

# Models and Algorithms for Ultra-Wideband Localization in Single- and Multi-Robot Systems

THÈSE N° 5746 (2013)

PRÉSENTÉE LE 7 JUIN 2013

À LA FACULTÉ DE L'ENVIRONNEMENT NATUREL, ARCHITECTURAL ET CONSTRUIT  
LABORATOIRE DE SYSTÈMES ET ALGORITHMES INTELLIGENTS DISTRIBUÉS  
PROGRAMME DOCTORAL EN INFORMATIQUE, COMMUNICATIONS ET INFORMATION

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Amanda Stella Markowska PROROK

acceptée sur proposition du jury:

Prof. J.-Y. Le Boudec, président du jury  
Prof. A. Martinoli, directeur de thèse  
Prof. V. Kumar, rapporteur  
Prof. D. Scaramuzza, rapporteur  
Dr J. Skaloud, rapporteur



ÉCOLE POLYTECHNIQUE  
FÉDÉRALE DE LAUSANNE

Suisse  
2013



# Acknowledgements

**T**HE accomplishment of this thesis stands on the shoulders of many people. I am grateful to all of those who supported me during the creation of this work, and who, in their own individual ways, have contributed to making these last few years an extraordinary time for me.

First of all, I would like to thank my advisor, Alcherio Martinoli, for all of his support during my work at the lab. His consistently constructive and dedicated approach towards directing my thesis, as well as his willingness to let me shape the course of my research, have made it a true pleasure for me to work on my projects. He has opened many doors for me over the course of my career, and I am very grateful for this.

A number of people contributed to the research that underpins this thesis. In particular, I am thankful to my close collaborators and co-authors: Alexander Bahr, an expert on collaborative localization, who guided many of the developments in this thesis, and who supported me and my work unconditionally; Phillip Tomé, an expert on localization systems, with the help of whom I was able to deepen my understanding of ultra-wideband and its potential for localization; Lukas Gonon, a gifted mathematician, who worked with me as a summer intern during a key period of my research, and who contributed to the development of the ultra-wideband model; Adrian Arfire, a talented scientist with a keen sense for detail, who worked with me as an intern and helped me with the tedious task of installing my experimental setup; and Emmanuel Droz, an engineer of the finest kind, for all his help and indispensable support in developing the hardware needed for my experiments. I am thankful to the team of the MICS-UWB project, in particular Cyril Botteron, Biswajit Mishra, John Farserotu, and Alexander Feldman, who have provided valuable scientific advice throughout the duration of this project. Finally, I am also thankful to Nikolaus Correll, Chris Cianci, Jim Pugh, and Pierre Roduit, who were senior colleagues at the lab, and provided guidance during my studies.

I have been lucky to have supervised talented and hard-working students: Yuangfang Ma, Louis Raymond, Rafael Mosberger, Dimitrios Chalikopoulos, and Christophe Schild. I am thankful for their genuine enthusiasm and for the great collaborations.

I would like to thank the members of the jury who kindly accepted to review this manuscript and provided me with valuable feedback: Jean-Yves Le Boudec, Vijay Kumar, Davide Scaramuzza, and Jan Skaloud.

## Acknowledgements

---

I am very thankful to all members of the DISAL lab, for caring, and for creating an exceptional working environment. It was a joy to come to work. Also, I would like to thank my close friends and former flatmates who share my adventures and who enrich my life—my memories of Lausanne are yours.

I am indebted to Felix Moesner, Paul Fivat, and Paul Dudler, who gave me unique opportunities during my career in Japan, and who have continued to support me.

I would like to express my deepest gratitude to my family for their boundless support; Augusta and Pirmin for the energy that they have given me; Heike and Charles for their encouragement; and especially my parents, Elza and Jacek, for their generosity, wisdom, and inspiration.

And finally, I would like to thank Sven, a truly amazing partner and enlightened scientist, for accompanying me on every part of the way, with endless patience, humor, and love.

*Lausanne, 15 May 2013*

Amanda Prorok



# Abstract

**L**OCATION is a piece of information that empowers almost any type of application. In contrast to the outdoors, where global navigation satellite systems provide geo-spatial positioning, there are still millions of square meters of indoor space that are unaccounted for by location sensing technology. Moreover, predictions show that people's activities are likely to shift more and more towards urban and indoor environments—the United Nations predict that by 2020, over 80% of the world's population will live in cities. Meanwhile, indoor localization is a problem that is not simply solved: people, indoor furnishings, walls and building structures—in the eyes of a positioning sensor, these are all obstacles that create a very challenging environment. Many sensory modalities have difficulty in overcoming such harsh conditions when used alone. For this reason, and also because we aim for a portable, miniaturizable, cost-effective solution, with centimeter-level accuracy, we choose to solve the indoor localization problem with a hybrid approach that consists of two complementary components: ultra-wideband localization, and collaborative localization. In pursuit of the final, hybrid product, our research leads us to ask what benefits collaborative localization can provide to ultra-wideband localization—and vice versa. The road down this path includes diving into these orthogonal sub-domains of indoor localization to produce two independent localization solutions, before finally combining them to conclude our work.

As for all systems that can be quantitatively examined, we recognize that the quality of our final product is defined by the rigor of our evaluation process. Thus, a core element of our work is the experimental setup, which we design in a modular fashion, and which we complexify incrementally according to the various stages of our studies. With the goal of implementing an evaluation system that is systematic, repeatable, and controllable, our approach is centered around the mobile robot. We harness this platform to emulate mobile targets, and track it in real-time with a highly reliable ground truth positioning system. Furthermore, we take advantage of the miniature size of our mobile platform, and include multiple entities to form a multi-robot system. This augmented setup then allows us to use the same experimental rigor to evaluate our collaborative localization strategies. Finally, we exploit the consistency of our experiments to perform cross-comparisons of the various results throughout the presented work.

Ultra-wideband counts among the most interesting technologies for absolute indoor localization known to date. Owing to its fine delay resolution and its ability to penetrate through various materials, ultra-wideband provides a potentially high ranging accuracy, even in clut-

## Abstract

---

tered, non-line-of-sight environments. However, despite its desirable traits, the resolution of non-line-of-sight signals remains a hard problem. In other words, if a non-line-of-sight signal is not recognized as such, it leads to significant errors in the position estimate. Our work improves upon state-of-the-art by addressing the peculiarities of ultra-wideband signal propagation with models that capture the spatiality as well as the multimodal nature of the error statistics. Simultaneously, we take care to develop an underlying error model that is compact and that can be calibrated by means of efficient algorithms. In order to facilitate the usage of our multimodal error model, we use a localization algorithm that is based on particle filters.

Our collaborative localization strategy distinguishes itself from prior work by emphasizing cost-efficiency, full decentralization, and scalability. The localization method is based on relative positioning and uses two quantities: relative range and relative bearing. We develop a relative robot detection model that integrates these measurements, and is embedded in our particle filter based localization framework. In addition to the robot detection model, we consider an algorithmic component, namely a reciprocal particle sampling routine, which is designed to facilitate the convergence of a robot's position estimate. Finally, in order to reduce the complexity of our collaborative localization algorithm, and in order to reduce the amount of positioning data to be communicated between the robots, we develop a particle clustering method, which is used in conjunction with our robot detection model.

The final stage of our research investigates the combined roles of collaborative localization and ultra-wideband localization. Numerous experiments are able to validate our overall localization strategy, and show that the performance can be significantly improved when using two complementary sensory modalities. Since the fusion of ultra-wideband positioning sensors with exteroceptive sensors has hardly been considered so far, our studies present pioneering work in this domain. Several insights indicate that collaboration—even if through noisy sensors—is a useful tool to reduce localization errors. In particular, we show that our collaboration strategy can provide the means to minimize the localization error, given that the collaborative design parameters are optimally tuned. Our final results show median localization errors below 10 cm in cluttered environments.

**Keywords:** Ultra-wideband, indoor localization, non-line-of-sight, collaboration, mobile robotics

# Résumé

**L**a localisation est une information qui valorise énormément d'applications. Contrairement à l'extérieur, où les systèmes de positionnement par satellites fournissent une couverture globale de géopositionnement, il y a encore des millions de mètres carrés d'espace intérieur qui ne sont pas équipés de technologie de localisation efficace. En outre, les prévisions montrent que les activités populaires sont susceptibles de basculer à l'intérieur—les Nations Unies prévoient, que d'ici 2020, plus de 80% de la population mondiale vivra dans les villes. Cependant, la localisation à l'intérieur est un problème qui n'est pas facilement résolu : les gens, les meubles, les murs et les structures des bâtiments sont tous des obstacles qui créent un environnement de perception hostile pour un capteur de positionnement. De nombreuses modalités sensorielles, lorsqu'elles sont utilisées seules, éprouvent des difficultés à surmonter ces conditions difficiles. Pour cette raison, et aussi parce que nous aspirons à une solution portable, miniaturisable, avec un bon rapport coût-efficacité, et avec une précision centimétrique, nous avons choisi de résoudre le problème de la localisation à l'intérieur avec une approche hybride qui se compose de deux volets complémentaires : la localisation ultra large bande et la localisation collaborative. Dans la poursuite du produit hybride final, notre recherche nous amène à nous poser la question de savoir quels avantages la localisation collaborative peut apporter à la localisation ultra large bande—et vice-versa. La réponse à cette question requiert de plonger dans ces deux sous-domaines orthogonaux de la localisation à l'intérieur pour produire deux solutions de localisation indépendantes, avant de finalement les combiner.

Comme pour tous les systèmes qui peuvent être examinés quantitativement, nous reconnaissons que la qualité de notre produit final est défini par la rigueur de notre processus d'évaluation. Ainsi, un élément essentiel de notre travail est le dispositif expérimental, que nous concevons de façon modulaire, et que nous adaptons en fonction des différentes étapes de nos études. Dans le but de mettre en place un système d'évaluation systématique, reproductible et contrôlable, notre approche est centrée autour du robot mobile. Nous tirons parti de cette plate-forme pour émuler des cibles mobiles, afin de les suivre en temps réel avec un système de positionnement vérité terrain très fiable. Par ailleurs, nous profitons de la petite taille de notre plate-forme mobile pour en inclure plusieurs et former un système multi-robots. Ce cadre augmenté nous permet ensuite d'utiliser la même rigueur expérimentale pour évaluer nos stratégies de localisation collaborative. Enfin, nous exploitons la cohérence de nos expériences pour effectuer des comparaisons croisées des différents résultats tout au long de notre travail.

A ce jour, l’ultra large bande compte parmi les technologies les plus intéressantes pour la localisation absolue à l’intérieur. Du fait de sa résolution temporelle fine et de sa capacité à pénétrer à travers divers matériaux, l’ultra large bande a le potentiel de fournir une précision élevée, même dans les zones encombrées, sans visibilité directe. Cependant, en dépit de ses traits désirables, la résolution des signaux en visibilité indirecte reste un problème difficile. En d’autres termes, si un signal en visibilité indirecte n’est pas reconnu en tant que tel, il conduit à d’importantes erreurs dans l’estimation de la position. Notre travail améliore l’état de l’art en abordant les particularités de la propagation des signaux ultra large bande avec des modèles qui prennent en compte la spatialité ainsi que la multimodalité de son erreur. Aussi, nous développons un modèle d’erreur sous-jacent qui est compact et qui peut être calibré à l’aide d’algorithmes efficaces. Afin de faciliter l’utilisation de notre modèle d’erreur multimodal, nous utilisons un algorithme de localisation basé sur des filtres à particules.

Notre stratégie de localisation collaborative se distingue des travaux antérieurs en insistant sur une solution totalement décentralisée, extensible et avec un bon rapport coût-efficacité. Dans ce but, nous utilisons une méthode de positionnement relative basée sur deux paramètres : la distance relative et le gisement. Nous développons un modèle de détection de robots qui intègre ces mesures, et est intégrée dans notre filtre à particules. En plus du modèle de détection, nous considérons une composante algorithmique sous la forme d’une routine d’échantillonnage réciproque de particules, qui est conçu pour faciliter la convergence de l’estimation de la position d’un robot. Enfin, afin de réduire la complexité de notre algorithme de localisation collaboratif, et afin de réduire la quantité de données de positionnement communiquées entre les robots, nous développons une méthode de regroupement de particules, qui est utilisé en conjonction avec notre modèle de détection.

La dernière étape de notre recherche se concentre sur les rôles combinés de la localisation collaborative et de la localisation ultra large bande. De nombreuses expériences sont en mesure de valider notre stratégie globale de localisation, et de montrer qu’une amélioration de la performance est possible en utilisant nos deux modalités sensorielles complémentaires. Puisque la fusion des capteurs de positionnement ultra large bande avec des capteurs extéroceptifs n’a guère été considérés jusqu’ici, nos études présentent un travail pionnier du domaine. Plusieurs résultats indiquent que la collaboration, même au travers de capteurs bruités, est un outil utile pour améliorer les estimations de positionnement absolu. En particulier, nous montrons que notre stratégie de collaboration donne les moyens d’obtenir une erreur de localisation minimale, si les paramètres collaboratifs sont optimisées. Nos résultats finaux montrent une erreur de localisation médiane inférieure à 10 cm dans les environnements encombrés.

**Mots-clés : ultra large bande, localisation à l’intérieur, sans visibilité directe, collaboration, robotique mobile**

# Zusammenfassung

**P**OSITIONSDATEN sind für vielerlei Anwendungen von Nutzen. Während im freien Gelände globale Navigationssatellitensysteme die Positionsbestimmung mit hoher Genauigkeit erlaubt, ist in Millionen von Quadratmeter Innenraum Lokalisierung nicht möglich. Dies ist von zunehmender Relevanz, da Prognosen darauf hinweisen, dass mehr und mehr Menschen in urbanen Räumen leben und arbeiten (gemäss einer Studie der Vereinten Nationen, werden im Jahr 2020 über 80% der Weltbevölkerung in Städten leben). Lokalisierung in geschlossenen Räumen ist ein schwieriges Problem: Menschen, Inneneinrichtung und Mauern sind für Lokalisierungssensoren Hindernisse, die ihren Gebrauch erschweren. Mit der Verwendung einer einzigen Sensormodalität sind solche Hindernisse schwer zu überwinden. Aus diesem Grund und auch weil wir nach einer tragbaren, miniaturisierbaren und kosteneffizienten Lösung streben, die die Fähigkeit hat zentimetergenau zu lokalisieren, verwenden wir einen hybriden Ansatz aus zwei sich ergänzenden Komponenten: Lokalisierung mit Ultra-Breitband Technologie und kollaborative Lokalisierung. Die zentrale Frage die sich daraus ergibt ist: Wie kann die kooperative Lokalisierung Daten, die aus dem Ultra-Breitband Lokalisierungssystem gewonnen wurden, nutzen—und umgekehrt? Um diese Frage zu beantworten, befasst sich diese Arbeit mit diesen beiden Teilbereichen. Wir entwickeln hierzu zwei unabhängige Lokalisierungsmethoden, deren Ergebnisse wir am Ende zusammenführen.

Wie bei allen Systemen, die quantitativ analysiert werden können, hängt die Qualität unserer Ergebnisse von der Gründlichkeit unseres Bewertungsprozesses ab. Daher ist der Versuchsaufbau ein zentrales Element dieser Arbeit. Unsere Versuchsanordnung ist modular aufgebaut, und ihre Komplexität wird im Laufe der Arbeit stetig erhöht. Dabei ist vor allem wichtig, dass unser System es uns ermöglicht systematische, wiederholbare und steuerbare Experimente durchzuführen. Mobile Roboter sind dafür das ideale Werkzeug und stehen daher im Mittelpunkt dieser Arbeit. Mit diesem Aufbau stehen uns mobile Marker (Roboter) zur Verfügung, deren Position wir in einem Referenzsystem in Echtzeit verfolgen können. Die kompakten Ausmasse unserer mobilen Plattform erlauben uns darüber hinaus ein System aus mehreren solcher Roboter zu bilden, wobei die Position aller Roboter im Referenzsystem zu jeder Zeit bestimmt werden kann. Dies ermöglicht eine quantitative Bewertung der kollaborativen Lokalisierungsstrategie. Die Konsistenz unserer Experimente erlaubt es uns, die Ergebnisse der Kombination der verschiedenen Lokalisierungsstrategien miteinander zu vergleichen.

Ultra-Breitband zählt zu den interessantesten der bekannten Technologien, die für absolu-

te Lokalisierung in geschlossenen Räumen gebraucht werden können. Aufgrund der hohen zeitlichen Auflösung des Signals und der Fähigkeit, verschiedene Materialien durchdringen zu können, bieten Ultra-Breitband Signale die Möglichkeit für eine sehr hohe Ortsauflösung, auch für den Fall, dass es keine Sichtverbindung zwischen dem Sender und Empfänger gibt. Doch trotz dieser wünschenswerten Eigenschaften, bleibt die Auswertung von Signalen aus sichtverbindungslosen Kanälen ein schwieriges Problem. Wenn ein Signal aus einer sichtverbindungslosen Verbindung stammt, und dieses nicht als solches erkannt wird, kann es zu grossen Abweichungen bei der Positionsbestimmung kommen. Unsere Arbeit verbessert den Stand der Technik, indem sie die Fehlerquellen der Ultra-Breitband Positionsbestimmung in einem ortsabhängigen Modell erfasst, welches auch die multimodale Fehlerverteilung berücksichtigt. Dieses Modell ist kompakt und kann durch effiziente Algorithmen kalibriert werden. Um die Nutzung unseres multimodalen Fehlermodells zu erleichtern, verwenden wir einen Lokalisierungsalgorithmus, der auf einem Partikel-Filter basiert.

