_k \mid y_{1:k-1}) &= \int p(x_k, x_{k-1} \mid y_{1:k-1}) \, \mathrm{d}x_{k-1} \\
&= \int p(x_k \mid x_{k-1}, y_{1:k-1}) p(x_{k-1} \mid y_{1:k-1}) \, \mathrm{d}x_{k-1} \qquad \text{(A.2)} \\
&= \int p(x_k \mid x_{k-1}) p(x_{k-1} \mid y_{1:k-1}) \, \mathrm{d}x_{k-1}.
\end{aligned}
$$

For completeness, the denominator of (A.1) yields

$$
\begin{aligned}
p(y_k \mid y_{1:k-1}) &= \int p(y_k, x_k \mid y_{1:k-1}) \, \mathrm{d}x_k \\
&= \int p(y_k \mid x_k, y_{1:k-1}) p(x_k \mid y_{1:k-1}) \, \mathrm{d}x_k \\
&= \int p(y_k \mid x_k) p(x_k \mid y_{1:k-1}) \, \mathrm{d}x_k.
\end{aligned}
$$

See (Bar-Shalom et al. 2001, pp. 373) for comparison.

A.2. Derivation of sequential update of the posterior density

As for the marginal posterior density, one can derive a recursive expression for the complete trajectory of states $p(x_{0:k} \mid y_{1:k})$, employing the Markov property for the state and measurement process and Bayes' theorem:

$$
\begin{aligned}
p(x_{0:k} \mid y_{1:k})p(y_{1:k}) &= p(x_{0:k}, y_{1:k}) \\
p(x_{0:k} \mid y_{1:k})p(y_k \mid y_{1:k-1})\cancel{p(y_{1:k-1})} &= p(y_k, x_{0:k} \mid y_{1:k-1})\cancel{p(y_{1:k-1})} \\
p(x_{0:k} \mid y_{1:k}) &= \frac{p(y_k \mid x_{0:k}, y_{1:k-1})p(x_{0:k} \mid y_{1:k-1})}{p(y_k \mid y_{1:k-1})} \\
&= \frac{p(y_k \mid x_{0:k}, y_{1:k-1})p(x_k \mid x_{0:k-1}, y_{1:k-1})p(x_{0:k-1} \mid y_{1:k-1})}{p(y_k \mid y_{1:k-1})} \\
&= \frac{p(y_k, x_k \mid x_{0:k-1}, y_{1:k-1})p(x_{0:k-1} \mid y_{1:k-1})}{p(y_k \mid y_{1:k-1})} \\
&= \frac{p(y_k \mid x_k)p(x_k \mid x_{0:k-1}, y_{1:k-1})p(x_{0:k-1} \mid y_{1:k-1})}{p(y_k \mid y_{1:k-1})} \\
&= \frac{p(y_k \mid x_k)p(x_k \mid x_{k-1})p(x_{0:k-1} \mid y_{1:k-1})}{p(y_k \mid y_{1:k-1})} \\
&= \frac{p(y_k \mid x_k)p(x_k \mid x_{k-1})}{p(y_k \mid y_{1:k-1})} p(x_{0:k-1} \mid y_{1:k-1}).
\end{aligned}
$$

See for example (Doucet, Freitas et al. 2001; Doucet and Johansen 2009). Note, that equation (A.1) can also be determined by integrating over the above equation with respect to $x_{1:k-1}$.

Appendix B

Adaption of Gaussian process models

It is possible to 'learn' the hyperparameters, in other words to adapt this Gaussian process model to the concrete problem. Let $\boldsymbol{\theta}$ denote the set of hyperparameters $\{\sigma_\varepsilon^2, \ell, \sigma_f^2\}$.

The problem of model selection can be expressed in a Bayesian perspective. Consider the equations (2.33), (2.34) plugged into (2.32). The uncertainty about the hyperparameters may be encoded in the posterior distribution by conditioning the distribution on the hyperparameters. To treat this dependence of the posterior distribution (2.32) on the hyperparameters, one would ideally integrate out $\boldsymbol{\theta}$

$$p(\mathbf{f}^* \mid X^*, \mathbf{y}, X) = \int p(\mathbf{f}^* \mid X^*, \mathbf{y}, X, \boldsymbol{\theta}) p(\boldsymbol{\theta} \mid \mathbf{y}, X) \, \mathrm{d}\boldsymbol{\theta}. \tag{B.1}$$

This includes the determination of the posterior over the hyperparameters

$$p(\boldsymbol{\theta} \mid \mathbf{y}, X) = \frac{p(\mathbf{y} \mid X, \boldsymbol{\theta}) p(\boldsymbol{\theta})}{p(\mathbf{y} \mid X)} \tag{B.2}$$

which in turn makes the evaluation of its denominator, given by

$$p(\mathbf{y} \mid X) = \int p(\mathbf{y} \mid X, \boldsymbol{\theta}) p(\boldsymbol{\theta}) \, \mathrm{d}\boldsymbol{\theta}, \tag{B.3}$$

necessary. A more detailed derivation can be found in (Bishop 2006, sec. 3.4; Rasmussen et al. 2006, sec. 5.2).

The integrals (B.1) and (B.3) are in general intractable. A common solution is to approximate the posterior over function (B.1) by picking likely values of

hyperparameters (MacKay 2005, sec. 45.5; Bishop 2006, sec. 6.4.3; Rasmussen et al. 2006, sec. 5.2)

$$p(\mathbf{f}^* \mid X^*, \mathbf{y}, X) \simeq p(\mathbf{f}^* \mid X^*, \mathbf{y}, X, \boldsymbol{\theta}_P).$$

The most likely values of the hyperparameters can be obtained by an approximation often referred to as type II maximum likelihood procedure. Instead of maximising the posterior over hyperparameters, $\boldsymbol{\theta}_p = \arg\max_\theta p(\boldsymbol{\theta} \mid \mathbf{y}, X)$, equation (B.2) is applied and the most probable values for the hyperparameters are yielded by

$$\boldsymbol{\theta}_P = \arg\max_\theta \frac{p(\mathbf{y} \mid X, \boldsymbol{\theta}) p(\boldsymbol{\theta})}{p(\mathbf{y} \mid X)}.$$

Disregarding the denominator, which is independent of $\boldsymbol{\theta}$, the latter equation may be rewritten as

$$\boldsymbol{\theta}_P = \arg\max_\theta (p(\mathbf{y} \mid X, \boldsymbol{\theta}) p(\boldsymbol{\theta})). \tag{B.4}$$

In practice the search for the hyperparameters (B.4) can be implemented by gradient based optimisation algorithm such as the conjugate gradients. Thus, the partial derivative with respect to the hyperparameters must be derived. Without a priori knowledge about the hyperparameters the second term, $p(\boldsymbol{\theta})$, is assumed to be uniformly distributed, hence it becomes constant and its derivatives vanish. Finally, the optimal hyperparameters are determined by maximising $p(\mathbf{y} \mid X)$, which is the evidence for the hyperparameters. This term is the likelihood function for the hyperparameters and $\boldsymbol{\theta}_P$ becomes a maximum likelihood estimate, or respectively the result of the minimisation of the negative log-likelihood function (Schölkopf et al. 2002, pp. 487–488; MacKay 2005, p. 546)

$$\boldsymbol{\theta}_P = \arg\min_\theta (-\ln p(\mathbf{y} \mid X, \boldsymbol{\theta})). \tag{B.5}$$