Unsere kooperative Lokalisierungsstrategie unterscheidet sich von bestehenden, insbesondere durch ihre Effizienz, vollständige Dezentralisierung und Skalierbarkeit. Unsere Lokalisierungsmethode basiert auf der Messung der Position (Winkel und Entfernung) der Roboter zueinander. Hierzu modellieren wir, wie jeweils ein Roboter einen anderen Roboter detektiert. Dieses Modell integriert diese relativen Messungen und ist in unseren Lokalisierungsalgorithmus eingebettet. Zusätzlich zu dem Detektionsmodell entwickeln wir eine algorithmische Komponente, *reciprocal sampling*, die die Konvergenz der Positionsschätzung erleichtert. Um die Komplexität des kollaborativen Lokalisierungsalgorithmus zu verringern, und um die Menge der Positionsdaten, die zwischen den Robotern ausgetauscht wird, zu reduzieren, entwickeln wir eine Methode, die die Partikel bündelt. Zur effektiven Bündlung verwenden wir das Detektionsmodell.

Im letzten Schritt kombinieren wir die kollaborative Lokalisierung mit der Ultra-Breitband Lokalisierung. Zahlreiche Experimente validieren unsere gesamte Lokalisierungsstrategie: die Resultate zeigen die Verbesserungen die erzielt werden können, wenn zwei komplementäre Sensormodalitäten verwendet werden. Weil die Fusion von Ultra-Breitband Lokalisierungssensoren mit exterozeptiven Sensoren bis heute kaum Beachtung gefunden hat, stellt unsere Arbeit in diesem Bereich eine Pionierleistung dar. Unsere Erkenntnisse deuten darauf hin, dass Kollaboration, auch bei der Verwendung von verrauschten Signalen, eine nützliche Methode zur Minimierung des Lokalisierungsfehlers ist. Die Optimierung der Parameter kann den Lokalisierungsfehler weiter verringern. In unseren Experimenten sind wir in der Lage, trotz einer Vielzahl von Hindernissen, einen Lokalisierungsfehler mit einem Medianwert von unter 10 cm zu erzielen.

**Schlüsselwörter:** Ultra-Breitband, Lokalisierung in geschlossenen Räumen, ohne Sichtverbindung, kollaborative Lokalisierung, Mobile Robotik

# Mathematical Symbols

## Robot System

Notation	Description
$\mathcal{R}_n$	Robot $n$
$N_{\mathcal{R}}$	Number of robots
$\mathbf{x}_{n,t}$	State of robot $\mathcal{R}_n$ at time $t$
$M$	Number of particles
$X_n$	Set of particles belonging to $\mathcal{R}_n$
$\mathbf{x}_{n,t}^{[i]}$	State of particle $i$ of robot $\mathcal{R}_n$ at time $t$
$w_{n,t}^{[i]}$	Weight of particle $i$ of robot $\mathcal{R}_n$ at time $t$
$\mathbf{Bel}_n$	Belief of robot $\mathcal{R}_n$

## Ultra-Wideband System

Notation	Description
$\mathcal{B}_u$	Base station $u$
$\langle \mathcal{B}_u, \mathcal{B}_v \rangle$	Base station pair
$\mathbb{B}$	Set of all base station pairs
$L_u$	Event that $\mathcal{B}_u$ is in LOS
$\bar{L}_u$	Event that $\mathcal{B}_u$ is in NLOS
$P_{L_u}$	Probability that $\mathcal{B}_u$ is in LOS
$\mathcal{M}_{uv}$	Map belonging to base station pair $\langle \mathcal{B}_u, \mathcal{B}_v \rangle$
$\mathcal{M}$	Set of all maps for all base station pairs
$A_a$	Delimited area in map
$N_A$	Number of areas in map
$m_{uv}$	Mapping function
$\boldsymbol{\theta}_{uv,a}$	Vector of model parameters for base station pair $\langle \mathcal{B}_u, \mathcal{B}_v \rangle$ and area $A_a$
$\hat{\boldsymbol{\theta}}_{uv,a}$	Estimated vector of model parameters
$p_{\ln \mathcal{N},u}$	Log-normal distribution associated to base station $\mathcal{B}_u$
$\mu_u, \sigma_u$	Scale and shape parameters of log-normal distribution $p_{\ln \mathcal{N},u}$
$\hat{\tau}_{uv,n,t}$	TDOA measurement between base station pair $\langle \mathcal{B}_u, \mathcal{B}_v \rangle$ , at position $\mathbf{x}_{n,t}$
$\tau_{uv,n,t}$	True TDOA between base station pair $\langle \mathcal{B}_u, \mathcal{B}_v \rangle$ , at position $\mathbf{x}_{n,t}$
$\Delta \tau_{uv,n,t}$	TDOA error for base station pair $\langle \mathcal{B}_u, \mathcal{B}_v \rangle$ , at position $\mathbf{x}_{n,t}$
$T_{n,t}$	Set of all TDOA measurements received by robot $\mathcal{R}_n$ at time $t$
$\mathcal{T}_{uv,a}$	Set of all TDOA measurements in area $A_a$ for base station pair $\langle \mathcal{B}_u, \mathcal{B}_v \rangle$
$p$	UWB measurement model

**Relative Positioning System**

Notation	Description
$\hat{r}_{mn,t}$	Relative range measurement between robot $\mathcal{R}_m$ and robot $\mathcal{R}_n$ at time $t$
$r_{mn,t}$	True relative range between robot $\mathcal{R}_m$ and robot $\mathcal{R}_n$ at time $t$
$\hat{\phi}_{mn,t}$	Relative bearing measurement between robot $\mathcal{R}_m$ and robot $\mathcal{R}_n$ at time $t$
$\phi_{mn,t}$	True relative bearing between robot $\mathcal{R}_m$ and robot $\mathcal{R}_n$ at time $t$
$\Delta r_{mn,t}$	Relative range measurement error between robot $\mathcal{R}_m$ and robot $\mathcal{R}_n$ at time $t$
$\Delta \phi_{mn,t}$	Relative bearing measurement error between robot $\mathcal{R}_m$ and robot $\mathcal{R}_n$ at time $t$
$\Delta \psi_{mn,t}$	Orientation difference error between robot $\mathcal{R}_m$ and robot $\mathcal{R}_n$ at time $t$
$\sigma_r$	Standard deviation of relative range measurements
$\sigma_\phi$	Standard deviation of relative bearing measurements
$\xi$	Covariance matrix of range and bearing measurements
$\tilde{\xi}$	Covariance matrix of range, bearing and orientation measurements
$d_{mn,t}$	Detection data sent from robot $\mathcal{R}_m$ to robot $\mathcal{R}_n$ at time $t$
$D_{n,t}$	Set of all detection data made of robot $\mathcal{R}_n$ at time $t$
$q$	Robot detection model



# Contents

<b>Acknowledgements</b>	<b>iii</b>
<b>Abstract (English/Français/Deutsch)</b>	<b>v</b>
<b>Mathematical Symbols</b>	<b>xi</b>
<b>I Introduction</b>	<b>1</b>
<b>1 Indoor Localization Today</b>	<b>3</b>
1.1 Commercial Indoor Localization Systems . . . . .	3
1.1.1 Topologies . . . . .	4
1.1.2 Measuring Principles . . . . .	5
1.1.3 Evaluation Principles . . . . .	7
1.1.4 Existing Systems . . . . .	9
1.2 Indoor Localization in Robotics Research . . . . .	12
1.2.1 Map-Based Localization . . . . .	12
1.2.2 Beacon-Based Localization . . . . .	14
<b>2 Systematic Evaluation of Localization Systems</b>	<b>17</b>
2.1 The Mobile Robot as a Tool for Systematic Evaluation . . . . .	17
2.2 The Experimental Setup . . . . .	18
2.2.1 Our Mobile Platform: The Khepera III Robot . . . . .	18
2.2.2 Experimental Infrastructure . . . . .	20
2.2.3 An Active Tracking Board . . . . .	21
2.3 Case Study: Navigation with the Khepera III Robot . . . . .	22
2.3.1 Calibration of Odometry . . . . .	22
2.3.2 Validation: A Basic Navigation Scenario . . . . .	25
<b>3 The Scope of this Thesis</b>	<b>27</b>
3.1 A Two-Dimensional Outline . . . . .	27
3.2 Contributions of our Work . . . . .	29
<b>II Ultra-Wideband Localization</b>	<b>31</b>
<b>4 Introduction</b>	<b>33</b>
4.1 Definition . . . . .	33

## Contents

---

4.2	Motivation . . . . .	35
4.3	Challenges . . . . .	36
4.4	Experimental Setup . . . . .	37
4.5	Case study: Evaluation of a Commercial UWB System . . . . .	38
<b>5</b>	<b>Sensor Model</b>	<b>43</b>
5.1	Problem Formulation . . . . .	43
5.2	The UWB Error Model . . . . .	44
5.2.1	Preliminaries . . . . .	45
5.2.2	General UWB Measurement Model . . . . .	46
5.2.3	Efficient TDOA Measurement Model . . . . .	47
5.2.4	Evaluation of Approximation . . . . .	50
5.2.5	Cramér Rao Lower Bound . . . . .	51
<b>6</b>	<b>Algorithms</b>	<b>53</b>
6.1	General Localization Framework . . . . .	53
6.1.1	Preliminaries . . . . .	53
6.1.2	The Particle Filter . . . . .	55
6.1.3	Localization using UWB Measurements . . . . .	56
6.2	Estimation of Model Parameters . . . . .	57
6.2.1	Derivation . . . . .	58
6.2.2	Batch Estimation Algorithm . . . . .	60
6.2.3	Online Estimation Algorithm . . . . .	60
6.2.4	Performance . . . . .	61
6.3	Mapping UWB Error Models . . . . .	61
6.3.1	Online Mapping . . . . .	61
6.3.2	Offline Mapping . . . . .	66
6.4	Performance . . . . .	67
<b>7</b>	<b>Conclusion</b>	<b>71</b>
7.1	Related Work . . . . .	71
7.2	Discussion . . . . .	72
7.3	Conclusion . . . . .	73
<b>III</b>	<b>Multi-Robot Localization</b>	<b>75</b>
<b>8</b>	<b>Introduction</b>	<b>77</b>
8.1	Motivation . . . . .	77
8.2	Centralized, Multi-Centralized or Decentralized? . . . . .	78
8.3	Experimental Setup . . . . .	80
8.4	Infrared-based Relative Positioning . . . . .	81
8.4.1	Hardware Platform . . . . .	82
8.4.2	Calibration . . . . .	82
<b>9</b>	<b>Sensor Model</b>	<b>85</b>
9.1	Problem Formulation . . . . .	85
9.2	Range & Bearing Model . . . . .	87

<b>10 Algorithms</b>	<b>91</b>
10.1 Multi-Robot Localization Framework . . . . .	91
10.2 Reciprocal Sampling . . . . .	93
10.3 Particle Clustering . . . . .	95
10.3.1 Clustering Algorithm . . . . .	96
10.3.2 Cluster Abstraction . . . . .	98
10.4 Performance . . . . .	100
<b>11 Conclusion</b>	<b>105</b>
11.1 Related Work . . . . .	105
11.2 Discussion . . . . .	107
11.3 Conclusion . . . . .	107
<b>IV Ultra-Wideband and Collaboration</b>	<b>109</b>
<b>12 Introduction</b>	<b>111</b>
12.1 Multi-sensor Data Fusion . . . . .	111
12.2 Motivation . . . . .	114
12.3 Experimental Setup . . . . .	115
<b>13 Algorithms</b>	<b>117</b>
13.1 Problem Formulation . . . . .	117
13.2 Fusion Algorithm . . . . .	118
13.3 Performance . . . . .	119
13.3.1 Overall Localization Error . . . . .	119
13.3.2 Evaluation of Collaboration . . . . .	120
13.3.3 Evaluation of Heterogeneity . . . . .	122
13.3.4 Evaluation of Reciprocal Sampling . . . . .	124
<b>14 Conclusion</b>	<b>127</b>
14.1 Related Work . . . . .	127
14.2 Discussion . . . . .	128
14.3 Conclusion . . . . .	128
<b>V Conclusion</b>	<b>131</b>
<b>15 Conclusion and Outlook</b>	<b>133</b>
<b>A Analysis of Reciprocal Sampling</b>	<b>137</b>
A.1 Analysis . . . . .	137
A.1.1 Robot $\mathcal{R}_m$ is continuously localized . . . . .	138
A.1.2 Robot $\mathcal{R}_m$ only initially localized . . . . .	138
A.2 Discussion . . . . .	140
<b>B Simultaneous Localization and Mapping of UWB Error Models</b>	<b>143</b>
<b>Glossary</b>	<b>147</b>

## **Contents**

---

<b>Bibliography</b>	<b>148</b>
<b>Curriculum Vitae</b>	<b>157</b>

# Introduction **Part I**



# 1 Indoor Localization Today

*Why location matters. Indoors.*

**J**UST as the Global Positioning System (GPS) went through a revolution in the late 1990s, we are at the brink of a similar paradigm shift for a thus far underestimated environment: the indoors. From the user's perspective, localization is commonly a synonym for navigation. This may indeed be one of its major purposes—however, with the advent of sub-meter or even centimeter level accuracy, encompassing both the in- and outdoors, a whole new dimension of location-based services and applications is on the rise. It is needless to say that the focus on built and structured environments will only increase in the near future: the United Nations predict that by 2020, over 80% of the worlds population will live in urban settings [123], with a large shift of activities to the indoors. Indoor localization is predictably a game-changing technology.

## 1.1 Commercial Indoor Localization Systems

More than a decade ago, Werb et al. [127] observed that "*millions of square meters of indoor space are out of reach of Navstar [positioning] satellites.*" It was at about that time that indoor localization was recognized as an enabling technology, with potential use cases spanning all sectors (commercial, public, and military). Commercial applications have already begun to embed a localization component in tasks such as inventory tracking, asset management and assembly control. Manufacturing industries (such as automotive and aerospace) are able to optimize resource allocations and thus benefit by reducing cycle times and increasing their throughput. People tracking is an emerging technology for commercial applications, where it is necessary to keep track of worker positions in automated, hazardous environments [79]. With a majority of today's population living in urban areas, the military domain benefits from accurate localization to track soldiers in Global Positioning System (GPS)-denied environments. Disaster response actions have similar needs, where successful localization of fire-fighters or rescue workers may be life-saving [72]. Straight forward usage of indoor localization services can also be found in the recreation and entertainment sector. Large restaurants can use the technology

to locate customers at a table when an order is ready. Similarly, large commercial areas (such as shopping malls) can take advantage of positioning to provide location-based services such as targeted advertising. Finally, indoor localization systems are also useful for academic research purposes, particularly in the domains of ubiquitous computing projects [103] and ad hoc sensor networks [111].

These incentives have, in recent years, intensified the development of systems and methods specific to indoor localization. In particular, terminology was defined [49], goals were set [127], and challenges were identified [84]—the groundwork was laid for a rich body of scientific work to follow.

### 1.1.1 Topologies

Stated simply, a localization system is composed of two components: emitters and receivers for a given sensing modality. To date, a number of sensing modalities have been adopted for positioning systems [49, 70]. These positioning systems vary in functionality as well as performance, due to the underlying physical constraints as well as the chosen implementations. Although the terms *emitter* and *receiver* relate directly to Radio Frequency (RF)-based systems, and RF-based systems do indeed represent the majority of localization systems, we shall use them in a larger sense:

An *emitter* is a device or object which emits a signal, either due to an inherent activity, or due to the capacity of reflecting a certain type of signal. A *receiver* is a device which receives the emitted signals and has the computational capacity to process them.

In terms of architecture, localization systems can be classified into two main domains [28]: remote positioning and self-positioning systems. Remote systems possess mobile signal emitters and fixed or mobile base stations that receive the emitters' signals. The position estimate is computed off-board (remotely), on a master station. Self-positioning systems possess mobile signal receivers that compute their own location on-board, based on measured signals. We note that there are also systems that combine receivers and emitters on-board the same device—the underlying positioning principle can then be either remote or self-positioning.

**Remote positioning** systems possess mobile signal emitters and fixed or mobile base stations that receive the transmitters' signals. The position estimate is computed off-board, on a master station.

**Self-positioning** systems possess mobile signal receivers that compute their own location on-board, based on measured signals. The emitters are located at known positions (either fixed or mobile).

**Indirect positioning** systems use an additional wireless data link that provides the position estimate to the remote side, in the case of self-positioning systems, or the 'self' side, in the case of remote positioning systems. One refers to those two sub-systems as *indirect remote positioning* systems and *indirect self-positioning* systems.



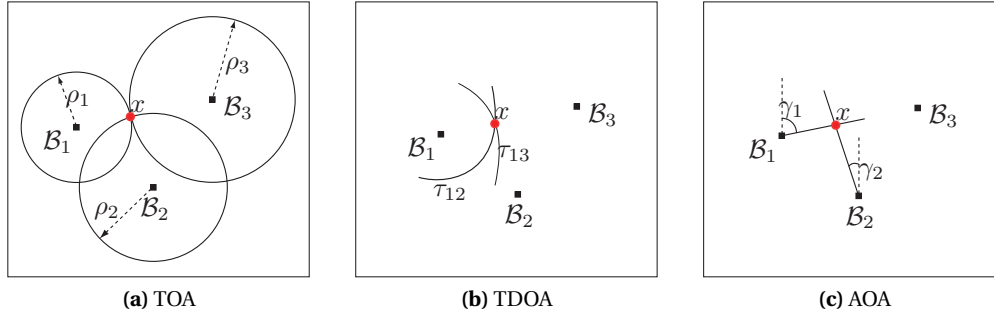
The choice of topology is often motivated by the end application and the constraints of the mobile platform. A mobile node may not have enough computational resources, or may be too small to carry the receiver payload. Conversely, a mobile node may need to run applications that are tightly coupled to the position estimate, hence the option of collocating both processes on the same device (GPS receivers that are embedded on mobile devices constitute an example of self-localization). In some cases, the combination of constraints (computational limitations plus tight coupling with other processes) may motivate an indirect architecture, which exploits an additional communication link between the emitter and receiver nodes.

### 1.1.2 Measuring Principles

For us to better understand the various localization systems, it is helpful to review the underlying measuring principles. Previous works [49, 70] suggest a classification of all common methods into three main domains: triangulation, scene analysis, and proximity.

**Triangulation** considers computations that rely on the geometric properties of triangles. Triangulation itself has two alternate implementations: lateration and angulation. In the case of lateration, the computations are based on range quantities. In the case of angulation, the computations are based on angle quantities. There are a number of ways to obtain range measurements from an emitter-receiver pair. Figure 1.1 illustrates three common methods. With the Time of Arrival (TOA) method, the one-way propagation time of a signal emitted by the target is measured, and the distance calculation exploits the fact that this time is directly proportional to the traveled distance. The resulting target position is then equal to the intersection of the circles / spheres that represent the distances to each of the base stations. In the case where no unique intersection exists (due to range errors), least-squares algorithms are often implemented to resolve the position estimates. With the Time Difference of Arrival (TDOA) method, the time difference between two received signals is measured. The target then lies on a hyperbola defining a constant range difference between two signal receivers. Received Signal Strength (RSS) methods measure the path loss of a signal, i.e., the attenuation of the signal due to its propagation in space. With the Roundtrip Time of Flight (RTOF) method, the range is calculated as for TOA, but where the target will only emit a signal upon a request from the receiver. The receiver base stations then apply the round-trip time (used for the signals there and back) to calculate the range. The Phase of Arrival (POA) method will consider the phase offset of received signals to determine the traveled distance. Algorithms for POA may be implemented directly as for TOA, or when considering phase differences between receivers, as for TDOA. Finally, Angle of Arrival (AOA) is an angulation rather than lateration technique, with the advantage that only two (instead of three) measuring units are needed in a 2D scenario and no system synchronization is required. The target position is found by intersecting the direction lines.

The aforementioned methods have advantages as well as disadvantages. In particular, factors such as Non Line of Sight (NLOS) or multipath may influence their performances, more or less strongly. For TOA, both transmitters as well as receivers need to be synchronized precisely. The TDOA method relaxes these synchronization constraints by considering only the clocks of the receiving stations. Yet, both TOA and TDOA are sensitive to NLOS propagation. RSS is less sensitive to NLOS, but is, however, affected by site-specific signal attenuation due to multipath. The RTOF method alleviates the problem of clock synchronization, but is burdened



**Figure 1.1:** Examples of positioning techniques.  $\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3$  represent base stations, and the emitter node is located at  $x$ . (a) range measurements,  $\rho_1, \rho_2, \rho_3$  (b) TDOA measurements  $\tau_{12}, \tau_{13}$ , and (c) AOA measurements  $\gamma_1, \gamma_2$ .

by the difficulty of determining the response delay of the target. POA will require direct Line of Sight (LOS) signaling to avoid ambiguous phase shifts. Finally, although AOA needs only two base-stations for a 2D system, it builds upon complex hardware, is sensitive to multipath and shadowing, and will degrade when the mobile targets are far from the measuring units.

With **scene analysis**, or location fingerprinting, the characteristics of a scene are collected a priori, and later, during a run-time localization exercise, matched with measurements to find the most accurate position estimate. It is mostly employed for RSS-based systems. The technique consists of two phases. First, in an offline stage, a database is constructed consisting of measured signal strengths and corresponding coordinate tags. In the second stage, measurements are taken online. Using a dedicated pattern recognition or classification algorithm, which is trained on the previously collected values in the database, a position estimate is made. Common algorithm choices include Bayesian modeling, k-nearest neighbor, neural networks, support vector machines, and SMP (smallest m-vertex polygon).

The method of **proximity** provides a symbolic type of positioning and relies on a dense infrastructure of detection units. The moving target is collocated with its nearest detection point, or in the case of multiple detecting points, with the one that receives the strongest signal. Underlying technologies include Infrared (IR) and Radio Frequency Identification (RFID). The application of proximity is also common in mobile cellular networks, where cell-identification or cell-of-origin methods are used to infer the approximate location of a mobile handset.

**Triangulation:** The positioning computations are based on the geometric properties of triangles. Underlying measurement principles include TOA, TDOA, AOA, RTOF, POA, and RSS.

**Scene analysis:** Also known as location fingerprinting, this method requires the characteristics of a scene to be collected a priori. During the run-time localization exercise, measurements are matched with database entries to find the best position estimate.

**Proximity:** The mobile target is symbolically collocated with the nearest detection point. Common implementations include cell-identification or cell-of-origin methods in mobile cellular networks.

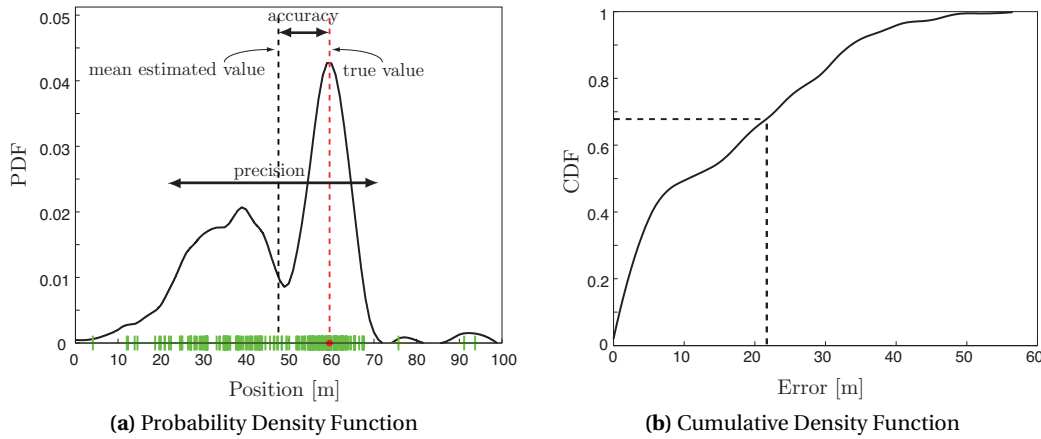
### 1.1.3 Evaluation Principles

In their early work, Tekinay et al. [115] went beyond common performance evaluation criteria to define a more comprehensive analysis for wireless geolocation systems. In particular, they discuss the shortcomings of considering accuracy as the only measure of performance. Liu et al. [70] extend this work by providing an analogous set of quality guidelines for indoor wireless localization systems. We will propose criteria along the lines of those presented in [70], grouping them into three more general categories: *(i)* accuracy and precision, *(ii)* scalability and coverage, and *(iii)* complexity and cost.

While Tekinay et al. defined **accuracy** as an area around the true location within which a certain percentage of position estimates lie, Liu et al. look at accuracy and precision separately. Statistically, accuracy measures the Euclidean distance error between the mean of the estimated values and the true value, for all estimates. However, since it only considers the mean of the estimates, this measure alone fails to evaluate the quality of the positioning technique. By adding a **precision** measure, we are also able to consider the consistency and reliability of the method. The precision is considered as the spread of all position estimates. More generally, accuracy and precision can also be related by considering the distribution of the distance errors. Since these errors can be distributed arbitrarily, it is useful to compute the Cumulative Density Function (CDF) and interpret the errors as follows: by considering a given error bound, we can determine the percentage of measurements that are made within this bound. Let us consider Figure 1.2. Figure 1.2a shows a one-dimensional positioning scenario, with position estimates scattered along the x-axis, and a probability density function characterizing the occurrence of a given position estimate. The panel also shows the true position and the mean estimated position. Figure 1.2b shows the cumulative probability density of the positioning errors. As an example, we see that 67% of the errors are below 22 m. By analogy, we note that some positioning manufacturers use the concepts of accuracy and precision by referring to the Circular Error Probability (CEP), a term originating from military ballistics, which measures the radius of a circle centered around the true position within which 50% of the position estimates lie.

In particular for applications with a large scope, it is important to also consider **scalability** issues. Scalability addresses both the **coverage** in terms of geographic space, as well as the density of mobile targets per unit of space and time. The performance of positioning systems tends to deteriorate over large distances, thus the challenge of building long-range systems. Also, when the density of mobile targets increases, the system may be strained by the additional computational and communications overhead. Optimally, a given system is able to handle large-scale scenarios. Potential solutions include a decentralization of the computational load, or an accommodation of additional infrastructure. However, certain infrastructure may not scale easily (such as the AOA method), and in such cases, completely new solutions may have to be considered. Finally, a system which is constructed to work in two-dimensional space only, may not necessarily be easily extendable to three dimensions.

When optimizing the performance of a system, clear trade-offs need to be made as a function of **complexity** and **cost**. Several questions arise: What is the allocated budget? What kind of infrastructure can be accommodated? What is the maximum computational burden that the emitters or receivers can carry? How large can the devices be? These questions are addressed by the constraints of a localization system in the context of the site where it is to be used. A



**Figure 1.2:** Example of accuracy and precision for a one-dimensional positioning scenario with 170 randomly sampled position estimates. (a) Probability density function of the position estimates. The depicted accuracy is 11 m. The precision can be calculated by taking the standard deviation of all position estimates, and amounts to 15 m (b) Cumulative density function of the position errors. The panel shows how 67% of the errors are below 22 m

remote positioning system is able to carry a higher computational load than a self-positioning system, because the computations are performed on a fixed station, which usually has access to a power line, and is not limited in size and weight due to its static position. Running complex algorithms on mobile targets is particularly challenging due to the unpredictability of how much energy the computations will draw: energy contributes significantly to the cost factor over time. The update frequency requirements also considerably contribute to the overall complexity, as higher processing speed and larger communication bandwidth are required, and the overall power consumption is increased. In designing an infrastructure, the scale will influence the complexity of certain types of systems more than it will influence others. For instance, in TDOA systems, receiver stations are mutually synchronized, which means that additional infrastructure (e.g., cables) has to be provisioned over longer ranges. On the other hand, for RSS-based scene analysis, a larger scope entails an increased fingerprinting effort, proportional to the increase in area / volume. Generally, the complexity of a system is influenced by the effort that needs to be put into its calibration and its maintenance, and is exacerbated by its sensitivity to environmental circumstances.

**Accuracy and Precision:** The accuracy measures the bias or offset of the position estimates to the true value, and the precision measures the reliability and consistency with which the estimates are made.

**Scalability and Coverage:** This criteria measures the how easily the system scales in terms of size (area and volume), as well as in terms of the density of localization targets.

**Complexity and Cost:** Infrastructure, energy consumption and maintenance are factors that influence the complexity and cost of a system. High complexity implies increased cost and increased human involvement to ensure the functionality of the system.

### 1.1.4 Existing Systems

The surveys performed by Liu et al. [70] and Hightower et al. [49] present a comprehensive overview of commercial indoor localization systems available at the respective dates of publication (2007 and 2001, respectively). This section will refer to some of the systems pointed out in the those works, extend the list with more recent implementations, and emphasize the underlying technologies. Later, we will put the systems into a global, comparative context by referring to the evaluation criteria introduced above (see Figure 1.3 for the comparison).

A system that is based on satellite technology and has global coverage is termed a Global Navigation Satellite System (GNSS). One fully operational GNSS instance is **GPS** (another is GLONASS). GPS satellites transmit ranging codes on two carrier frequencies: one at 1575.42 MHz (called L1), and a second at 1227.60 MHz (called L2). The main challenge of implementing GPS-based systems for the indoors lies in overcoming the signal weakness due to penetration through building structures. In 2006, u-blox<sup>1</sup> first released their SuperSense technology, which has now become the default technology for their high-sensitivity GPS modules and chips. The modules' tracking, navigation and reacquisition sensitivity is reported at -160 dBm, and the performance is reported with a CEP of 4 m. Assisted-GPS (or A-GPS) modules became popular in the early 2000's. The main merit of A-GPS devices is the acceleration of the time it takes for a GPS receiver to calculate its position. The method relies on an additional network-based infrastructure, which provides the receivers with the satellite information (almanac and ephemeris) needed to proceed with the location computations—the download of this information is normally serviced by the GPS satellites themselves, but because transmission rates are limited to 50 bits/s, this takes a long time. As a consequence, this technology facilitates the usage of GPS where signals are weak, since data-heavy operations can be outsourced to the A-GPS servers. Locata Corporation<sup>2</sup> provides a technology which relies on a networked system of ground stations that are precisely synchronized and transmit GPS-like signals (essentially replicating a terrestrial GPS system). The technology uses carrier-phase for positioning, and uses the GPS L1 frequency as well as the 2.4 GHz Industrial, Scientific, and Medical (ISM) band. Tests have shown that in a clear, obstacle-free indoor environment, errors range from centimeters to a few meters [105].

**RFID** systems are composed of three main elements, RFID readers, RFID tags, and one or several servers. The servers' role is to collect tags' identities, which are sent by the readers, and apply localization algorithms. The tags and readers use RF electromagnetic waves to transfer data. There are several bands, of which the following four are the most common: LF (125 kHz), HF (13.56 MHz), UHF (433, 868–915 MHz), and microwave frequency (2.45 GHz, 5.8 GHz). Tags can be either active or passive. Passive tags typically do not have an internal power supply, and have a limited range (around 1 m). Active tags possess an internal power source which enables them to emit RF signals. As a consequence, potentially large ranges (tens of meters) can be serviced. According to the comprehensive comparison of RFID localization systems by Bouet et al. [14], active RFID tags are the more popular choice. The performance of RFID systems varies a lot, since it depends on the density of the reader network (amongst other things). Yet, in general, the comparison points out that improved performance is achieved with systems based on scene analysis including additional help from so-called reference tags.

---

<sup>1</sup><http://www.u-blox.com>

<sup>2</sup><http://www.locatacorp.com/>

Localization over **cellular** networks is particularly interesting, because infrastructure (such as the one provided by the GSM standard) includes wide-spread deployments that essentially span the globe. Since the location is inferred by the cell ID (which identifies coverage sectors), typical accuracy lies in the range of 50-200 m, depending on the density of the antennas. However, in a relatively recent work, Varshavsky et al. [124] show that it is possible to achieve errors up to a few meters only (including the detection of the correct building floor). Their main innovation includes a scene analysis method which considers *wide* signal-strength fingerprints, reading from the six strongest GSM cells and from up to 29 additional GSM channels. Indeed, the additional channels are strong enough to be detected but too weak to be used for communication.

Similar to the cellular-based methods above, **Wireless Local Area Network (WLAN)** is a wide-spread technology, with infrastructure in place in most modern indoor environments. WLAN operates on the ISM 2.4 GHz frequency band. One of the earliest implementations of a WLAN-based localization system was done by Bahl et al. [4]. Based on underlying RSS measurements, the authors suggest two approaches towards solving the localization problem. The first method consists of a first, offline data collection phase (termed by the authors as an *empirical* method, but today commonly referred to as *fingerprinting*), and a second, online phase where signals are best matched to the values in the database to determine location. The second approach consists of radio signal propagation modeling. Tests reported an accuracy of roughly 3 m (for the empirical method, and higher error values for the model-based method). Subsequent works followed this line of research, focusing in particular on variant fingerprinting methods [9, 129].

**Bluetooth** operates in the same, 2.4 GHz frequency band as WLAN. Even though its range is quite short (10-15 m), its advantage is its high ubiquity due to low cost, small size, and low energy consumption: to date, it has been embedded in a large variety of personal mobile devices. The currently dominant commercial system using Bluetooth technology is called Topaz, and is serviced by Tadlys<sup>3</sup>. The company has reported an accuracy of roughly 2 m (for a latency of 10-30 s).