The distribution in equation (B.5) – the marginal likelihood function – can be derived by integrating over the product of the likelihood function and the prior distribution: $p(\mathbf{y} \mid X) = \int p(\mathbf{y} \mid \mathbf{f}, X) p(\mathbf{f} \mid X) \, \mathrm{d}\mathbf{f}$. Under some sensible assumptions an analytic solution exists for this integral. A Gaussian process prior distribution with a zero mean function is a normal distribution $\mathbf{f} \mid X \sim \mathcal{N}(0, K)$. As mentioned in section 2.3.2, the noise process of the underlying function is

assumed to be Gaussian distributed, $\epsilon \sim \mathcal{N}(0, \sigma_\epsilon^2)$, thus also the Gaussian process likelihood function is normal $\mathbf{y} \mid \mathbf{f}, X \sim \mathcal{N}(\mathbf{f}, \sigma_\epsilon^2 I)$. As the product of two Gaussian distributions give another (un-normalised) Gaussian distribution, the marginal likelihood function becomes $\mathbf{y} \mid X \sim \mathcal{N}(0, K + \sigma_\epsilon^2 I)$ (Rasmussen et al. 2006, p. 16; Schölkopf et al. 2002, pp. 487)

$$p(\mathbf{y} \mid X, \boldsymbol{\theta}) = \frac{1}{(2\pi)^N |K + \sigma_\epsilon^2 I|} \exp(-\frac{1}{2}(\mathbf{y}^T (K + \sigma_\epsilon^2 I)^{-1} \mathbf{y})).$$

Thus, the log-likelihood function can be given with

$$\ln p(\mathbf{y} \mid X, \boldsymbol{\theta}) = -\frac{N}{2} \ln 2\pi - \frac{1}{2} \ln |K + \sigma_\epsilon^2 I| - \frac{1}{2}(\mathbf{y}^T (K + \sigma_\epsilon^2 I)^{-1} \mathbf{y}).$$

The optimisation problem of maximising the log-likelihood function, is commonly solved numerically. Hence, the derivatives with respect to the hyperparameters are determined. The derivation of the negative log-likelihood function with respect to a hyperparameter is

$$\frac{\partial}{\partial \boldsymbol{\theta}} \ln p(\mathbf{y} \mid X, \boldsymbol{\theta}) = -\frac{1}{2} \operatorname{tr} \left((K + \sigma_\epsilon^2 I)^{-1} \frac{\partial (K + \sigma_\epsilon^2 I)}{\partial \boldsymbol{\theta}} \right) + \frac{1}{2}(\mathbf{y}^T (K + \sigma_\epsilon^2 I)^{-1} \frac{\partial (K + \sigma_\epsilon^2 I)}{\partial \boldsymbol{\theta}} (K + \sigma_\epsilon^2 I)^{-1} \mathbf{y}). \quad \text{(B.6)}$$

A gradient search algorithm, employing the derivatives to find the minimum, may provide the optimal hyperparameters.

Appendix C

Comparison of hyperparameters

In section 3.4.2 we have seen that the Gaussian process prior distributions for RSSIs in either indoor or outdoor environment are indistinguishable. The following tables compare the hyperparameters of Gaussian process models, to check if models adapted for either indoor or outdoor environments are distinguishable by their hyperparameters.

The Gaussian process regression models were fitted to data of the

1. Indoor,
2. Outdoor-1 or
3. Outdoor-2

environments. Each table presents one hyperparameter for six models, corresponding to six access points, AP-1–AP-6. All models are based on the constant mean function and Matérn class functions.

The class of Matérn functions rely on the modified Bessel function K_ν. For the two input points $\mathbf{x}_p$ and $\mathbf{x}_q$ it is given by

$$k_{\text{MAT}}(\mathbf{x}_p, \mathbf{x}_q) = \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu}}{\ell} ||\mathbf{x}_p - \mathbf{x}_q|| \right)^\nu K_\nu \left(\frac{\sqrt{2\nu}}{\ell} ||\mathbf{x}_p - \mathbf{x}_q|| \right),$$

where ν determines its roughness and ℓ its length scale. Both parameter influence the ability of the function to adapt to rather smooth or rough data. As ν decreases the Matérn function becomes rougher and as ν increases it becomes smoother. For $\nu \to \infty$ Matérn function converges to the squared exponential function (2.26).

Choosing $\nu = a + 1/2$, where a is a non-negative integer leads to a simple expression of the Matérn function as product of an exponential and a polynomial of order a (Rasmussen et al. 2006). The most common values for for machine learning applications are ν are $3/2$ and $5/2$ (Rasmussen et al. 2006). For $\nu = 3/2$ the Matérn function is given by

$$k_{\nu=3/2}(\mathbf{x}_p, \mathbf{x}_q) = \sigma_f^2 \left(1 + \frac{\sqrt{3}}{\ell}||\mathbf{x}_p - \mathbf{x}_q||\right) \exp\left(-\frac{\sqrt{3}}{\ell}||\mathbf{x}_p - \mathbf{x}_q||\right) + \sigma_n^2.$$

Here, we added in addition two parameters: σ_f^2 to scale the signal power and σ_n^2 to model the power of additive noise noise.

As the Gaussian process prior distributions determine the principle structure, and the hyperparameters adapt the process to the training data. We compare the optimized hyperparameters (ℓ – table C.1, σ_f – table C.2, σ_n – table C.3) of three models, which differ by their prior distribution's roughness parameter ($\nu = 1/2, 3/2, 5/2$).

The estimated length scale is shown in table C.1. No clear pattern of the parameter ℓ for indoor and outdoor environment is observable. For the indoor data one may expect smaller length scale as for outdoor data, but this varies for different access points and different prior distributions ($\nu = 1/2$ and $\nu = 3/2, 5/2$.

The estimated signal magnitude is shown in table C.2. The strength of the signal is difficult to predict – being the reason why one resorts to fingerprinting in the first place – since it depends on the signal propagation path between access point and fingerprint position. They are different for each access point, thus, the estimate of σ_f is expected to be different for each access point and scenario.

The estimated noise magnitude is shown in table C.3. The estimates are similar for all three environments, a significant difference for indoor and outdoor environments is not visible.

Table C.1.: Hyperparameter ℓ (length scale) optimized with RSSI training data from different regions (Indoor, Outdoor-1, Outdoor-2) of Matérn covariance function with different parameter ν.

	$\nu = 1/2$		
	Indoor	Outdoor-1	Outdoor-2
AP-1	34.4	55.3	27.6
AP-2	19.5	29.8	53.6
AP-3	16.9	31.0	13.4
AP-4	13.0	75.0	11.9
AP-5	29.9	19.0	23.0
AP-6	14.0	4.4	9.3

	$\nu = 3/2$		
	Indoor	Outdoor-1	Outdoor-2
AP-1	18.1	18.6	17.8
AP-2	10.1	13.2	25.6
AP-3	7.4	13.6	13.0
AP-4	6.5	56.0	7.2
AP-5	15.7	8.6	16.1
AP-6	10.7	3.1	9.9

	$\nu = 5/2$		
	Indoor	Outdoor-1	Outdoor-2
AP-1	14.8	14.7	15.8
AP-2	7.8	10.8	21.5
AP-3	6.1	10.4	12.5
AP-4	5.4	44.8	6.8
AP-5	12.9	7.2	26.1
AP-6	11.6	2.9	19.1

Table C.2.: Hyperparameter σ_f (signal standard deviation) optimized with RSSI training data from different regions (Indoor, Outdoor-1, Outdoor-2) of Matérn covariance function with different parameter ν.

	$\nu = 1/2$		
	Indoor	Outdoor-1	Outdoor-2
AP-1	8.2	13.6	8.4
AP-2	6.5	11.2	10.4
AP-3	14.5	8.4	6.1
AP-4	13.8	6.1	9.2
AP-5	11.1	7.4	5.1
AP-6	2.4	4.2	3.3

	$\nu = 3/2$		
	Indoor	Outdoor-1	Outdoor-2
AP-1	9.7	14.5	7.8
AP-2	6.7	8.0	9.5
AP-3	14.6	10.0	5.7
AP-4	13.9	8.3	9.0
AP-5	12.0	6.9	4.2
AP-6	2.7	4.0	3.1

	$\nu = 5/2$		
	Indoor	Outdoor-1	Outdoor-2
AP-1	9.5	14.2	7.6
AP-2	6.4	7.6	9.2
AP-3	14.7	9.3	5.7
AP-4	13.9	8.1	8.6
AP-5	11.7	6.7	5.5
AP-6	2.7	4.0	3.1

Table C.3.: Hyperparameter σ_n (noise standard deviation) optimized with RSSI training data from different regions (Indoor, Outdoor-1, Outdoor-2) of Matérn covariance function with different parameter ν.

	$\nu = 1/2$		
	Indoor	Outdoor-1	Outdoor-2
AP-1	3.7	3.5	3.8
AP-2	3.4	3.5	2.8
AP-3	5.3	3.7	4.8
AP-4	4.8	3.5	0.0
AP-5	4.4	3.6	0.9
AP-6	2.9	3.6	3.8

	$\nu = 3/2$		
	Indoor	Outdoor-1	Outdoor-2
AP-1	3.8	3.8	4.5
AP-2	3.5	4.0	3.7
AP-3	5.5	4.1	5.3
AP-4	5.2	3.7	0.6
AP-5	4.5	3.8	2.8
AP-6	3.0	3.7	4.0

	$\nu = 5/2$		
	Indoor	Outdoor-1	Outdoor-2
AP-1	3.9	3.9	4.6
AP-2	3.5	4.0	3.8
AP-3	5.5	4.1	5.3
AP-4	5.2	3.7	0.7
AP-5	4.5	3.8	2.9
AP-6	3.0	3.8	4.1

Nomenclature

Abbreviations

API	application programming interface
BIC	Bayesian information criterion
CDF	cummulative distribution function
CDMA	code division multiple access
EAP	expected a posteriori
GNSS	global navigation satellite system
GNU	Unix-like operating system, being a recursive acronym for 'GNU's not Unix.'
GPS	global positioning system
IEEE 802.11	IEEE standard for information technology 802, specific requirements part 11: wireless LAN medium access control (MAC) and physical layer (PHY) specifications
I.I.D.	identical, independent distributed
INS	inertial navigation systems
JNI	java native interface
KF	Kalman filter

k-NN	*k*-nearest neighbours
LBS	location-based service
LOS	line-of-sight
MAC	media access control
MAP	maximum a posteriori
ML	maximum likelihood
MLE	maximum likelihood estimator
MMSE	minumum mean square error
NLOS	non-line-of-sight
OSM	open street map
PDF	probability density function
PF	particle filter
PRN	pseudo random noise
PVT	position, velocity and time
RMS	root mean square
RMSE	root mean square error
RSSI	received signal strength indicator
SD	standard deviation
SQL	structured query language
TDOA	time difference of arrival
TOA	time of arrival
TTFF	time-to-first-fix
USB	universal serial bus
WLAN	wireless local area network
WPS	WLAN positioning system

Notation

$\arg\max_a$	the argument maximising the operand
$\mu(\cdot)$	mean (function)
$\text{cov}(\cdot)$	covariance (function)
$\delta(\cdot)$	Dirac function
$p(\cdot)$	probability or probability density function
$p(\cdot \mid \cdot)$	conditional probability or conditional probability density function
$\|\| \cdot \|\|$	2-norm
$\mathbb{E}[\cdot]$	expectation
$\frac{\partial}{\partial \mathbf{a}}$	partial derivate with respect to $\mathbf{a}$
$\mathcal{A}$	set
a	scalar
$\mathbf{a}$	column vector
$a_{[i]}$	ith element of $\mathbf{a}$
$a_{0:N}$	sequence of values $a_{0:N} \triangleq a_0, a_1, \ldots, a_N$
a_t	continuous-time variable
a_k	discrete-time variable
$\hat{a}$	estimate of a parameter
A	matrix
A^{-1}	inverse of a matrix
A^T	transpose of a matrix (or vector)
$\|A\|$	determinant of a matrix
$\text{tr}(\cdot)$	trace of a matrix
0	zero matrix
I	identity matrix

Bibliography

Alspach, D. L., and H. W. Sorenson. 1972. 'Nonlinear Bayesian estimation using Gaussian sum approximations'. *IEEE Transactions on Automatic Control* 17, no. 4 (August): 439–448. doi:10.1109/TAC.1972.1100034. (Cited on page 21).

Anagnostopoulos, G. G., and M. Deriaz. 2015. 'Automatic switching between indoor and outdoor position providers'. In *International Conference on Indoor Positioning and Indoor Navigation (IPIN), 2015.* October. doi:10.1109/IPIN.2015.7346948. (Cited on page 5).

Atia, M. M., A. Noureldin and M. J. Korenberg. 2012. 'Dynamic Propagation Modeling for Mobile Users' Position and Heading Estimation in Wireless Local Area Networks'. *IEEE Wireless Communications Letters* 1, no. 2 (April): 101–104. doi:10.1109/WCL.2012.020612.110279. (Cited on pages 48, 50).

———. 2013. 'Dynamic Online-Calibrated Radio Maps for Indoor Positioning in Wireless Local Area Networks'. *IEEE Transactions on Mobile Computing* 12, no. 9 (September): 1774–1787. doi:10.1109/TMC.2012.143. (Cited on page 95).

awiloc. 2013. 'awiloc®: awiloc® Fraunhofer IIS's self-contained Wireless LAN Positioning Technology for Cities and Buildings'. Fraunhofer Institute for Integrated Circuits. Accessed 19th February 2013. http://www.awiloc.com. (Cited on page 42).

Bagdonas, K., and K. Borre. 2008. 'Ubiquitous WiFi/GNSS Positioning System - TOA Based Distance Estimation'. In *Proceedings of the 21st International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2008),* 1773–1779. Savannah, GA, September. (Cited on page 42).