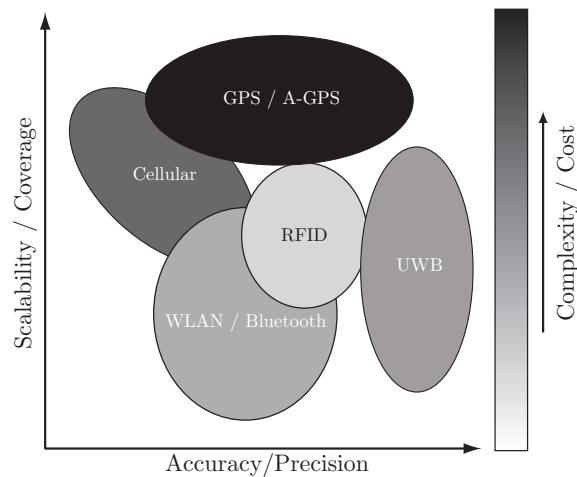
The main difference between **Ultra-Wideband (UWB)** and other radio frequency signals is that UWB transmits its signal over multiple bands of frequencies simultaneously, in the interval of 3.1 to 10.6 GHz, exceeding a bandwidth of 500 MHz or 20% of the center frequency. In the time-domain, this essentially means that very short pulses are emitted (with a duration in the order of nanoseconds). Additionally, UWB systems typically run on very low duty cycles, and thus are very low power. In the context of localization, UWB features a potentially high positioning accuracy (due to the fine time resolution of the emitted signals) and high material penetrability (due to the large bandwidth). Time Domain's<sup>4</sup> PulsOn system uses two-way time-of-flight (receiver and emitter are combined on the same module) and operates with a bandwidth of 3.1-5.3 GHz. It has a 2 cm accuracy in LOS and 0.5 m in NLOS, a maximum update rate of 100 Hz, and a maximum range of 350 m. Ubisense's<sup>5</sup> Series 7000 sensors and tags use TDOA and AOA combined, and operate with a bandwidth of 6-8 GHz. A constellation of four sensors have a maximum coverage of 400 m<sup>2</sup> and provide a reported average accuracy of 15 cm. The emitter tag has a maximum update rate of 33 Hz. Further companies have

---

<sup>3</sup><http://www.tadlys.co.il/>

<sup>4</sup><http://www.timedomain.com/>

<sup>5</sup><http://www.ubisense.net/>



**Figure 1.3:** Overview of positioning technologies, as a function of our evaluation criteria: accuracy/-precision, scalability/coverage, and complexity/cost. An ideal system would be located at the top right corner of the chart and colored in white.

developed UWB solutions (i.e., Zebra Technologies<sup>6</sup> and Aetherwire and Location<sup>7</sup>), but do not provide any details about performance.

**Hybrid technologies** have also been developed: a combination of RF and IR has become very popular, in particular for asset management in hospitals<sup>8 9</sup>. The two media complement each other: low-power RF allows for approximate localization and basic communication, whereas IR allows for more precise localization within the rooms. The advantage of this composite method is that it is cheap, effective, extremely scalable (over tens of thousands of tags), and easy to maintain.

Figure 1.3 shows an overview of positioning systems grouped according to the underlying media/technique, and plotted as a function of our first two evaluation criteria (see Section 1.1.3): accuracy/precision and scalability/coverage. The grayscale axis represents our third criteria: complexity/cost. An ideal positioning system would be located in the upper right corner of the plot, colored in white, meaning that it is very scalable, very accurate, and not overly complex and costly (easy to maintain and low power consumption). UWB provides unprecedented accuracy, whereas the scalability of GPS-based systems is currently unbeatable. RFID remains the simplest and cheapest of all methods. Finally, despite the fact that cellular and WLAN-based methods rely on relatively complex systems and do not scale as well as some of the other methods, the infrastructure is often already in place for primary usage purposes (such as internet access and telephony). The installation and maintenance costs are thus significantly reduced.

<sup>6</sup><http://www.zebra.com/gb/en.html>

<sup>7</sup><http://www.aetherwire.com>

<sup>8</sup><http://www.versustech.com/>

<sup>9</sup><http://www.radianse.com/>

### 1.2 Indoor Localization in Robotics Research

Since the 80's, the robotics community has contributed significantly to research in the domain of localization, primarily for indoor environments, but increasingly also for the outdoors. However, one could argue that the original motivation for solving the localization problem for autonomous mobile platforms lay embedded in other problems, such as navigation and control, but which needed localization information as an elementary building block. In the meantime, the localization problem has become so important that it has evolved to a stand-alone field within robotics.

The most basic ingredient of mobile robot localization is **dead-reckoning** [114]. This is a positioning method which uses the robot's local reference frame to incrementally update its position based on differential movements. As the name suggests, this method is a 'blind' method, and if not corrected at regular checkpoints, will produce a run-away error. Such checkpoints are provided by position calculations with respect to an external reference frame. In a large sense, there are two ways in which a robot can localize with respect to an external reference frame (illustrated in Figure 1.4): (i) by perceiving features of a structured environment and utilizing a map, or (ii) by relating geometrically to sensor beacons with known positions. Using sensor models and an internal representation of the surroundings, robots perform a localization cycle which continuously updates their estimated positions [114]. Elements of the environment, or landmarks, act as reference points and, upon detection, enable the robot to determine its position with respect to the latter. In the first case (i), the external reference frame is provided by landmark positions contained in the map, whereas in the second case (ii), the external reference frame is provided by the beacon positions. Indeed, we could possibly view (ii) as a sub-class of (i). Leonard et al. [67] state the following:

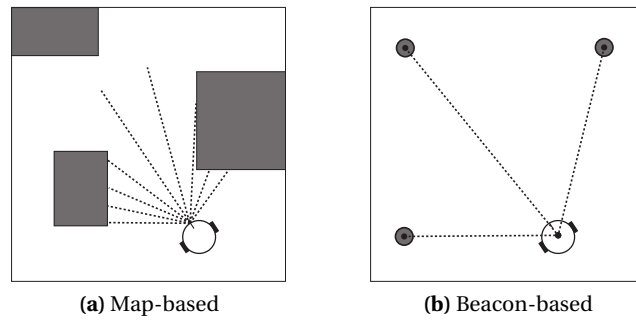
*A geometric beacon is a special type of target that can be reliably observed in successive sensor measurements and that can be accurately described in terms of a concise geometric parametrization.*

Although the work in [67] considers beacons as naturally occurring landmarks, we will consider that the beacons themselves are endowed with sensing technology. In other words, the beacons comprise an integral element of the robot's complete sensing mechanism, but are physically not part of the robot. In other words, map-based localization forces the robot to carry all sensors on-board, whereas beacon-based localization allows some of the technology to be outsourced to the external infrastructure, off-board the robot. In this manuscript, we shall also make the difference between *beacons* and *base stations*, where beacons embed active sensor technology (emitters), and base stations embed passive sensor technology (receivers).

#### 1.2.1 Map-Based Localization

Map-based localization consists of providing the robot with a model of the environment. During run-time, the localization algorithm's task is to identify landmark features in the robot's observation. If a landmark is identified, it is then matched against the landmarks registered in the map. When a correspondence is made, the robot is able to update its position with respect to the stored landmark position. Various map representations exist. They vary in their type (continuous / discrete / topological), and also in their degree of detail (complete CAD





**Figure 1.4:** One can classify the approaches towards solving the robot localization problem into two main fields: (a) localization based on a map of the environment, and (b) localization based on sensor beacons.

model of the environment vs. a simple graph of interconnections between the landmarks). When a map is not available a priori, the problem generalizes to Simultaneous Localization and Mapping (SLAM) [29]. Indeed, mapping and localization employ the same set of sensors. In this section, we will briefly review two groups of sensors which have become standard choices for map-based localization: vision-based sensing and active ranging. We note that our summary only reflects what we deem most relevant in the context of this manuscript, and a more complete overview of sensors used for map-based localization can be found in [31, 114].

Possibly the most powerful sensor for feature-based localization is the digital **camera**. DeSouza et al. [27] present a comprehensive survey of vision-based localization for mobile robots, referring to the previous decades of development in the area. Camera sensors are very popular for mobile robots since they are passive sensors, simultaneously providing rich information about the environment. Recent works have demonstrated successful navigation of autonomous vehicles using a single camera as the only sensor [12, 37]. The art of mastering vision-based localization algorithms can be divided into two main domains: feature extraction (which belongs to the field of image processing), and place recognition (which belongs to the field of computer vision). A recent review of vision-based localization for mobile robots is available in [114]. Although progress is constantly being made in this domain, issues such as variable background illumination, clutter, and dynamic sceneries can significantly affect performance. Recently, cameras are being built that work in conjunction with complementary, active range sensing with the goal to provide three dimensional image data (in real time). As an example, in 2010, Microsoft released a new consumer-grade range sensor called Kinect<sup>10</sup>. It was originally designed for user interactions in computer games. However, due to its powerful design, it has attracted significant attention in the localization and mapping community, especially within robotics [30, 32]. The sensor consists of an IR laser emitter, an IR camera, and an RGB camera [36]. It functions by capturing depth and color images simultaneously, and returns a depth image composed of a colored point cloud. The sensor has been shown to produce best results when used within a range of 1-3 m to the obstacles, and, when carefully calibrated, with a precision comparable to that of a laser range finder [60]. A similar product is being produced by Mesa Imaging<sup>11</sup>. Their so-called time-of-flight-camera possesses an IR light emitter, and uses an internal sensor composed of tens of thousands of pixels that each measure the time-

<sup>10</sup><http://www.microsoft.com/en-us/kinectforwindows/>

<sup>11</sup><http://www.mesa-imaging.ch>

of-flight of the returning (reflected) signals to produce high resolution three-dimensional imagery.

Active ranging provides a second pillar for mobile robot localization, and is comprehensively addressed in [119]. **Laser range finders** are time-of-flight sensors that provide range estimates based on the time needed for a light beam to reflect off a target and return. The sensors are comprised of moving parts, i.e., a mechanical design that allows a mirror to sweep the light beam in order to cover the whole area of interest. Implementations exist in 2D as well as 3D [117]. The advantage of laser range finders is their accuracy and precision, owing to the underlying medium of laser light, and their relatively long range of up to 80 m [114]. The disadvantages of this sensor are its heavy weight, high cost, and high energy consumption.

Low-cost alternatives to laser range finders include ultrasound sensors and IR proximity sensors. The **ultrasound sensor** is, much like the laser range finder, a time-of-flight sensor: it emits a pressure wave and provides a range estimate based on the time needed for the signal to reflect off a target and return. It is less precise than laser range finders, nevertheless yielding a resolution of approximately 2 cm. Its maximum range is commonly around 5 m. Ultrasound sensors are very cheap, can be significantly miniaturized, and consume very little power. However, the sensor has a few down-sides. One is that it suffers from cross talk with other ultrasound sensors, and, when installed in ring constellations, must be serviced sequentially to minimize interference. Another disadvantage is that it is sensitive to the surface material of the targets: some materials poorly reflect the ultrasound waves, and thus fail to return signals that are strong enough. Finally, due to its inherent time-of-flight measuring principle, the maximum frequency is limited by the operational range. For instance, for a range of 3 m, the maximum frequency is limited to 56 Hz (assuming a propagation speed of 340 m/s and a round-trip distance of 6 m).

Active **IR proximity sensors** work by emitting a pulse of infrared light, and measuring the energy returned by the reflection off nearby obstacles. Whereas infrared is at the lower end in terms of precision, it is very good at asserting if an obstacle is present or not, particularly in very cluttered and dense areas. Its maximum range can vary, but is usually in the range 1-3 m. The early work by Flynn [34] shows how infrared and ultrasound are successfully combined to take advantage of their complementary nature.

Static and extremely low-frequency magnetic fields in buildings arise from both natural sources (the Earth) and manmade sources (electric and electronic appliances, building structures). These fields can be sensed by **magnetic sensors** [65]. In order to localize, the magnetic fields in a given space are profiled a priori to create maps with measurement signatures—if the maps exhibit sufficient variability, unambiguous localization can be achieved. The approach in [47] solves a one-dimensional localization problem, for a mobile robot that travels within corridors, and reports an accuracy in the range of 0.1-3.5 m.

### 1.2.2 Beacon-Based Localization

Compared to map-based localization, beacon-based localization provides a very convenient approach, as some of the software and hardware complexity can potentially be outsourced to the beacon infrastructure. Ultimately, this type of architecture addresses a much broader class of mobile platforms, which may range from actual robots (as above) of various sizes and

computational capacities, to handheld or embedded miniature devices. The underlying measuring principles relate directly to the ones employed on commercial systems, as introduced in Section 1.1.2. The robotics community has adopted a number of these methodologies and adapted them to specific scenarios.

One of the earliest beacon-based systems includes the work by Kleeman [61]. Kleeman devised an infrastructure of **ultrasound emitter beacons**, which are geometrically arranged and spatially spread out over the given space. The beacons emit ultrasound pulses sequentially, and are controlled by a centralized transmitter station to work in conjunction with a ring of ultrasound receivers mounted on the mobile platform, so that the time of firing is known to the receiver modules. A similar system (dubbed Cricket) is presented in [87]. Here, the beacons emit a short RF signal as well as an ultrasound pulse. The mobile receiver then takes the difference of the RF and ultrasound signal-arrival-times to determine the nearest beacon. The authors report a maximum granularity of about 1.2 m.

The work done by Hähnel et al. [45] studies the usage of **passive RFID technology** in mobile robot localization. The authors develop a sensor measurement model and show how to map the locations of passive RFID tags in an environment, given a robot carrying an active RFID reader. The measurement model is also used to show that successful localization can be achieved (using a map of tag positions). Park et al. [85] develop a similar system, but base their methodologies on a very dense grid of RFID tags (with a spacing of 34 cm). Their system shows average localization errors in the order of 10 cm.

Raghavan et al. [102] develop a basic **bluetooth** localization system based on RSS measurements. A mobile robot repetitively queries the bluetooth emitter beacons with known positions, and infers its position via trilateration. Their experimental work shows that accuracies in the order of 50 cm are attainable.

Ladd et al. [64] were one of the first authors to use **WiFi** for mobile robot localization. The proposed method uses RSS measurements and is divided into two phases. The first phase is a training phase, where a sensor map of the environment is built by gathering RSS samples at checkpoint positions within the environment. The second phase is the actual localization phase, and utilizes the calibrated probabilistic models to infer position estimates for given measurements. Subsequent works [11, 54] use the same general approach. For all the above studies, the reported error was in the order of 1 m.

One of the earliest works embedding **UWB** in a robotic setting was performed by Jourdan et al. [57]. Their experimental setup employs a mobile agent carrying an UWB emitter, one real receiver station, and one simulated receiver station. The authors demonstrate a performance of roughly 50 cm accuracy in an office environment. The study performed by Gonzalez et al. [40] develops a probabilistic model for UWB range measurements. Their setup consists of a mobile robot carrying an UWB transceiver and three fixed UWB transceivers. Short experiments show the feasibility of obtaining high precision in clear environments (as good as 5 cm). Segura et al. [112] develop a self-positioning system based on UWB TDOA measurements. Their system is composed of an UWB receiver board mounted on the robot, and three external UWB emitter beacons at fixed locations. The positioning accuracy is tested (statically) at five different locations in the experimental space, with errors in the order of 20 cm. Lastly, in a recent study, Hollinger et al. [50] use UWB range data from five fixed base stations to track a mobile robot, which carries an UWB emitter. The experimental tracking scenario tests ranging through walls,

and does not rely on the fusion of additional odometry measurements for localization. The final results show room-level accuracy.

Finally, motion capture systems based on **camera** technology are becoming increasingly popular in robotics, in particular in the domain of aerial robots [51, 77]. The two main system types are classified by the markers which are attached to the target objects: active markers consist of pulsed IR emitting diodes, and passive markers are coated with a reflective material. The installation usually includes at least eight cameras, and, in the case of passive markers, also includes infrared emitting diodes which are mounted around the camera lens. Since the camera lens is covered with an IR pass filter, only the markers are detected. The detected points from all cameras are then used to calculate 3D positions via triangulation. The advantages of such systems are that they are extremely precise (sub-centimeter), and very fast (over 300 Hz). Additionally, in the case of passive markers, the localization system is completely off-board the mobile targets—an ideal trait for platforms that cannot be burdened with additional payload. The down-sides are that the system is limited to fairly small environments, and that it requires manual installation and calibration.

### Summary

In this chapter, we looked at some of the basic concepts of indoor localization systems. Two main topologies exist, self-positioning and remote positioning, and three main measuring principles are commonly applied, triangulation, scene analysis and proximity. Various media can be considered as the underlying physics of the system, however, we have seen that RF-based systems represent a majority. Our three evaluation criteria helped us classify the systems and provided a more general understanding of the state of the art. Robotics-based approaches tackle localization either by using maps of the environment, or by relying on an external infrastructure (beacon-based approach). Although map-based approaches are self-contained, beacon-based approaches provide the opportunity to outsource some of the hardware and software complexity to an external, fixed location.