Bahl, P., and V. N. Padmanabhan. 2000. 'RADAR: an in-building RF-based user location and tracking system'. In *Proceedings of Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. INFOCOM 2000.* 2:775–784. doi:10.1109/INFCOM.2000.832252. (Cited on pages 43, 45).

Bancroft, S. 1985. 'An Algebraic Solution of the GPS Equations'. *IEEE Transactions on Aerospace and Electronic Systems* AES-21, no. 1 (January): 56–59. doi:10.1109/TAES.1985.310538. (Cited on page 40).

Barber, D. 2014. *Bayesian Reasoning and Machine Learning*. DRAFT February 25, 2014. http://web4.cs.ucl.ac.uk/staff/D.Barber/pmwiki/pmwiki.php?n=Brml.Online. (Cited on page 16).

Bar-Shalom, Y., X.-R. Li and T. Kirubarajan. 2001. *Estimation with applications to tracking and navigation.* Theory Algorithms and Software. John Wiley & Sons, Inc. (Cited on pages 14, 17–19, 103, 142).

Battiti, R., M. Brunato and A. Villani. 2002. *Statistical Learning Theory for Location Fingerprinting in Wireless LANs.* Technical Report DIT-0086. Technical report. Informatica e Telecomunicazioni, Università degli Studi di Trento, October. Accessed 10th June 2013. http://rtm.science.unitn.it/~battiti/archive/86.pdf. (Cited on page 47).

Beder, C., and M. Klepal. 2012. 'Fingerprinting based localisation revisited: A rigorous approach for comparing RSSI measurements coping with missed access points and differing antenna attenuations'. In *International Conference on Indoor Positioning and Indoor Navigation (IPIN).* November. doi:10.1109/IPIN.2012.6418940. (Cited on pages 45, 48, 51, 94).

Bekkali, A., T. Masuo, T. Tominaga, N. Nakamoto and H. Ban. 2011. 'Gaussian processes for learning-based indoor localization'. In *IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC), 2011*, 1–6. September. doi:10.1109/ICSPCC.2011.6061737. (Cited on pages 50, 77, 78).

Bishop, C. M. 2006. *Pattern Recognition and Machine Learning.* 1st ed. Edited by M. Jordan, J. Kleinberg and B. Schölkopf. Springer New York. (Cited on pages 33, 145, 146).

Bittins, B., and J. Sieck. 2011. 'Multisensor and Collaborative Localization for Diverse Environments'. In *Fifth UKSim European Symposium on Computer Modeling and Simulation (EMS), 2011*, 406–411. November. doi:10.1109/EMS.2011.60. (Cited on page 5).

blackberry.com. 2013. 'Location-Based Services'. Research In Motion Limited. Accessed 6th January 2013. http://www.blackberry.com/developers/docs/6.0.0api/lbs-summary.html. (Cited on page 5).

Bonnor, N. 2012. 'A Brief History of Global Navigation Satellite Systems'. *Journal of Navigation* 65 (01): 1–14. doi:10.1017/S0373463311000506. (Cited on pages 36–38).

Brown, R. G., and P. Y. C. Hwang. 1997. *Introduction to Random Signals and Applied Kalman Filtering: With Matlab Exercises and Solutions.* 3rd ed. John Wiley & Sons. (Cited on page 102).

Chai, X., and Q. Yang. 2007. 'Reducing the Calibration Effort for Probabilistic Indoor Location Estimation'. *IEEE Transactions on Mobile Computing* 6 (6): 649–662. doi:10.1109/TMC.2007.1025. (Cited on pages 50, 51).

Chan, J., and E. T. Wong. 2012. 'Empirical modelling of received signal strength in indoor localization'. In *International Conference on Automatic Control and Artificial Intelligence, (ACAI 2012)*, 978–981. doi:10.1049/cp.2012.1140. (Cited on page 51).

Chen, A., C. Harko, D. Lambert and P. Whiting. 2007. 'An Algorithm for Fast, Model-free Tracking Indoors'. *SIGMOBILE Mobile Computing and Communications Review* (New York, NY, USA) 11, no. 3 (July): 48–58. doi:10.1145/1317425.1317431. (Cited on pages 51, 52).

Chitte, S. D., S. Dasgupta and Z. Ding. 2009. 'Source Localization from Received Signal Strength Under Log-Normal Shadowing: Bias and Variance'. In *2nd International Congress on Image and Signal Processing, CISP.* October. doi:10.1109/CISP.2009.5301003. (Cited on page 44).

Corte-Valiente, A. del, J. M. Gómez-Pulido and O. Gutiérrez-Blanco. 2009. 'Efficient techniques and algorithms for improving indoor localization precision on wlan networks applications'. *International Journal of Communications, Network and System Sciences* 2 (07): 645. (Cited on page 47).

Dardari, D., M. Luise and E. Falletti, eds. 2012. *Satellite and Terrestrial Radio Positioning Techniques: A signal processing perspective.* 1st ed. Oxford, UK: Academic Press. (Cited on page 2).

developer.android.com. 2013. 'Location Strategies'. Google Inc. and the Open Handset Alliance. 8th January. Accessed 6th January 2013. http://developer.android.com/guide/topics/location/strategies.html. (Cited on page 5).

developer.apple.com. 2013. 'CLLocationManager Class Reference'. Apple Inc. Accessed 6th January 2013. https://developer.apple.com/library/ios/#documentation/CoreLocation/Reference/CLLocationManager_Class/CLLocationManager/CLLocationManager.html#//apple_ref/doc/uid/TP40007125. (Cited on page 5).

developer.nokia.com. 2012. 'Location API'. Nokia. 11th January. Accessed 6th January 2013. https://www.developer.nokia.com/Community/Wiki/Location_API. (Cited on page 5).

Djurić, P. M., J. H. Kotecha, J. Zhang, Y. Huang, T. Ghirmai, M. F. Bugallo and J. Miguez. 2003. 'Particle filtering'. *IEEE Signal Processing Magazine* 20, no. 5 (September): 19–38. doi:10.1109/MSP.2003.1236770. (Cited on pages 23, 27).

Doucet, A., N. de Freitas and N. Gordon, eds. 2001. *Sequential Monte Carlo Methods in Practice.* In *Statistics for Engineeringand Information Science,* 1st ed. Springer. doi:10.1007/978-1-4757-3437-9. (Cited on pages 24, 26, 29, 143).

Doucet, A., S. Godsill and C. Andrieu. 2000. 'On sequential Monte Carlo sampling methods for Bayesian filtering'. *Statistics and Computing* 10 (August): 197–208. (Cited on pages 28, 29).

Doucet, A., and A. M. Johansen. 2009. 'A tutorial on particle filtering and smoothing: fifteen years later'. *Handbook of Nonlinear Filtering:* 656–704. (Cited on pages 28, 143).

Duvallet, F., and A. D. Tews. 2008. 'WiFi position estimation in industrial environments using Gaussian processes'. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*. 2216–2221. September. doi:10.1109/IROS.2008.4650910. (Cited on page 50).

Duvenaud, D. 2014. 'Automatic Model Construction with Gaussian Processes' [in English]. PhD diss., Computational and Biological Learning Laboratory, University of Cambridge. Accessed 19th May 2015. http://people.seas.harvard.edu/~dduvenaud/thesis.pdf. (Cited on page 63).

Duvenaud, D., H. Nickisch and C. E. Rasmussen. 2011. 'Additive Gaussian Processes'. In *Advances in Neural Information Processing Systems 24*, 226–234. Granada, Spain. http://arxiv.org/pdf/1112.4394.pdf. (Cited on page 62).

E, J., and J. Ma. 2013. 'A Research on Seamless Indoor and Outdoor Positioning'. *Journal of Computers* 8 (12). http://ojs.academypublisher.com/index.php/jcp/article/view/jcp081230473057. (Cited on page 5).

Eck, D., K. Schilling, A. Abdul-Majeed, J. Thielecke, P. Richter, J. Gutiérrez Boronat, I. Schens, B. Thomas, B. Williger and F. R. Lang. 2012. 'Mobility assistance for older people'. *Applied Bionics and Biomechanics* 9 (1): 69–83. doi:10.3233/ABB-2012-0053. (Cited on page 6).

Elnahrawy, E., X. Li and R. P. Martin. 2004. 'The limits of localization using signal strength: a comparative study'. In *First Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks, 2004. IEEE SECON 2004.* 406–414. October. doi:10.1109/SAHCN.2004.1381942. (Cited on pages 44, 50–52).

Fang, S.-H., and T.-N. Lin. 2008. 'Indoor Location System Based on Discriminant-Adaptive Neural Network in IEEE 802.11 Environments'. *IEEE Transactions on Neural Networks* 19, no. 11 (November): 1973–1978. doi:10.1109/tnn.2008.2005494. (Cited on page 47).

FAA (Federal Aviation Administration). 2016. *Global Positioning System (GPS) Standard Positioning Service (SPS) Performance Analysis Report* 93. Reporting Period: 1 January – 31 March 2016. Federal Aviation Administration, 30th April. Accessed 8th June 2016. http://www.nstb.tc.faa.gov/REPORTS/PAN93_0416.pdf. (Cited on page 2).

Fernandez, D., F. Barcelo-Arroyo, I. Martin-Escalona, M. Ciurana, M. Jofre and E. Gutierrez. 2011. 'Fusion of WLAN and GNSS observables for positioning in urban areas: The position ambiguity'. In *IEEE Symposium on Computers and Communications (ISCC), 2011*, 748–751. July. doi:10.1109/ISCC.2011.5983929. (Cited on page 6).

Fernández-Prades, C., L. L. Presti and E. Falletti. 2011. 'Satellite Radiolocalization From GPS to GNSS and Beyond: Novel Technologies and Applications for Civil Mass Market'. *Proceedings of the IEEE* 99, no. 11 (November): 1882–1904. doi:10.1109/JPROC.2011.2158032. (Cited on page 4).

Ferris, B., D. Hähnel and D. Fox. 2006. 'Gaussian processes for signal strength-based location estimation'. In *Proceedings of Robotics: Science and Systems II*. Cambridge, Massachusetts: MIT Press, August. http://www.roboticsproceedings.org/rss02/p39.html. (Cited on pages 50, 51, 95, 114).

Gallagher, T., B. Li, A. G. Dempster and C. Rizos. 2011. 'Power Efficient Indoor/Outdoor Positioning Handover'. In *Proceedings of the International Conference on Indoor Positioning and Indoor Navigation (IPIN'11)*. September. (Cited on page 5).

Galler, S., J. Schroeder, G. Rahmatollahi, K. Kyamakya and K. Jobmann. 2006. 'Analysis and Practical Comparison of Wireless LAN and Ultra-Wideband Technologies for Advanced Localization'. In *Proceedings of IEEE/ION PLANS 2006*, 1:198–203. doi:10.1109/PLANS.2006.1650604. (Cited on page 42).

Gordon, N. J., D. J. Salmond and A. F. M. Smith. 1993. 'Novel approach to nonlinear/non-Gaussian Bayesian state estimation'. *IEE Proceedings F Radar and Signal Processing* 140, no. 2 (April): 107–113. (Cited on pages 24, 28).

Groves, P. D. 2008. *Principles of GNSS, inertial, and multisensor integrated navigation systems*. 1st ed. Artech House Publishers. (Cited on pages 1, 37, 41).

Günther, A., and C. Hoene. 2005. 'Measuring Round Trip Times to Determine the Distance Between WLAN Nodes'. In *Proceedings 4th International IFIP-TC6 Networking Conference*, 768–779. Springer-Verlag, May. doi:10.1007/11422778_62. (Cited on page 42).

Gustafsson, F., and F. Gunnarsson. 2005. 'Mobile positioning using wireless networks: possibilities and fundamental limitations based on available wireless network measurements'. *IEEE Signal Processing Magazine* 22, no. 4 (July): 41–53. doi:10.1109/MSP.2005.1458284. (Cited on page 13).

Hansen, R., R. Wind, C. S. Jensen and B. Thomsen. 2009. 'Seamless Indoor/Outdoor Positioning Handover for Location-Based Services in Streamspin'. In *Tenth International Conference on Mobile Data Management: Systems, Services and Middleware, MDM '09*. 267–272. May. doi:10.1109/MDM.2009.39. (Cited on page 5).

Hatami, A., and K. Pahlavan. 2006. 'Comparative statistical analysis of indoor positioning using empirical data and indoor radio channel models'. In *3rd IEEE Consumer Communications and Networking Conference, CCNC.* 2:1018–1022. doi:10.1109/CCNC.2006.1593192. (Cited on page 44).

Hejc, G., J. Seitz, J. G. Boronat and T. Vaupel. 2013. 'Seamless Indoor Outdoor Positioning Using Bayesian Sensor Data Fusion on Mobile and Embedded Devices'. In *Proceedings of the 26th International Technical Meeting of The Satellite Division of the Institute of Navigation*, 1252–1259. (Cited on page 6).

Hejc, G., J. Seitz and T. Vaupel. 2014. 'Bayesian sensor fusion of Wi-Fi signal strengths and GNSS code and carrier phases for positioning in urban environments'. In *IEEE/ION Position, Location and Navigation Symposium - PLANS 2014*, 1026–1032. May. doi:10.1109/PLANS.2014.6851470. (Cited on pages 6, 114, 134).

Heurtefeux, K., and F. Valois. 2012. 'Is RSSI a Good Choice for Localization in Wireless Sensor Network?' In *IEEE 26th International Conference on Advanced Information Networking and Applications (AINA)*, 732–739. March. doi:10.1109/AINA.2012.19. (Cited on page 44).

Hightower, J., and G. Borriello. 2001. 'Location systems for ubiquitous computing'. *IEEE Computer* 34, no. 8 (August): 57–66. doi:10.1109/2.940014. (Cited on page 2).

Ho, Y.-C., and R. Lee. 1964. 'A Bayesian approach to problems in stochastic estimation and control'. *IEEE Transactions on Automatic Control* 9, no. 4 (October): 333–339. doi:10.1109/TAC.1964.1105763. (Cited on pages 16, 17).

Honkavirta, V., T. Perälä, S. Ali-Löytty and R. Piché. 2009. 'A Comparative Survey of WLAN Location Fingerprinting Methods'. In *Proceedings of the 6th Workshop on Positioning, Navigation and Communication 2009 (WPNC'09)*, 243–251. March. http://math.tut.fi/posgroup/honkavirta_et_al_wpnc09a.pdf. (Cited on pages 47, 48).