## 2 Systematic Evaluation of Localization Systems

**T**HE quality of a product is defined by the rigor of its evaluation process. A thought-through evaluation will, eventually, lead to a better product. The question is: how can we evaluate a product or technology in the best possible way, and what means are necessary to this end? Indeed, if the product is providing unprecedented performance, it may be quite challenging to think of ways of measuring its efficiency in a rigorous and critical way. In this chapter, we will propose a particular approach to the evaluation of localization systems, and show how it was implemented for our studies.

### 2.1 The Mobile Robot as a Tool for Systematic Evaluation

If we survey the landscape of publications that present localization systems or techniques, we come across a variety of (more or less rigorous) evaluation methods. In fact, many publications (academic as well as commercial), do not state at all how their systems are evaluated. This is a major shortcoming, since localization performance is a function of the evaluation method. Without information on how exactly certain performance values are obtained, only poor judgement, let alone cross-comparisons, can be made.

With the goal of implementing an evaluation system that is systematic, repeatable, and controllable, our approach is centered around the mobile robot. Our approach consists of four core elements: *(i)* a ground truth positioning system, *(ii)* a mobile platform that emulates a mobile target, and that is tracked by the ground truth positioning system, *(iii)* a delimited, well-defined experimental space, and lastly *(iv)* the positioning system to be evaluated. The first question to be answered is how to obtain the ground truth:

The *ground truth* is an error-free measure of a given quantity. Often, such a measure is not achievable, hence, one resorts to the best possible measure given current constraints.

In particular, when designing the ground truth positioning system, one must keep in mind that the accuracy and precision must be superior to those of the system to be evaluated. The choice of the mobile platform may be motivated by factors such as its capacity to embed

additional sensors, its computing resources, its size, its autonomy, and its mobility. Finally, one is left with the choice of the experimental space. Often, this choice is obvious (e.g., the experimental site of a company, the laboratory of a research institution). However, since the physical traits of surroundings affect the behavior of the underlying media employed in a localization system, it is crucial that the space is carefully described and specified. Indeed, benchmark test environments have yet to be developed.

Previous works have considered a variety of evaluation methods. In [40], Gonzalez et al. study UWB localization in an indoor environment. They equip a mobile robot with a laser range finder (as well as the UWB transceivers). In order to obtain ground truth positioning, they perform laser scan matching with help of an office map. In a similar approach, Quigley et al. [101] evaluate the localization capabilities of a set of low-cost sensors. They equip a human with a backpack that carries the set of low-cost sensors (to be evaluated) as well as a laser range finder to obtain a ground truth pedestrian path. However, instead of using a building model as in the former work, they drive a robot through the environment *a priori*, running a SLAM algorithm, to build depth point cloud maps. In their studies on WiFi localization, Ladd et al. [64] use a human to carry the WiFi antenna (attached to a computer), walking around a building floor according to a specific trajectory. Since the human ‘knows’ its approximate trajectory, and WiFi positioning is not expected to produce overly high accuracy, their method is sufficient. Segura et al. [112] evaluate a custom UWB system by using a mobile robot to carry the receiver board. They measure ground truth positioning by attaching a marker to the robot’s ‘tail’ that leaves a track on the floor, which is then measured manually. Rizos et al. [105] use a total station to evaluate the Locata positioning system. They place a surveying prism on their mobile platform (a cart), which also carries the Locata receiver. The total station tracks the prism and logs the position data.

## 2.2 The Experimental Setup

This section is devoted to our setup. In particular, we will detail our mobile platform, and then look into how we combine the remaining three elements (items *(i)*, *(iii)*, and *(iv)* of above) into one infrastructure. The last section in this chapter presents a case study, where our mobile platform itself is the positioning system to be evaluated.

### 2.2.1 Our Mobile Platform: The Khepera III Robot

Since the beginning of navigation research with autonomous mobile robots in the mid 80’s, a multitude of robotic platforms have been developed to satisfy the needs of this increasingly important domain. Whereas early studies were often performed on robots with simplistic sensing capabilities (for example, limited to the combination of sonar sensors and wheel encoders, as in [3, 63], or to the combination of an optical range finder and odometry as in [24]), the platforms used today are often equipped with powerful sensing units, such as laser range finders and cameras [16, 116]. Moreover, *ad hoc* as well as commercially available platforms are often tailored specifically for the tasks they are meant to solve. When operating in large spaces, i.e. spaces which span several rooms or even entire building floors, robots typically need augmented autonomy and combined sensing and computational capabilities. In [108, 120], the robot Minerva is used as a tour guide in a museum during opening hours. Apart from



**Figure 2.1:** Khepera III robot, with dimensions of 12 cm in diameter and 7 cm in height (excluding extension modules). The robot is carrying a tracking module, which consists of a red and a green LED, that are lit up at run-time.

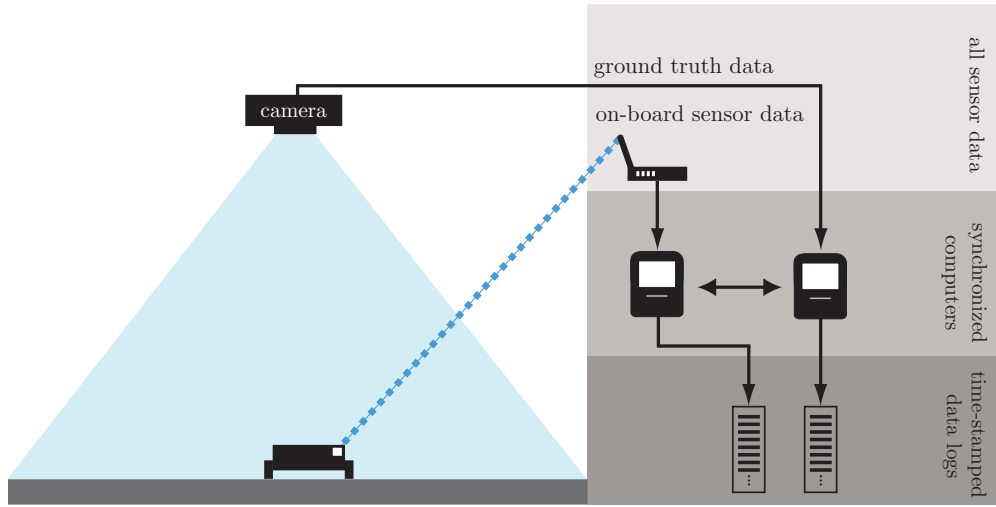
using sonars and wheel encoders, it uses 2 laser range finders which provide a  $360^\circ$  field of view, and an upwards pointing camera which is used for localization based on the structured ceiling. Similarly, in [109], the RWI B21 robot covers museum grounds, and uses an array of 24 ultrasound sensors and a laser range finder. Further, in [118], 3D mapping techniques are deployed on Pioneer robots equipped with 2 laser range finders.

Although the above mentioned robots have shown to successfully perform sophisticated real-life tasks, their large size (roughly 1 m in height, 0.5 m in diameter) makes it hard to integrate them into systematic evaluation systems providing precise ground truth comparisons. More recently, studies have also been performed on smaller platforms. The reduced robot size (below 0.2 m in height, 0.15 m in diameter) is leveraged to conduct systematic evaluation of specific navigation techniques. In [21] as well as in [2], Khepera III and Khepera II robots are used, respectively, for the evaluation of novel calibration techniques. Also, thanks to the reduced size of such platforms, rigorous research in the domain of multi-robot systems has become more tractable.

In order to integrate single- and multi-robot systems into systematic and precise evaluation frameworks, it is practical to reduce the individual robot size to its minimum possible, without stripping it from vital resources such as autonomy, computation, and sensing. Also, a number of further design choices can be considered to maximize the utility of a robotic platform: *(i)* off-the-shelf components and software/hardware standards, *(ii)* modularity enabling extensions for different modalities, *(iii)* non-stop energetic autonomy, and *(iv)* wireless techniques for programming, data logging, and controlling. The Khepera III fulfills all the above requirements. Indeed, it leverages the embedded system/cell phone market, uses wireless software and hardware standards, and runs Linux as its OS *(i)*, for which it is aligned with its device size. The platform is modular *(ii)*, possesses a modern battery technology *(iii)*, and includes both WiFi and bluetooth technology *(iv)*.

The Khepera III is a differential drive robot of 12 cm diameter, produced by K-Team corporation<sup>1</sup> with development assistance from the Distributed Intelligent Systems and Algorithms Laboratory (at EPFL). It is a descendant of the first generation Khepera robot which is smaller

<sup>1</sup><http://www.k-team.com/>



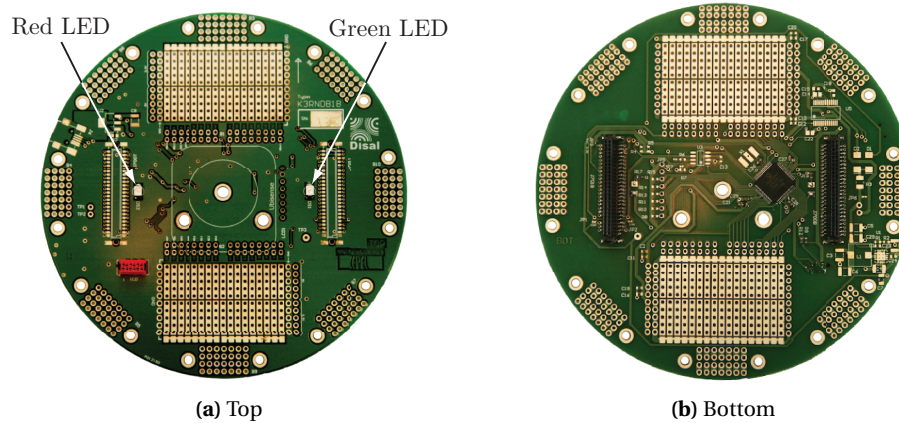
**Figure 2.2:** The experimental infrastructure is composed of two main elements: a vision-based ground truth positioning system, and a mobile platform (which in this case is the positioning system to be evaluated). The overhead camera monitors the mobile robot which drives around in a delimited area. The information from the two subsystems is collected and time-stamped by synchronized computers to produce data logs.

in size (5.5 cm diameter) and since its first release in 1995 has been in use worldwide—to date, more than 2000 Khepera I and Khepera II robots have been sold to over 600 universities. In its latest incarnation, the Khepera III offers a much higher computing power than its predecessors. The KoreBot II extension board provides a standard embedded Linux operating system (Ångström distribution) on an Intel XSCALE PXA-270 processor running at 624 MHz. Basic sensing is formed by a ring of 9 infrared sensors (TCRT5000 Vishay Telefunken) and an additional pair facing downwards (for table top navigation and line following), and 5 ultrasound sensors (400ST/R100 Midas Components Ltd.) placed on the front semicircle. For a precise positioning of the sensors, the reader should refer to [88]. The infrared sensors have both active and passive functioning modes (for reflected emitted light and ambient light measurements, respectively), and an approximate range of 2-25 cm. The ultrasound sensors have an approximate range of 20-400 cm and record a maximum of 3 echoes and their corresponding timestamps. Further, the robot has a stackable expansion bus that enables the addition of custom robot modules. Finally, in order to facilitate communication, an IEEE 802.11 wireless card can be installed in the built-in CompactFlash slot. Unless explicitly stated otherwise, for our experiments, we require the robot to drive (randomly) in the arena at a speed of one robot size per second. The robot runs a Braitenberg controller [15] on-board for basic obstacle avoidance based on information obtained from the robots' infrared proximity sensors.

### 2.2.2 Experimental Infrastructure

An illustration of our baseline experimental infrastructure is depicted in Figure 2.2. Let us relate to the four building blocks we introduced earlier. In our setup, we have (i) an overhead camera system, (ii) a mobile robot (embodied by a Khepera III) and (iii) an arena delimiting the drivable area. Finally, for the sake of introducing our setup in its simplest form, item (iv) is the robot itself. We shall see in later chapters how this setup is easily extendable to various scenarios.





**Figure 2.3:** The tracking module that was developed for our ground truth positioning system. The top view shows the LED pair and the additional placeholders. The bottom view shows the micro-controller, and the 50-pin board-to-board connectors.

There are several possible ways to gather ground truth information about a robot's position, or even its trajectory. Due to its practicality, precision, and efficiency, our method of choice is vision. We installed an overhead camera system in combination with the the tracking software SwisTrack<sup>2</sup>. This software package is especially useful due to its modular architecture including numerous algorithmic components [71]. As well as using the available camera calibration components, we exploited real-time tracking modules enabling us to perform precise analyses of time-dependent robot behavior. In our setup, we use a GigE color camera (Basler-SCA1000-30GC) which has a standard resolution of 1032x778 pixels, and is mounted 2.5 m above the robotic arena. The resulting picture resolution is 5.5 pixels per cm<sup>2</sup>. Detected ground positions are calibrated using the calibration algorithm proposed by Tsai et al. [122], a method which requires information on the position of at least 5 non-collinear points on the image. We performed the calibration procedure on 64 points by installing a uniform grid with dots of 3 cm diameter at its intersections. Further, we performed a second calibration for robot detection which additionally integrates a model of the setup, taking into account the robot height (11 cm). The robot image captured by the tracking software is processed in a pipeline, which outputs the corresponding robot orientation in addition to the 2D position. Using this described framework, we perform real-time tracking at a frequency of 10 Hz. In order to synchronize the real-time robot data with the ground truth measurements and the measurements from the remote positioning system, we build a UDP communication channel. A simple script timestamps the data from the three sources at the incoming ports. This architecture proved highly reliable.

### 2.2.3 An Active Tracking Board

In order to increase the accuracy and precision of our tracking system, we developed a dedicated hardware tracking board based on LEDs (see Figure 2.3). This printed circuit board is designed as an extension module for the Khepera III robot, and was developed in a team-effort with several members of the Distributed Intelligent Systems and Algorithms Laboratory (at EPFL). Its main elements include a red and a green LED, which are controllable by the ro-

<sup>2</sup>[urlhttp://en.wikibooks.org/wiki/Swisstrack](http://en.wikibooks.org/wiki/Swisstrack)

	Dot detection	KIII detection
Mean pos. error [cm]: $\mu \pm \sigma$	$0.71 \pm 0.31$	$1.2 \pm 0.58$
Max. pos. error [cm]	1.45	2.74
Mean orient. error [rad]: $\mu \pm \sigma$	–	$0.068 \pm 0.055$
Max. orient. error [rad]	–	0.253

**Table 2.1:** Accuracy of Overhead Tracking System

bot through an I<sup>2</sup>C communication bus, and a dsPIC33 microcontroller. Also, by providing numerous placeholders in the form of breadboards, the board is designed to accommodate additional testing and prototyping functionalities. We configure the robot tracking software to include a red-green blob detection module, which returns the 2D position (center of the axis defined by the red and green blob), and orientation (perpendicular straight along the axis defined by the blobs, with red to the right, and green to the left). In fact, the placement of the LEDs was optimized so that the orientation would yield best possible results (i.e., by maximizing the distance in between LEDs), while maintaining a robust tracking performance (i.e., by minimizing the distance between the LEDs to avoid interference with potential other robots).

We evaluate our ground truth measurement system by calculating the mean positioning error on detected grid dots placed on the ground, as well as detected KIII robots. We measure the positions of 64 dots, and of 28 placements of the KIII robot, uniformly distributed in a 9m<sup>2</sup> square arena. We also measure the error of the orientation of the robot by rotating it 4 times by 90° per position. While the error on the dot detection is absolute, the error on KIII detection is subject to minor misalignments due to manual placement of the robot on the grid. The results, reported in Table 2.1, show that centimeter-level accuracy can be achieved.

## 2.3 Case Study: Navigation with the Khepera III Robot

In this case-study, we will show how our experimental setup can be conveniently used to both calibrate and evaluate the navigation of the Khepera III robot. Simultaneously, this case study helps validate our proposed experimental setup in an illustrative manner.

### 2.3.1 Calibration of Odometry

Navigation filters typically use two components to estimate a position: a sensor model and a motion model. Here, we will take a closer look at our platform’s motion model and its calibration. The Khepera III robot possesses high resolution wheel encoders (2764 ticks per revolution, roughly 7 ticks per degree). Yet in order to take full advantage of the high precision, odometry parameters must be calibrated. Throughout this section, we will use three robots, and denote them  $\mathcal{R}_1$ ,  $\mathcal{R}_2$ , and  $\mathcal{R}_3$ . We adopt the same formalism as described in [2], where robot velocity  $\nu$  and angular velocity  $\omega$  are related to the angular velocities of the robot wheels  $\omega_R, \omega_L$  as in

$$\begin{pmatrix} \nu \\ \omega \end{pmatrix} = C \begin{pmatrix} \omega_R \\ \omega_L \end{pmatrix}.$$

with matrix  $C$  equal to

$$\begin{pmatrix} \frac{r_R}{2} & \frac{r_L}{2} \\ \frac{r_R}{b} & -\frac{r_L}{b} \end{pmatrix},$$

and where  $r_R$  and  $r_L$  are the right and left wheel radii respectively, and  $b$  the robot wheelbase. The Khepera III factory specifications are  $r_R = r_L = 0.0021\text{m}$  and  $b = 0.08841\text{m}$ . Using these nominal values we have

$$C = \begin{pmatrix} 0.0105 & 0.0105 \\ 0.23753 & -0.23753 \end{pmatrix}. \quad (2.1)$$

In order to correctly predict robot displacement, these parameters ( $r_R$ ,  $r_L$  and  $b$ ) are tuned for an individual robot in a specific setting.