Kaemarungsi, K., and P. Krishnamurthy. 2004a. 'Modeling of indoor positioning systems based on location fingerprinting'. In *Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies, INFOCOM 2004.* 2:1012–1022. doi:10.1109/INFCOM.2004.1356988. (Cited on page 51).

———. 2004b. 'Properties of indoor received signal strength for WLAN location fingerprinting'. In *The First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, 2004. MOBIQUITOUS 2004.* 14–23. doi:10.1109/MOBIQ.2004.1331706. (Cited on pages 44, 51).

———. 2012. 'Analysis of WLAN's received signal strength indication for indoor location fingerprinting'. Special Issue: Wide-Scale Vehicular Sensor Networks and Mobile Sensing, *Pervasive and Mobile Computing* 8 (2): 292–316. doi:10.1016/j.pmcj.2011.09.003. (Cited on pages 51, 52, 65, 67, 69).

Kalman, R. E. 1960. 'A new approach to linear filtering and prediction problems'. *Journal of Basic Engineering* 82 (1): 35–45. (Cited on pages 21, 22).

Kaplan, E. D., and C. J. Hegarty, eds. 2006. *Understanding GPS: principles and applications: Principles and Applications.* 2nd ed. Artech House Publishers. (Cited on pages 2, 3, 36, 37, 39–41, 100, 104).

'Keep Environment Nature's Blog'. 2008. 16th August. Accessed 16th August 2016. http://blog.xuite.net/lwkntu/blog/18689761-(GIS%E5%B0%8E%E8%AB%96-20)GPS%E6%A6%82%E8%AB%96. (Cited on pages 40, 41).

Khider, M., T. Jost, P. Robertson and E. Abdo-Sánchez. 2013. 'Global navigation satellite system pseudorangebased multisensor positioning incorporating a multipath error model'. *IET Radar, Sonar Navigation* 7, no. 8 (October): 881–894. doi:10.1049/iet-rsn.2012.0206. (Cited on page 134).

Koenig, S., M. T. Schmidt and C. Hoene. 2011. 'Multipath mitigation for indoor localization based on IEEE 802.11 time-of-flight measurements'. *2013 IEEE 14th International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM)* (Los Alamitos, CA, USA) (June). doi:10.1109/WoWMoM.2011.5986392. (Cited on page 42).

Kushki, A. 2009. 'A Cognitive Radio Tracking System for Indoor Environments'. PhD diss., University of Toronto, Department of Electrical & Computer Engineering. Accessed 13th June 2013. http://hdl.handle.net/1807/17260. (Cited on page 44).

Kushki, A., K. N. Plataniotis and A. N. Venetsanopoulos. 2007. 'Kernel-Based Positioning in Wireless Local Area Networks'. *IEEE Transactions on Mobile Computing* 6 (6): 689–705. doi:10.1109/TMC.2007.1017. (Cited on pages 47, 48, 51).

Kuusniemi, H., L. Chen, L. Ruotsalainen, L. Pei, Y. Chen and R. Chen. 2011. 'Multi-sensor multi-network seamless positioning with visual aiding'. In *International Conference on Localization and GNSS (ICL-GNSS), 2011*, 146–151. June. doi:10.1109/ICL-GNSS.2011.5955257. (Cited on page 6).

Ladd, A. M., K. E. Bekris, A. Rudys, L. E. Kavraki and D. S. Wallach. 2005. 'Robotics-based location sensing using wireless ethernet' [in English]. *Wireless Networks* 11, no. 1 (January): 189–204. doi:10.1007/s11276-004-4755-8. (Cited on page 51).

Le Dortz, N., F. Gain and P. Zetterberg. 2012. 'WiFi fingerprint indoor positioning system using probability distribution comparison'. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2301–2304. March. doi:10.1109/ICASSP.2012.6288374. (Cited on pages 47, 51).

Li, B., J. Salter, A. G. Dempster and C. Rizos. 2006. 'Indoor Positioning Techniques Based on Wireless LAN'. In *Auswireless Conference.* Accessed 10th June 2013. http://hdl.handle.net/2100/170. (Cited on page 51).

Li, B., Y. K. Tan and A. G. Dempster. 2011. 'Using two global positioning system satellites to improve wireless fidelity positioning accuracy in urban canyons'. *IET Communications* 5, no. 2 (January): 163–171. doi:10.1049/iet-com.2010.0064. (Cited on page 6).

Li, B., Y. Wang, H. K. Lee, A. G. Dempster and C. Rizos. 2005. 'Method for yielding a database of location fingerprints in WLAN'. *IEE Proceedings - Communications* 152, no. 5 (October): 580–586. doi:10.1049/ip-com:20050078. (Cited on page 50).

Li, B., and K. O'Keefe. 2013. 'WLAN TOA Ranging with GNSS Hybrid System for Indoor Navigation'. In *Proceedings of the 26th International Technical Meeting of The Satellite Division of the Institute of Navigation*, 416–425. (Cited on page 6).

Li, X., K. Pahlavan, M. Latva-aho and M. Ylianttila. 2000. 'Comparison of indoor geolocation methods in DSSS and OFDM wireless LAN systems'. In *52nd Vehicular Technology Conference, 2000. IEEE VTS-Fall VTC 2000.* 6:3015–3020. doi:10.1109/VETECF.2000.886867. (Cited on page 42).

Liu, H., H. Darabi, P. Banerjee and J. Liu. 2007. 'Survey of Wireless Indoor Positioning Techniques and Systems'. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews* 37, no. 6 (November): 1067–1080. doi:10.1109/TSMCC.2007.905750. (Cited on pages 4, 13).

Lloyd, J. R., D. Duvenaud, R. Grosse, J. B. Tenenbaum and Z. Ghahramani. 2014. 'Automatic Construction and Natural-Language Description of Nonparametric Regression Models'. In *Association for the Advancement of Artificial Intelligence (AAAI)*. http://arxiv.org/pdf/1402.4304.pdf. (Cited on page 63).

Lorga, J. F. M., P. F. Silva, J. S. Silva, T. F. Rocha da Silva, M. S. Nunes, F. D. Nunes and F. M. G. Sousa. 2010. 'GNSS hybridization for indoor positioning'. In *Proceedings of the 5th ESA Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing (NAVITEC), 2010.* December. doi:10.1109/NAVITEC.2010.5708027. (Cited on page 6).

MacKay, D. J. C. 2005. *Information Theory, Inference and Learning Algorithms.* Fourth printing of the first edition. Cambridge University Press. http://www.inference.phy.cam.ac.uk/mackay/itila/. (Cited on pages 25, 26, 146).

Mardeni, R., and Y. Solahuddin. 2012. 'Path loss model development for indoor signal loss prediction at 2.4 GHz 802.11n network'. In *Proc. Int Microwave and Millimeter Wave Technology (ICMMT) Conf*, 2:1–4. doi:10.1109/ICMMT.2012.6230089. (Cited on page 44).

Mautz, R. 2012. 'Indoor Positioning Technologies' [in English]. Habilitation thesis, ETH Zürich, Department of Civil, Environmental and Geomatic Engineering. Accessed 5th December 2014. doi:10.3929/ethz-a-007313554. (Cited on pages 2, 3).

Mirowski, P., H. Steck, P. Whiting, R. Palaniappan, M. MacDonald and T. K. Ho. 2011. 'KL-divergence kernel regression for non-Gaussian fingerprint based localization'. In *International Conference on Indoor Positioning and Indoor Navigation (IPIN).* September. doi:10.1109/IPIN.2011.6071928. (Cited on pages 45, 47, 48, 51).

Mirowski, P., P. Whiting, H. Steck, R. Palaniappan, M. MacDonald, D. Hartmann and T. K. Ho. 2012. 'Probability kernel regression for WiFi localisation'. Special Issue: Indoor Positioning and Navigation. Part I: Fingerprint, *Journal of Location Based Services* 6 (2): 81–100. doi:10.1080/17489725.2012.694723. (Cited on pages 45, 48, 52).

Nur, K., S. Feng, C. Ling and W. Ochieng. 2013. 'Integration of GPS with a WiFi high accuracy ranging functionality'. *Geo-spatial Information Science* 16 (3): 155–168. doi:10.1080/10095020.2013.817106. (Cited on page 6).

Pahlavan, K., F. O. Akgul, M. A. Heidari, A. Hatami, J. M. Elwell and R. D. Tingley. 2006. 'Indoor geolocation in the absence of direct path'. *IEEE Wireless Communications* 13, no. 6 (December): 50–58. doi:10.1109/MWC.2006.275198. (Cited on page 42).

Palaniappan, R., P. Mirowski, T. K. Ho, H. Steck, P. Whiting and M. MacDonald. 2011. 'Autonomous RF Surveying Robot for Indoor Localization and Tracking'. In *International Conference on Indoor Positioning and Indoor Navigation (IPIN), 2011.* Accessed 22nd June 2016. http://ipin2011.dsi.uminho.pt/PDFs/Shortpaper/49_Short_Paper.pdf. (Cited on page 48).

Pan, J. J., J. T. Kwok, Q. Yang and Y. Chen. 2006. 'Multidimensional Vector Regression for Accurate and Low-Cost Location Estimation in Pervasive Computing'. *IEEE Transactions on Knowledge and Data Engineering* 18, no. 9 (September): 1181–1193. doi:10.1109/TKDE.2006.145. (Cited on pages 47, 51).

Pany, T., J. Winkel, B. Riedl, H. Niedermeier, B. Eissfeller, T. Wörz, R. Schweikert et al. 2010. 'Experimental results from an ultra-tightly coupled GPS/Galileo/WiFi/ZigBee/MEMS-IMU indoor navigation test system featuring coherent integration times of several seconds'. In *5th ESA Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing (NAVITEC), 2010.* December. doi:10.1109/NAVITEC.2010.5708020. (Cited on page 6).

Papandrea, M., and S. Giordano. 2012. 'Enhanced Localization Solution'. In *IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, 241–246. March. doi:10.1109/PerComW.2012.6197487. (Cited on page 5).

Park, J. 2013. 'Indoor Localization using Place and Motion Signatures'. PhD diss., MIT. Accessed 16th December 2014. http://people.csail.mit.edu/jgpark/docs/phdthesis_jgpark.pdf. (Cited on page 48).

Park, J., D. Curtis, S. Teller and J. Ledlie. 2011. 'Implications of Device Diversity for Organic Localization'. In *Proc. 30th IEEE International Conference on Computer Communications (INFOCOM 2011)*, 3182–3190. http://people.csail.mit.edu/jgpark/docs/PCTL11.pdf. (Cited on page 51).

Pearl, J. 1988. *Probabilistic reasoning in intelligent systems: networks of plausible inference.* 2pr. Morgan Kaufmann. (Cited on page 15).

Pei, L., R. Chen, Y. Chen, H. Leppakoski and A. Perttula. 2009. 'Indoor/Outdoor Seamless Positioning Technologies Integrated on Smart Phone'. In *First International Conference on Advances in Satellite and Space Communications, 2009. SPACOMM 2009.* 141–145. July. doi:10.1109/SPACOMM.2009.12. (Cited on page 5).

Rappaport, T. S. 2001. *Wireless Communications Principles and Practice: Principles and Practice.* In *Prentice Hall Communications Engineering and Emerging Technologies Series,* 2nd ed. Upper Saddle River, NJ: Prentice Hall PTR. (Cited on pages 43, 44, 51).

Rasmussen, C. E., and C. K. I. Williams. 2006. *Gaussian Processes for Machine Learning.* MIT Press. Accessed 8th April 2013. www.GaussianProcess.org/gpml. (Cited on pages 30–33, 35, 36, 61, 145–147, 150).

Reyero, L., and G. Delisle. 2008. 'A Pervasive Indoor-Outdoor Positioning System'. *Journal of Networks* 3, no. 8 (November): 70–83. http://ojs.academypublisher.com/index.php/jnw/article/view/03087083. (Cited on page 5).

Richter, P., J. Seitz, L. Patiño-Studencka, J. Gutiérrez Boronat and J. Thielecke. 2012. 'Sensor Data Fusion for Seamless Navigation using Wi-Fi Signal Strengths and GNSS Pseudoranges'. In *Proceedings of European Navigation Conference (ENC), 2012.* April. (Cited on page 6).

Richter, P., and M. Toledano-Ayala. 2015. 'Revisiting Gaussian Process Regression Modeling for Localization in Wireless Sensor Networks'. *Sensors* 15, no. 9 (15th August): 22587–22615. doi:10.3390/s150922587. (Cited on page 52).

Roos, T., P. Myllymäki, H. Tirri, P. Misikangas and J. Sievänen. 2002. 'A Probabilistic Approach to WLAN User Location Estimation'. *International Journal of Wireless Information Networks* 9 (3): 155–164. doi:10.1023/A:1016003126882. (Cited on pages 47, 51).

Samama, N. 2008. *Global positioning: technologies and performance.* Wiley-Interscience. (Cited on pages 13, 36, 37, 40).

Schölkopf, B., and A. J. Smola. 2002. *Learning with Kernels.* Support Vector Machines, Regularization, Optimization and Beyond. The MIT Press. (Cited on pages 146, 147).

Shah, Z., R. A. Malaney, X. Wei and K. Tai. 2007. 'Experimental Deployment of Particle Filters in WiFi Networks'. In *IEEE International Conference on Communications, 2007. ICC '07.* 4692–4697. June. doi:10.1109/ICC.2007.775. (Cited on page 6).

Singh, R., M. Guainazzo and C. S. Regazzoni. 2004. 'Location determination using WLAN in conjunction with GPS network (Global Positioning System)'. In *IEEE 59th Vehicular Technology Conference, 2004. VTC 2004.* 5:2695–2699. May. doi:10.1109/VETECS.2004.1391410. (Cited on page 6).

skyhookwireless. 2013. 'SKYHOOK°: The worldwide leader in location positioning, context and intelligence.' Skyhook Wireless, Inc. Detailed information can be found on the press site, see articles from 30/06/2008 (XPS), 09/02/2007 (SiRF), 30/09/2008 (Broadcom), 17/11/2008 (Qualcomm), 16/02/2009 (Texas Instruments). Accessed 15th February 2013. http://www.skyhookwireless.com/. (Cited on page 42).