Since the beginning of research on odometry calibration in the late 80's, a number of calibration methods have been proposed [2, 13, 59, 109, 126]. We have chosen to test the appropriateness of 3 particular methods here

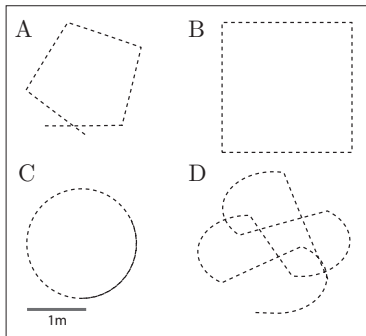
- a** Borenstein et al. [13]
- b** Kelly et al. [59]
- c** Antonelli et al. [2]

using three sets of trajectories, as shown in Figure 2.4:

- set1** 10 trajectories on shape B (clockwise/counter-clockwise)
- set2** 12 trajectories on shapes A and C (clockwise/counter-clockwise)
- set3** 3 trajectories on shape D

Each of the methods **a**, **b** and **c** is based on a different concept and separate hypotheses. In their work, Borenstein et al. propose a straightforward calibration method which treats systematic errors (as opposed to non-systematic). Their parameter correction method is based on a geometric derivation, and assumes that the average of the actual wheel diameters is equal to the nominal one. In order to apply the method, only the beginning- and end-positions of the robot have to be recorded. We performed this calibration on **set1**, for each robot.

Kelly et al. pose odometry as a nonlinear dynamical system. Optimal odometry parameters are found by solving a nonlinear optimization problem which minimizes the error between actual



**Figure 2.4:** Trajectories used to calibrate and evaluate the motion model. Trajectories **A**, **B**, and **C** were performed in both CW and CCW directions.

	Borenstein	Kelly	Antonelli
$\mathcal{R}_1$	$\begin{pmatrix} 0.0105 & 0.0105 \\ 0.2345 & -0.2359 \end{pmatrix}$	$\begin{pmatrix} 0.0155 & 0.0106 \\ 0.2357 & -0.2369 \end{pmatrix}$	$\begin{pmatrix} 0.0142 & 0.0074 \\ 0.2345 & -0.2354 \end{pmatrix}$
$\mathcal{R}_2$	$\begin{pmatrix} 0.0105 & 0.0105 \\ 0.2366 & -0.2365 \end{pmatrix}$	$\begin{pmatrix} 0.0106 & 0.0106 \\ 0.2385 & -0.2385 \end{pmatrix}$	$\begin{pmatrix} 0.0105 & 0.0112 \\ 0.2347 & -0.2344 \end{pmatrix}$
$\mathcal{R}_3$	$\begin{pmatrix} 0.0105 & 0.0105 \\ 0.2348 & -0.2353 \end{pmatrix}$	$\begin{pmatrix} 0.0106 & 0.0106 \\ 0.2389 & -0.2393 \end{pmatrix}$	$\begin{pmatrix} 0.0118 & 0.0097 \\ 0.2336 & -0.2341 \end{pmatrix}$

**Table 2.2:** Odometry coefficients derived from calibration methods

and nominal trajectories. Thus, we have to gather and synchronize robot wheel speed data with tracking information. We performed this calibration on **set2**, for each robot.

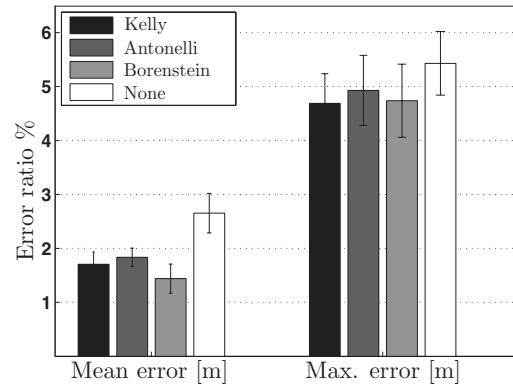
Antonelli et al. propose a least-squares method to estimate the matrix  $C$  directly. In order to apply this method, robot wheel speed data as well as the robot end-positions is necessary. We performed this calibration on **set2**, for each robot.

Finally, we obtain the calibrated parameters shown in Table 2.2.

In order to evaluate the performance of the tested methods, we compute the following metric which gives us the normalized error at time  $T$

$$\epsilon_{od}(T) = \frac{2}{T} \int_0^T \frac{\epsilon(t)}{d_R(t) + d_L(t)} dt \quad (2.2)$$

where  $\epsilon(t)$  is the Euclidean distance between the actual and estimated robot positions, and  $d_L(t)$ ,  $d_R(t)$ , are the distances traveled by the left and right wheels at time  $t$ . We perform this evaluation on both **set2** and **set3**, for each robot. The performance of the calibrated motion models is reported in Figure 2.5. The results of running the Wilcoxon rank-sum test (with threshold 0.05) on the mean error performance data show that while there is a statistically significant improvement in performance when using the calibration methods, there is no significant difference in performance between the three methods themselves. In terms of maximum error, there is no statistically significant difference in the results obtained with calibration from the ones obtained without calibration.



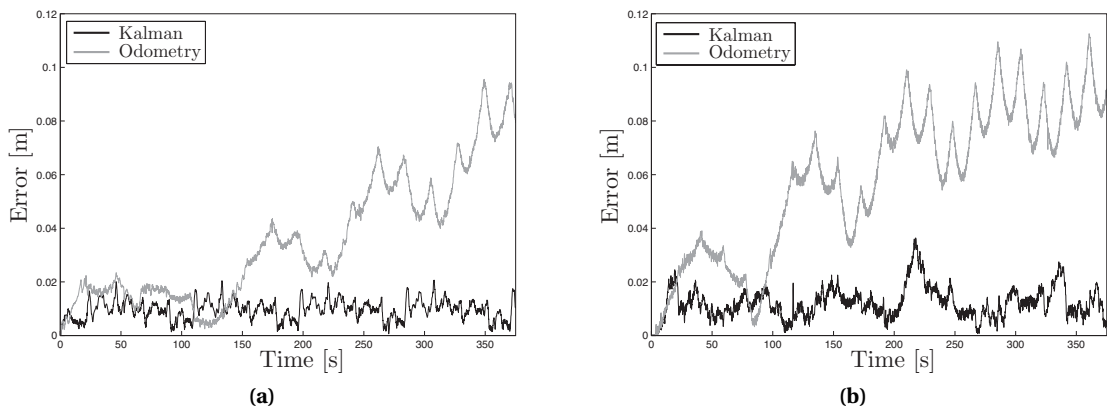
**Figure 2.5:** Normalized (over distance) error ratio obtained for 3 different odometry calibration methods, as well as for non-calibrated robots. The error bars show a 95% confidence interval.

### 2.3.2 Validation: A Basic Navigation Scenario

Finally, we design a simple navigation scenario to test our experimental setup. We require the robot to drive a 20 m long path along a 1 m large square, at an average speed of 5 cm/s. The robot localizes itself while moving along an outer wall, where in a first sub-scenario, it does this by utilizing its infrared sensors, and in a second sub-scenario, it utilizes its ultrasound sensors. The wall is placed in such a way that its visibility with respect to the sensors is preserved throughout the run. Odometry measurements are taken at a frequency of 10 Hz. We perform two runs per robot (CW and CCW directions), for each sensor configuration. Throughout the experiment, we employ our tracking setup and create synchronized data logs of robot navigation data and ground truth positioning data.

In order to localize, we implement an extended Kalman filter referring to a feature-based map. We use an ultrasound measurement model by implementing the scan predictor and matching method as proposed by Leonard et al. [66]. A validation gate is used to determine the correspondence between predictions and observations (we set the gate threshold to two standard deviations). For the 9 infrared sensors, we use a range model to generate a sensor scan, which is then translated and rotated to produce a maximum overlap with our map. The resulting pose update is then integrated into our Kalman filter as proposed in [44].

The localization performance can be observed in Figure 2.6: the error of the position estimates remains bounded, whereas the dead-reckoning error accumulates constantly. The mean localization error for Figure 2.6a is 0.8 cm, and for Figure 2.6b is 1 cm. We note that, since these values are in the order of the ground truth measurement error, we cannot make any conclusions about their exact values.



**Figure 2.6:** A qualitative overview of localization performance. Localization was performed (i) with an extended Kalman filter and (ii) with dead-reckoning (odometry), (a) using 9 concurrently activated infrared sensors, and (b), using 5 concurrently activated ultrasound sensors.

### Summary

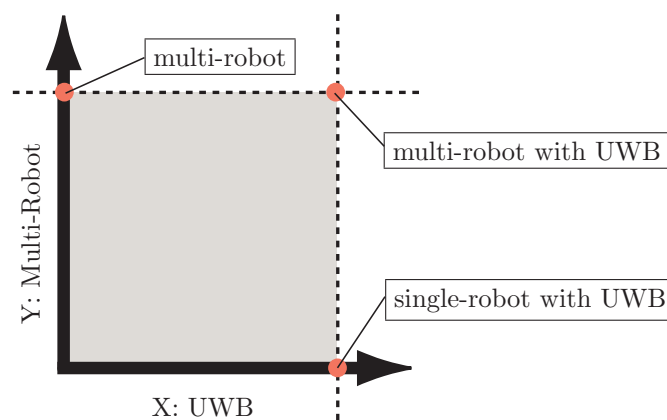
In this chapter, we propose the mobile robot as a systematic tool for evaluating positioning systems. Given its suitability for research purposes, our platform of choice is the Khepera III robot. Also, as we will see later in this manuscript, we exploit this platform's small size to perform multi-robot experiments in a controlled setting. We use a vision-based installation to track the robot and obtain ground-truth positioning with centimeter-level accuracy. The information from the individual positioning subsystems is collected and time-stamped to produce synchronized data logs. Finally, although our approach provides some guidance on how to evaluate localization systems accurately and efficiently, we note that the research community would benefit from more work on general evaluation frameworks that lead to cross-comparable results.

## 3 The Scope of this Thesis

ULTRA-WIDEBAND localization is the focus of this thesis. However, in the early phases of our work, we realized that some of the challenges that UWB presented were best tackled from a completely different angle—literally. Working in a laboratory which claims its expertise in distributed robotic systems, it was a natural choice to look at the problem from a multi-agent perspective. Concretely, we asked ourselves if collaborative methods, in the form of relative positioning, could provide any benefit at all to UWB localization—and vice versa. The road down this path included diving into both of these topics individually, and then merging them together again to conclude our study.

### 3.1 A Two-Dimensional Outline

To help organize our research efforts, we focused our work along two axes: multi-robot localization, and UWB localization (see Figure 3.1). Our intention was to explore and exploit the individual axes themselves and the space spanned by them, and finally to extract a combined system and the knowledge of how to configure it to provide the best possible performance.



**Figure 3.1:** The scope of this thesis

According to this convention, our studies are organized in three parts:

- 1: Single-robot localization with UWB
- 2: Multi-robot localization
- 3: Multi-robot localization with UWB

The first part (**Part II** of this manuscript), single-robot localization with UWB, focuses on the development of probabilistic UWB measurement models for localization. Our models are then embedded in single-robot localization algorithms and tested in our evaluation setup including a real UWB component. Relevant publications include:

- **A. Prorok**, L. Gonon, A. Martinoli, “Online Model Estimation of Ultra-Wideband TDOA Measurements for Mobile Robot Localization”, *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 807–814, 2012
- **A. Prorok**, A. Arfire, A. Bahr, J. R. Farserotu, A. Martinoli, “Indoor Navigation Research with the Khepera III Mobile Robot: An Experimental Baseline with a Case-Study on Ultra-Wideband Positioning”, *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2010. DOI: 10.1109/IPIN.2010.5647880

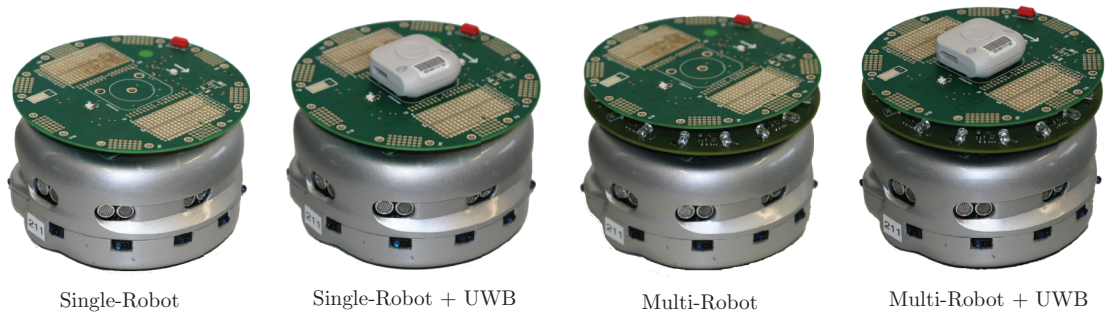
The second part (**Part III** of this manuscript), multi-robot localization, focuses on relative positioning methods for fully decentralized multi-robot systems. The algorithms are tested in our experimental setup on a multi-robot system providing real inter-robot relative positioning data. Relevant publications include:

- **A. Prorok**, A. Bahr, A. Martinoli, “Low-Cost Multi-Robot Localization”, in *Redundancy in Robot Manipulators and Multi-Robot Systems*, D. Milutinovic and J. Rosen, Eds., Lecture Notes in Electrical Engineering, Springer, 2012, pp. 15–34
- **A. Prorok**, A. Bahr, A. Martinoli, “Low-Cost Collaborative Localization for Large-Scale Multi-Robot Systems”, *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4236–4241, 2012
- **A. Prorok** and A. Martinoli, “A Reciprocal Sampling Algorithm for Lightweight Distributed Multi-Robot Localization”, *Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3241–3247, 2011
- S. Goyal, **A. Prorok**, A. Martinoli, “Two-Phase Online Calibration for Infrared-Based Inter-Robot Positioning Modules”, *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3313–3319, 2011

Our last part (**Part IV** of this manuscript), multi-robot localization with UWB, focuses on a combination of the previous two subsystems. Here, the goal is to explore the benefits of merging the technologies, and to define how exactly this merging should be engineered. Relevant publications include:

- **A. Prorok** and A. Martinoli, “Accurate Indoor Localization with Ultra-Wideband using Spatial Models and Collaboration”, *International Journal of Robotics Research (IJRR)*. To appear, 2013





**Figure 3.2:** The experimental platforms used in our work. The Khepera III robot is equipped with additional extension modules. For UWB localization studies, we equip the robot with an UWB emitter. For multi-robot localization studies, we equip the robot with a relative positioning board. In all cases, the top layer extension module is the LED tracking module.

- **A. Prorok** and A. Martinoli, “Accurate Localization with Ultra-Wideband: Tessellated Spatial Models and Collaboration”, *Proceedings of the 13th International Symposium on Experimental Robotics (ISER), Springer Tracts in Advanced Robotics. To appear, 2014*
- **A. Prorok**, P. Tomé, A. Martinoli, “Accommodation of NLOS for Ultra-Wideband TDOA Localization in Single- and Multi-Robot Systems”, *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2011. DOI: 10.1109/IPIN.2011.6071927

Figure 3.2 shows how the Khepera III robot was configured for the various experimental setups featured in our projects. We were able to conveniently exploit the robot’s stackable property to include the hardware modules that were needed for the dedicated experimental work.

## 3.2 Contributions of our Work

The contributions of our work are grouped into three main modules, and attributed to the three points within the space of our research (as in Figure 3.1):

### 1. Single-robot localization with UWB:

We improve upon state-of-the-art by addressing the peculiarities of the propagation of UWB ranging signals with a model that can capture the spatiality as well as the statistical multimodality of the error behavior. To this means, we develop a general TOA measurement model that is a mixture of Gaussian and log-normal distributions, and extend this model to include TDOA measurements. As the resulting general TDOA measurement model is intractable, we develop an approximated closed-form model that successfully captures the essential error modes. Our new, analytically tractable formulation enables the application of an efficient estimation algorithm, namely Expectation Maximization (EM), to determine the values of the model parameters. To the best of our knowledge, our model is the first UWB TDOA measurement model for mobile robot localization. Our approach is experimentally validated in a room-sized setup including obstacles that induce NLOS signal propagation. Final results show a localization performance with errors slightly above 10 cm.

### 2. Multi-robot localization:

Our collaborative localization strategy is based on a Monte Carlo method and distinguishes itself from prior work by emphasizing cost-efficiency, full decentralization, and scalability. We develop a dedicated robot detection model that builds upon a particle-based belief representation, and that integrates relative range and bearing measurements, made available through noisy relative positioning sensors. In addition to this model, we develop an algorithmic component, namely a reciprocal sampling routine. This novel routine exploits collaboration in the robot team by drawing new particles in regions of the state space that are likely at correct positions, according to neighboring robots' beliefs. The routine facilitates the convergence of position estimates and increases the robustness of the localization algorithm. In order to reduce the complexity of the collaborative localization algorithm, we develop a second algorithmic component, namely a clustering routine. This routine includes a novel, summarized formulation of relative observations based on particles, and integrates directly into our original detection model. Our final collaborative localization strategy has a complexity of  $O(MK)$ , for  $M$  particles and  $K$  clusters, and in practice, is fully scalable since it does not depend on the number of robots in the team.

### 3. Multi-robot localization with UWB:

By combining UWB localization with collaborative localization, we present a novel setup and perform pioneering experimental work in order to study the combined usage of the two sensor modalities and their associated methods. To the best of our knowledge, our setup is the first to consider the fusion of UWB positioning data with exteroceptive sensing data. Moreover, our experiments with teams of up to 10 mobile robots, equipped with both UWB emitters as well as relative positioning modules, are unprecedented in this context. Various experimental results indicate that collaboration—even if through noisy relative positioning sensors—is a useful tool to reduce overall localization errors. Final results show a localization performance with errors below 10 cm.

The following three parts of this manuscript are dedicated to the above three modules, respectively, and will elaborate our contributions in detail.

## Summary

In this chapter, we introduced the general outline of this thesis and presented the hardware platforms that were used to perform the experimental work. A brief summary of the three parts of our work also included a list of our relevant publications. Our contributions can be associated with the three main modules of this manuscript: UWB localization, collaborative localization, and combined UWB and collaborative localization.

# Ultra-Wideband Localization **Part II**



## 4 Introduction

*Making what is good better.*

To date, one of the most competitive techniques for indoor localization is based on UWB technology, mainly because it provides a very interesting trade-off in terms of complexity/cost, scalability/coverage and accuracy/precision. Furthermore, being a relatively new addition to the field of localization, there is still a lot of room for dedicated development—on both hardware and software fronts. Our approach is directed towards the latter: by proposing new models and algorithms, our aim is to accommodate the particularities of UWB signals in the context of localization. This chapter lays the ground for our developments. We start off by understanding the meaning of UWB, and its implications when used for localization. We then introduce the experimental setup that we use to test and validate our proposed methods. Finally, using a simple case study with a commercial system, we experimentally motivate the need of developing more sophisticated UWB localization systems.

### 4.1 Definition

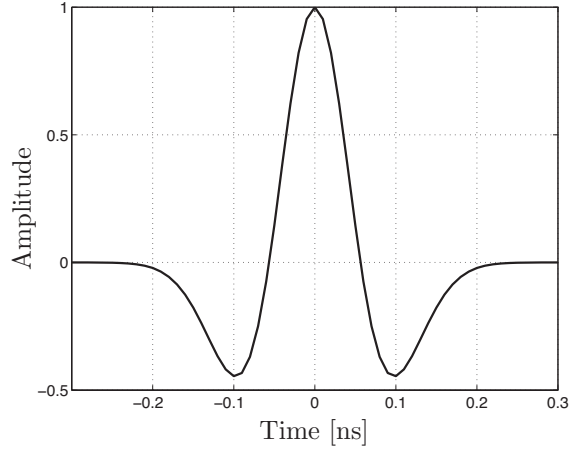
According to the US Federal Communications Commission (FCC), an UWB signal is defined as a signal that has a fractional (relative) bandwidth larger than 20%, or an absolute bandwidth of at least 500 MHz. The absolute bandwidth is defined as the difference between the upper and lower cut-off frequencies (at -10 dB). Thus for a bandwidth  $B$  defined as

$$B = f_H - f_L \quad (4.1)$$

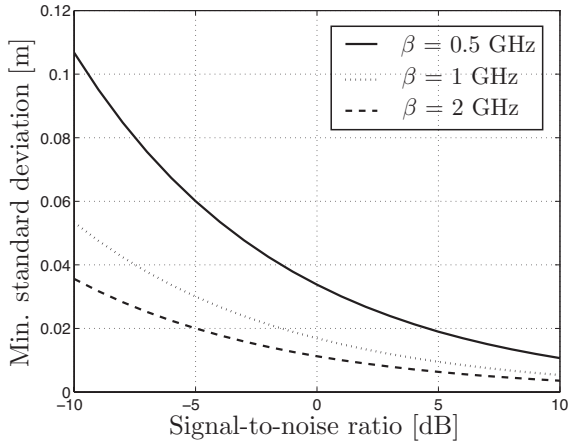
the fractional bandwidth is defined as

$$B_{frac} = \frac{2(f_H - f_L)}{f_H + f_L}. \quad (4.2)$$

Generally, as the bandwidth relates to the inverse of the wave duration, large bandwidths induce very short waveforms. Indeed, since the UWB waveforms are commonly in the order



**Figure 4.1:** Second-order Gaussian pulse as in Equation (4.3), with  $\zeta = 0.2$  and  $a = 1$ . The pulse width is approximately 0.5 ns.



**Figure 4.2:** The CRLB in Equation (4.4), for an unbiased TOA estimate based on a measurement model with additive, zero-mean Gaussian noise. The symbol  $\beta$  denotes the effective bandwidth.

of a nanosecond, and are often emitted with very low duty-cycles, they are referred to as *pulses* [110]. Systems characterized by their low duty-cycles (in contrast to continuous transmissions) are called *impulse radio* (IR) systems. The waveform itself can take various forms. Common implementations are based on derivatives of the Gaussian pulse, modified Hermite polynomials, and wavelet functions. As an example, let us consider the second-order Gaussian pulse

$$\omega(t) = a \left( 1 - \frac{4\pi t^2}{\zeta^2} \right) \exp^{-2\pi t^2/\zeta^2}, \quad (4.3)$$

where  $t$  is time,  $a > 0$  is the pulse amplitude, and  $\zeta$  is a time constant. This pulse is shown in Figure 4.1, in the time domain, and is characterized by its narrow peak.

Since UWB devices occupy a large range of frequencies, if allowed to transmit unrestrictedly, they could potentially jam other systems working within the same frequency range. For this reason, UWB systems have to adhere to power emission limits. These limits are defined as a function of the equivalent isotropically-radiated power (EIRP), which is the amount of power that an antenna would emit in a given direction relative to a theoretically isotropic antenna. Thus, for any given direction, the regulations state that the EIRP must not exceed -41.3 dBm/MHz in the frequency range 3.1-10.6 GHz, in indoor as well as outdoor environments. Outside of this frequency band, the emission level must be even lower (e.g., protecting WiFi and GPS signals).

When used for localization purposes, it is interesting to analyze the ranging characteristics of an UWB signal. The theoretical accuracy of a TOA signal can be visualized as a function of the Signal-to-Noise Ratio (SNR), as shown in Figure 4.2. The plot shows the Cramér Rao Lower Bound (CRLB) for an unbiased TOA estimate  $\hat{r}$ , which is based on a measurement model including an additive, zero-mean Gaussian white noise component. The expression for the CRLB is [110]

$$\sigma_{\hat{r}} \geq \frac{c}{2\sqrt{2\pi}\sqrt{\text{SNR}}\beta}, \quad (4.4)$$

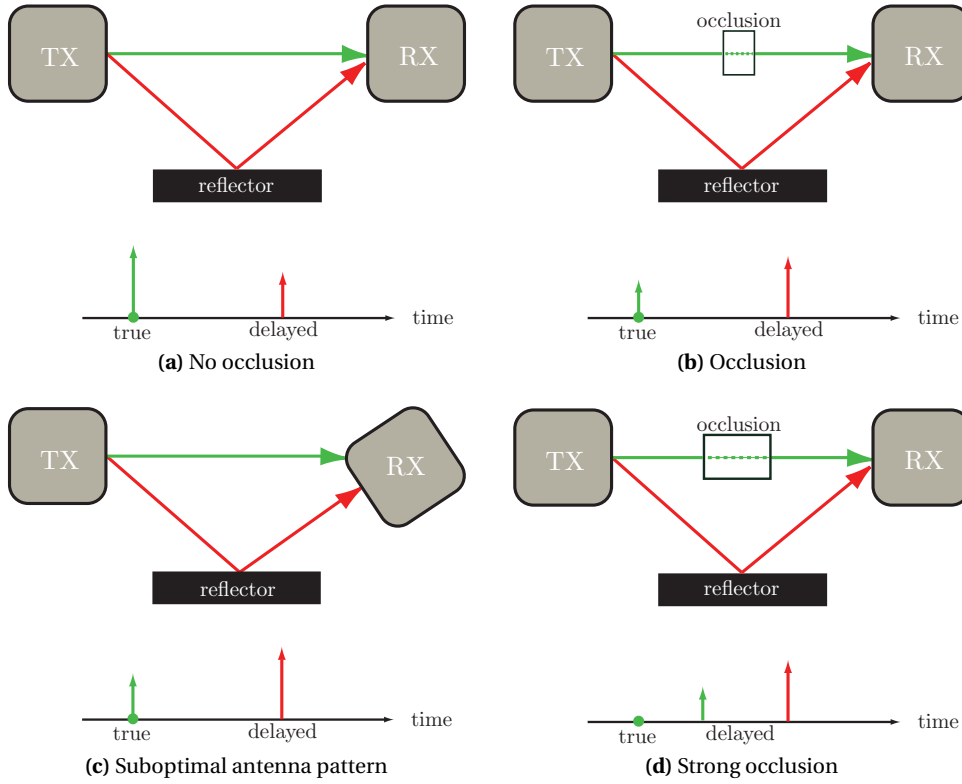
where  $c$  is the speed of light, SNR is the signal-to-noise ratio, and  $\beta$  is the effective bandwidth. The plot shows how, theoretically, the error of an unbiased estimate of the TOA is in the order of a few centimeters, for a wide range of SNRs.

## 4.2 Motivation

UWB has gained in popularity since 2002, when the FCC in the USA allowed the unlicensed use of UWB devices (subject to the regulations elaborated above). Due to its unique characteristics, UWB has attracted the attention of developers in many industries. The localization industry in particular has taken keen interest in UWB, for the following reasons:

- High ranging accuracy: The large frequency spectrum produces narrow waveforms. Thus, ranging signals, which are measured in the time domain, provide potentially high accuracy due to the fine time resolution. Also, as a consequence, the resolution of multipath components is facilitated.
- Penetration through obstacles: Unlike narrowband systems, UWB systems can penetrate effectively through various materials. The low frequencies included in the broad frequency spectrum have long wavelengths, which allows UWB signals to penetrate a variety of materials, including walls and the ground. This capability is particularly valid when a UWB system is allowed to occupy the low-frequency portion of the radio spectrum.
- Low-complexity transmitter: Since UWB transmission is carrier-less (the signals are not modulated on a continuous waveform with a specific carrier frequency), the transmitter design requires fewer RF components than traditional (narrowband) carrier-based technologies. The boards are, thus, less complex and less expensive to produce than their narrowband counterparts. Also, the generation of UWB pulses consumes very little power: even when using high duty cycle rates, the consumption is in the order of a few  $\mu\text{W}$  [75].

Indeed, when using UWB for localization, the basic measured quantity is time-of-flight. In practice, however, TOA systems can be cumbersome due to the complexity induced by the required synchronization of a mobile node with the base stations. Instead, it is a common choice to implement TDOA systems which are significantly more practical, since only the synchronization among base stations is required (refer to the elaborations in Section 1.1.2). Whereas UWB transmitter circuitry is already low-cost and low-power, such a design choice enables a further reduction of the consumption power, and a potential miniaturization of the emitter boards.



**Figure 4.3:** Schematic illustration of the signal path, for four different cases. (a) In this LOS scenario, the multipath signal is delayed and attenuated with respect to the true TOA signal. (b) In this NLOS scenario, the true TOA is attenuated, and is weaker than the multipath signal. (c) Due to a suboptimal antenna pattern, the true TOA signal is attenuated, and is weaker than the multipath signal. (d) In this severe NLOS scenario, the true TOA signal is attenuated as well as delayed.

### 4.3 Challenges

Despite the desirable traits listed above, UWB localization is challenged by NLOS scenarios that induce multipath signal components [110]. Indeed, in the absence of multipath components, the estimation of a TOA signal is relatively straight-forward: a template signal is cross-correlated with the received signal to determine the correlation peak (which corresponds to the TOA). In a NLOS scenario, signals traveling along the direct signal path are in competition with signals reflected by nearby objects: multiple replicas of the template signal arrive at the receiver end, with varying delays and varying attenuation levels. Thus, even though the fine time granularity of UWB signals facilitates the resolution of multipath components, it becomes much harder to determine which is the true TOA. If the correlation peak is the result of a multipath signal, the strongest path is not necessarily the first arriving path, and leading edge detection algorithms are required to determine the shortest path [26]. Such algorithms, however, are susceptible to a low SNR and require a well calibrated threshold. Today, state-of-the art algorithms consider the determination of the first arriving path as an estimation problem for channel delays [130], with Maximum Likelihood methods amongst one of the more common approaches. Even with these sophisticated methods, however, the resolution of multipath signals is still a hard problem. For instance, if the multipath signals overlap with the true TOA signal, the detected



correlation peak will be shifted with respect to its true counterpart. The task of detecting first signal paths is generally exacerbated by the necessity of maintaining very high signal sampling rates, in the order of several GHz.

In this thesis, we build upon the low-level TOA detection algorithms by proposing a high-level measurement model that takes the raw TOA estimate as its input, and returns a measure of its likelihood. Our model complements the low-level approaches by capturing all types of propagation environments, thus increasing the robustness of the final position estimate. Figure 4.3 illustrates four different scenarios which provoke multipath signals, highlighting the challenges encountered in determining the true TOA in face of multipath components. In particular, we note how a delayed signal or a multipath signal adds a *positive* bias to the true TOA estimate. We will later refer to this artifact when we develop our model (in Section 5.2.2).

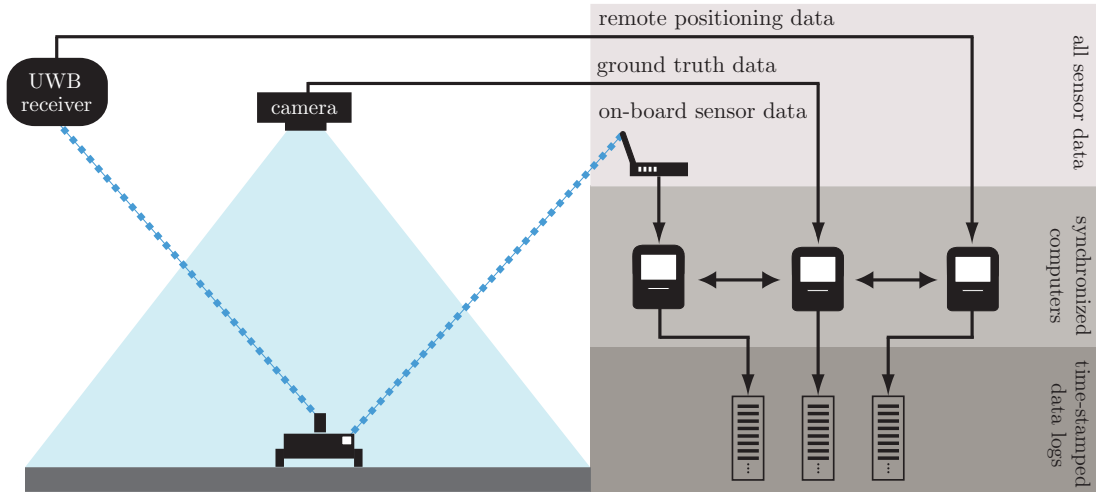
**Line-of-sight (LOS)** refers to a signal propagation characteristic. For various propagation methods, signal transmissions are successful only in scenarios where the direct path between emitter and receiver is not occluded. This property can be inferred by geometric *raytracing*.

**Non-line-of-sight (NLOS)** is the inverse of a LOS scenario: a geometric *ray*, which is cast from an emitter, is blocked from the receiver by an object lying on its path. Although signal transmissions do not always fail entirely in this case, they may nevertheless be challenged.

## 4.4 Experimental Setup

The studies in this part focus on methods that potentially improve UWB localization. In order to test and validate our proposed methods, we use an experimental setup that features an UWB component. We have already established a baseline experimental system in Section 2.2.2—this system is now easily extended to take into account the additional UWB component. In reality, we will consider a remote UWB positioning system (i.e., mobile emitter and fixed receivers). By equipping the mobile robot with an UWB emitter tag, and including the UWB sensor measurements in our data pipeline, we are able to track the resulting positioning performance. Figure 4.4 illustrates our new setup.

The UWB localization system employed in this work is commercially available from Ubisense, Series 7000 (sensors and compact tags), and consists of four receivers (base stations) and a set of emitter tags. This system operates in the 6-8 GHz channel, and uses an additional narrowband channel (2.4 GHz) for telemetry. The emitter tags have configurable pulse emission rates—we use a maximum emission rate of 10 Hz. The actual update rate (deduced from our data logs) is approximately 2.6 Hz. The positioning method provided by the Ubisense system is based on TDOA as well as AOA measurements. The TDOA measurement method requires that the base stations are synchronized. Practically, this is done by routing Ethernet cables from each of the base stations to one of the base stations which is designated as the master. The timing computed at the master station serves as a reference for all other timings, and is a common variable among all TDOA equations. The base stations are installed on the ceiling, in the corners of our 40 m<sup>2</sup> laboratory, and the emitter tag is attached to the robot's tracking



**Figure 4.4:** The experimental infrastructure is composed of three elements: a vision-based ground truth positioning system, a UWB remote positioning system, and one mobile robot carrying an emitter tag. The overhead camera monitors the mobile robot which drives around in a delimited area. The information from the three subsystems is collected and time-stamped by synchronized computers to produce data logs.