Sun, G., J. Chen, W. Guo and K. J. R. Liu. 2005. 'Signal processing techniques in network-aided positioning: a survey of state-of-the-art positioning designs'. *IEEE Signal Processing Magazine* 22, no. 4 (July): 12–23. doi:10.1109/MSP.2005.1458273. (Cited on page 4).

Takasu, T. 2013. *RTKLIB Manual.* Version 2.4.2. (Cited on pages 104, 111).

Thrun, S., W. Burgard and D. Fox. 2006. *Probabilistic Robotics.* MIT Press. (Cited on page 45).

Titterton, D. H., and J. L. Weston. 2004. *Strapdown Inertial Navigation Technology.* 2nd ed. Peter Peregrinus Ltd. (Cited on pages 1, 13).

Tseng, C.-H., and S.-T. Cheng. 2008. 'Location Management Scheme with WLAN Positioning Algorithm for Integrated Wireless Networks'. *Computer Communications* (Amsterdam, The Netherlands, The Netherlands) 31, no. 18 (December): 4304–4311. doi:10.1016/j.comcom.2008.07.007. (Cited on page 51).

u-blox. 2013. *u-blox 6 Receiver Description: Including Protocol Specification.* GPS.G6-SW-10018-F. (Public Release). 18th April. Accessed 20th June 2016. https://www.u-blox.com/en/product/neolea-6t. (Cited on pages 102, 111).

Vaupel, T., J. Seitz, F. Kiefer, S. Haimerl and J. Thielecke. 2010. 'Wi-Fi positioning: System considerations and device calibration'. In *International Conference on Indoor Positioning and Indoor Navigation (IPIN), 2010.* September. doi:10.1109/IPIN.2010.5646207. (Cited on pages 42, 51).

Vermaak, J., and A. Blake. 2001. 'Nonlinear filtering for speaker tracking in noisy and reverberant environments'. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, 2001 (ICASSP '01)*, 5:3021–3024. doi:10.1109/ICASSP.2001.940294. (Cited on page 100).

Wallbaum, M., and T. Wasch. 2004. 'Markov localization of wireless local area network clients'. In *Wireless On-Demand Network Systems*, vol. 2928 of *Lecture Notes in Computer Science*, edited by R. Battiti, M. Conti and R. L. Cigno, 1–15. Springer Berlin Heidelberg. doi:10.1007/978-3-540-24614-5_1. (Cited on page 51).

Wang, X., A. K.-S. Wong and Y. Kong. 2012. 'Mobility tracking using GPS, Wi-Fi and Cell ID'. In *International Conference on Information Networking (ICOIN), 2012*, 171–176. February. doi:10.1109/ICOIN.2012.6164371. (Cited on page 5).

Weyn, M. 2001. 'Opportunistic Seamless Localization'. PhD diss., University of Antwerp. (Cited on page 26).

Widyawan, M. Klepal and D. Pesch. 2007. 'Influence of Predicted and Measured Fingerprint on the Accuracy of RSSI-based Indoor Location Systems'. In *4th Workshop on Positioning, Navigation and Communication 2007*, 145–151. March. doi:10.1109/WPNC.2007.353626. (Cited on page 103).

Xu, L., S. Zhang, J. Quan and X. Lin. 2009. 'Vehicle Positioning Using Wi-Fi Networks and GPS/DR System'. In *5th International Conference on Mobile Ad-hoc and Sensor Networks, 2009. MSN '09*, 287–293. December. doi:10.1109/MSN.2009.57. (Cited on page 6).

Yamasaki, R., A. Ogino, T. Tamaki, T. Uta, N. Matsuzawa and T. Kato. 2005. 'TDOA location system for IEEE 802.11b WLAN'. In *IEEE Wireless Communications and Networking Conference, 2005*, 4:2338–2343. March. doi:10.1109/WCNC.2005.1424880. (Cited on page 42).

Yan, M., X. Yubin and C. Xiuwan. 2012. 'Wireless Local Area Network Assisted GPS in seamless Positioning'. In *International Conference on Computer Science and Electronics Engineering (ICCSEE), 2012*, 3:612–615. IEEE Computer Society, March. doi:10.1109/ICCSEE.2012.452. (Cited on page 6).

Yang, F., and D. Aoshuang. 2009. 'A Solution of Ubiquitous Location Based on GPS and Wi-Fi ULGW'. In *Ninth International Conference on Hybrid Intelligent Systems, 2009. HIS '09.* 2:260–263. August. doi:10.1109/HIS.2009.163. (Cited on page 5).

Yeh, S.-C., W.-H. Hsu and Y.-S. Chiou. 2010. 'Adaptive-Weighting Schemes for Location-Based Services over Heterogeneous Wireless Networks'. In *IEEE 71st Vehicular Technology Conference (VTC 2010), 2010.* May. doi:10.1109/VETECS.2010.5494100. (Cited on page 6).

Yeh, S.-C., W.-H. Hsu, M.-Y. Su, C.-H. Chen and K.-H. Liu. 2009. 'A study on outdoor positioning technology using GPS and WiFi networks'. In *International Conference on Networking, Sensing and Control, 2009. ICNSC '09*, 597–601. March. doi:10.1109/ICNSC.2009.4919345. (Cited on page 6).

Yim, J. 2008. 'Introducing a decision tree-based indoor positioning technique'. *Expert Systems with Applications* 34 (2): 1296–1302. doi:10.1016/j.eswa.2006.12.028. (Cited on page 47).

Yin, F., C. Fritsche, F. Gustafsson and A. M. Zoubir. 2013. 'Received signal strength-based joint parameter estimation algorithm for robust geolocation in LOS/NLOS environments'. In *Proc. IEEE Int Acoustics, Speech and Signal Processing (ICASSP) Conf,* 6471–6475. doi:10.1109/ICASSP.2013.6638912. (Cited on page 51).

Yoo, J., and H. J. Kim. 2014. 'Target Tracking and Classification from Labeled and Unlabeled Data in Wireless Sensor Networks'. *Sensors* 14 (12): 23871–23884. doi:10.3390/s141223871. (Cited on page 50).

Youssef, M., A. Agrawala and U. Shankar. 2003. 'WLAN Location Determination via Clustering and Probability Distributions'. In *Proceedings of the First IEEE International Conference on Pervasive Computing and Communications, 2003. (PerCom 2003).* March. (Cited on pages 51, 52).

Zandbergen, P. A., and S. J. Barbeau. 2011. 'Positional Accuracy of Assisted GPS Data from High-Sensitivity GPS-enabled Mobile Phones'. *Journal of Navigation* 64 (03): 381–399. doi:10.1017/S0373463311000051. (Cited on page 2).

Zirari, S., P. Canalda and F. Spies. 2010. 'WiFi GPS based combined positioning algorithm'. In *IEEE International Conference on Wireless Communications, Networking and Information Security (WCNIS)*, 684–688. June. doi:10.1109/WCINS.2010.5544653. (Cited on page 6).