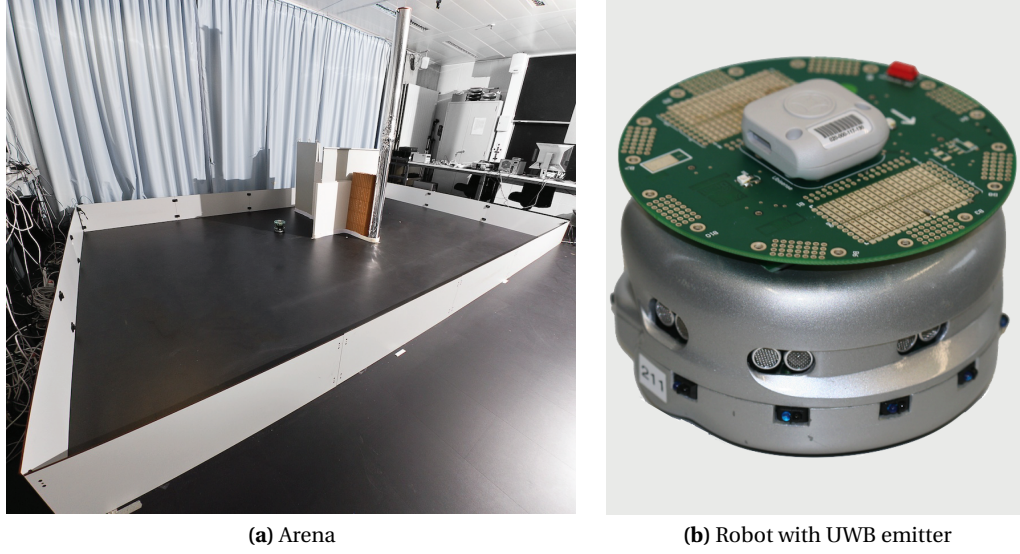
module (see Figures 4.5a and 4.5b).

In order to create a NLOS propagation scenario that occludes direct signal paths between the UWB emitters carried by the robots and the four base stations, we design an experimental setup that includes an obstacle. We install a cross-shaped object in our arena (see Figures 4.5a): the obstacle is 1.5 m long, 1 m high and 20 cm thick, and is composed of several modules made of various materials (brick, plaster, metal, wood). At its extremity, in the center of the arena, we attach a 2 m high tube covered in aluminum. These shapes and materials are chosen in order to realistically emulate the various effects of a typical indoor environment on UWB signal propagation, as previously discussed in Section 4.3. Figure 4.6a shows a schematic overhead view of our arena including the obstacle and the four base stations. Figure 4.6b illustrates an example of the NLOS effect we wish to provoke with the obstacle in this particular setup.

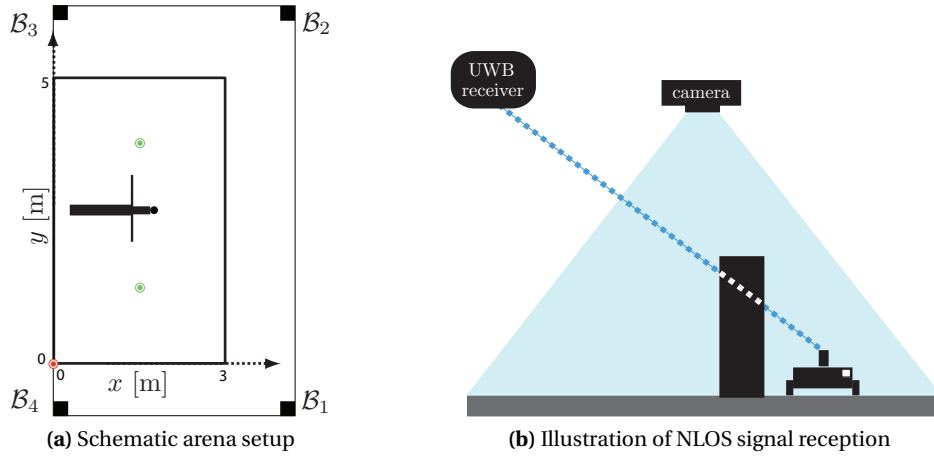
## 4.5 Case study: Evaluation of a Commercial UWB System

Before embarking on our research mission, let us analyze the performance of a ready-made UWB system, produced by Ubisense, using the position estimates that are output by the system software (which fuses AOA and TDOA measurements). We design a basic experiment where the robot is required to drive around the obstacle in a square, ten times. We log and synchronize the data (Ubisense position estimates and ground truth data), and use this information to calculate the positioning error: the Euclidean distance of the Ubisense estimate to the associated ground truth position.

Figure 4.7 shows, qualitatively, where the position estimates are placed with respect to the ground truth trajectory. Figure 4.8 shows the error of the Ubisense position estimates as a function of time. The robot takes approximately 175 s to complete one tour of the square. The plot illustrates how one specific locality along this trajectory produces higher error values than

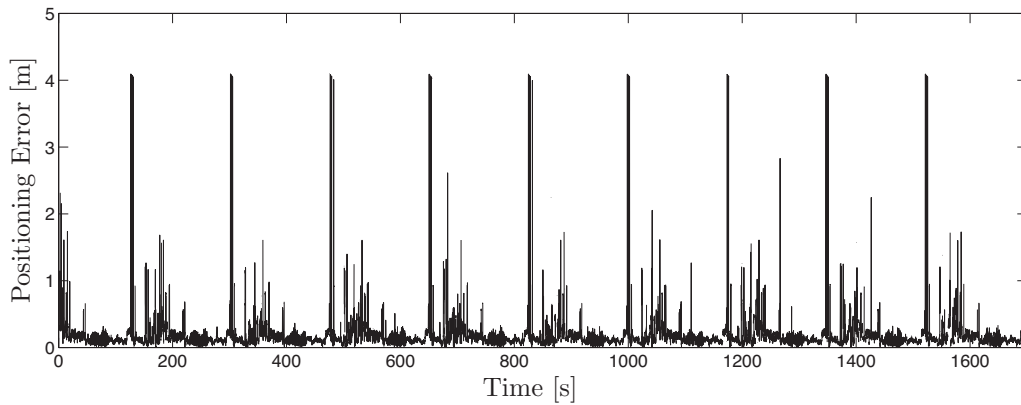
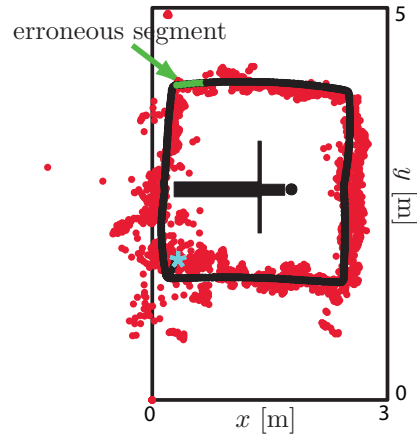


**Figure 4.5:** The experimental setup is located in a 40 m<sup>2</sup> laboratory, with UWB receivers mounted in the corners of the room. The actual experimental space is delimited by a 3 m × 5 m large arena. The center of the arena is occupied by an obstacle, purposely placed to provoke NLOS scenarios. The overhead camera system is composed of two separate cameras mounted on the ceiling, each overlooking one half of the rectangular shaped arena.



**Figure 4.6:** (a) Schematic illustration of the arena setup. The origin is marked in red—the four base stations are mounted on the ceiling and located as follows:  $B_1$  at [3.57, -0.19],  $B_2$  at [3.77, 5.52],  $B_3$  at [0.05, 5.55],  $B_4$  at [0.07, -0.19]. The obstacle is placed at  $y = 2.7$  m, and is 1.5 m long. Two cameras, marked in green circles, are mounted on the ceiling and located at [1.5, 1.5] and [1.5, 3.5]. (b) Example illustration of the NLOS effect we wish to achieve in our setup. Even though our setup employs two cameras to cover the 3 m × 5 m space, care has to be taken that the obstacle does not block the view from at least one of the cameras (this would lead to interrupts in the ground-truth tracking).

**Figure 4.7:** The robot drives in a square trajectory 10 times around the obstacle, in counter-clockwise direction. At time zero, the robot is at the location marked with a star. The ground truth trajectory is marked in black, and the Ubisense position estimates are marked by red dots. The arrow designates the short segment of the trajectory that produced the highest positioning errors (above 3 m).



**Figure 4.8:** Positioning error in meters as a function of time (along the trajectory depicted in Figure 4.7). Ten tours around a square trajectory were made, where each tour takes approximately 175 s. The periodicity of the tours is apparent through the regularity of the error behavior, for revisited localities along the trajectory.

the rest of the trajectory—this locality is marked by the arrow in Figure 4.7. Although the overall positioning accuracy is quite good, with a mean error of 0.23 m, these results demonstrate that UWB positioning is not as robust as we could hope for—in this case, even in presence of the redundant combination of AOA and TDOA measurements. For this particular run, only 5% of the error data lies beyond 0.5 m. However, by avoiding these erroneous estimates, the overall error could potentially be reduced to a mean of 0.15 m, improving the overall performance by 35%.

In fact, this data supports the assumption that spatiality is an important factor. As discussed before and graphically illustrated in Figure 4.3, UWB signal propagation is affected by the physical environment. Thus, for a static environment, and assuming that occlusion is the main source of error, we can assume that positioning performance is systematically reproducible in a given region of space. We will revisit this assumption later on in this manuscript.

Finally, this experiment also demonstrates the utility of using a controllable mobile robot for evaluating positioning systems. We have shown how to exploit its precise maneuverability to test the system's performance as a function of space and time. The results are conveniently analyzed, and the whole setup is easily reproduced.

##### **Summary**

In this chapter we introduced UWB localization by outlining its promises and limitations. Our motivation of pursuing research in this field is three-fold: potential high ranging accuracy, penetration through obstacles, and cost-efficient transmitters. These features are especially attractive for localization, targeting ubiquitous miniature embedded devices. We identify NLOS propagation scenarios as one of the great challenges in dealing with UWB for localization. We also introduce our experimental setup, which builds upon our baseline setup by adding a UWB component. To conclude this chapter, a simple experiment demonstrates the error behavior of a commercially available UWB system. An analysis of the error behavior suggests that performance can be improved significantly by tackling NLOS, and that this error behavior is in fact a spatial phenomenon.



## 5 Sensor Model

THE ability to accurately predict measurement errors is the key to accurate localization. As we saw in the previous chapter, UWB measurement errors display a very particular behavior. In this chapter, we elaborate our reasonings on UWB error behavior, and subsequently derive our error model. The final result is an elegant and efficient error model, which is specially tailored for TDOA measurements. Also, as observed earlier in this manuscript, UWB errors exhibit a dependency on *space*. For this reason, we take care to build our model on a spatial template that features various design parameters, and can be configured according to user needs. The following text will first formally introduce our problem statement, and then continue to elaborate our approach within this formal context.

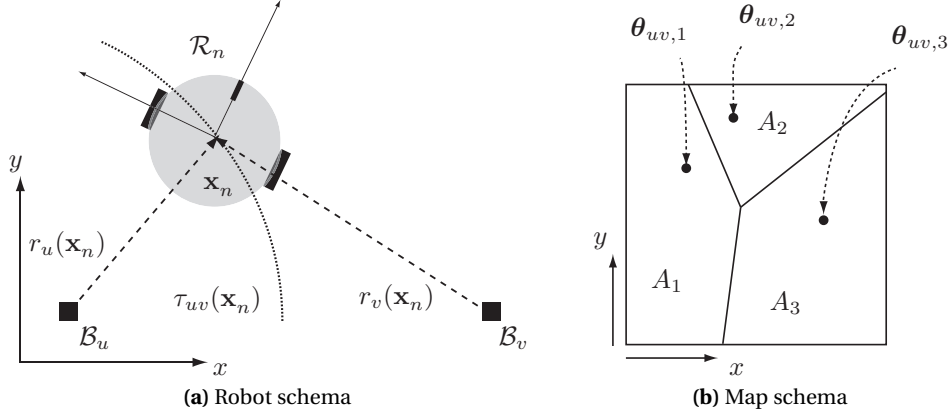
### 5.1 Problem Formulation

We consider a pair of UWB base stations  $\langle \mathcal{B}_u, \mathcal{B}_v \rangle$ , both fixed and well-localized in an absolute coordinate system, and a robot  $\mathcal{R}_n$ , equipped with an UWB emitter tag, at position  $\mathbf{x}_{n,t}$ , as illustrated in Figure 5.1a. At any given time  $t$ , the robot  $\mathcal{R}_n$  may receive a measured TDOA value  $\hat{\tau}_{uv,n,t}$  from any pair of base stations  $\langle \mathcal{B}_u, \mathcal{B}_v \rangle$ . We denote by  $T_{n,t} = \{ \langle \hat{\tau}_{uv,n,t}, \langle \mathcal{B}_u, \mathcal{B}_v \rangle \rangle | \exists \langle \mathcal{B}_u, \mathcal{B}_v \rangle \in \mathbb{B} \}$  the set of TDOA measurements received by a robot  $\mathcal{R}_n$  at a given time  $t$ , where  $\mathbb{B}$  is the set of all base station pairs.

The TDOA measurement error  $\Delta\tau_{uv,n,t}$  for robot  $\mathcal{R}_n$  and base station pair  $\langle \mathcal{B}_u, \mathcal{B}_v \rangle$  is defined as the difference between the nominal (error-free) TDOA value at the actual robot position and the measured TDOA value:

$$\Delta\tau_{uv,n,t} \triangleq \Delta\tau_{uv}(\hat{\tau}_{uv,n,t}, \mathbf{x}_{n,t}) = \hat{\tau}_{uv,n,t} - \tau_{uv}(\mathbf{x}_{n,t}), \quad (5.1)$$

where  $\tau_{uv}(\mathbf{x}_{n,t}) = r_u(\mathbf{x}_{n,t}) - r_v(\mathbf{x}_{n,t})$ , and  $r_u(\mathbf{x}_{n,t})$  is the distance between base station  $\mathcal{B}_u$  and  $\mathbf{x}_{n,t}$  at time  $t$ . In order to model the UWB error behavior, we take account of spatiality by defining a set  $\mathcal{M}_{uv}$  of a number of  $N_A$  areas:  $\mathcal{M}_{uv} = \{ \langle A_a, \boldsymbol{\theta}_{uv,a} \rangle | a = 1, \dots, N_A \}$  where  $\boldsymbol{\theta}_{uv,a}$  is a parameter vector, and  $A_a \subset \mathbb{R}^2$ . Note that the areas are disjoint  $\bigcap_n A_a = \emptyset$  and their union  $\bigcup_n A_a$  covers the whole space. In other words, each area  $A_a$  is associated with a parameter vector  $\boldsymbol{\theta}_{uv,a}$ , as illustrated in Figure 5.1b. We refer to  $\mathcal{M}_{uv}$  as the *map* for base station pair  $\langle \mathcal{B}_u, \mathcal{B}_v \rangle$ , and denote the set of all maps as  $\mathcal{M} = \{ \mathcal{M}_{uv} | \exists \langle \mathcal{B}_u, \mathcal{B}_v \rangle \in \mathbb{B} \}$ . Furthermore, we define



**Figure 5.1:** (a) System of one robot at position  $\mathbf{x}_n$  and two well-localized UWB base stations  $\mathcal{B}_u$  and  $\mathcal{B}_v$ . The figure shows the true ranges  $r_u(\mathbf{x}_n)$  and  $r_v(\mathbf{x}_n)$  of robot  $\mathcal{R}_n$  to the respective base stations, as well as a segment of the hyperbola resulting from the range-difference  $\tau_{uv,n}$ . (b) Distinct UWB error models  $p(\Delta\tau_{uv,n,t}; \theta_{uv,a})$  are mapped to individual areas  $A_a$ .

a function  $m_{uv} : \mathbb{R}^2 \mapsto \Theta$  that maps to any position in two-dimensional space a parameter vector in the finite set  $\Theta$  :

$$m_{uv}(\mathbf{x}_{n,t}) = \theta_{uv,a}, \text{ such that } \exists \langle A_a, \theta_{uv,a} \rangle \in \mathcal{M}_{uv}, \text{ with } \mathbf{x}_{n,t} \in A_a. \quad (5.2)$$

Finally, we model the error  $\Delta\tau_{uv,n,t}$  for a given base station pair  $\langle \mathcal{B}_u, \mathcal{B}_v \rangle$  with a probability density function  $p$  that covers an area  $A_a$  (such that  $\mathbf{x}_{n,t} \in A_a$ ) and that depends on the parameter vector  $\theta_{uv,a}$ . We define our error model as

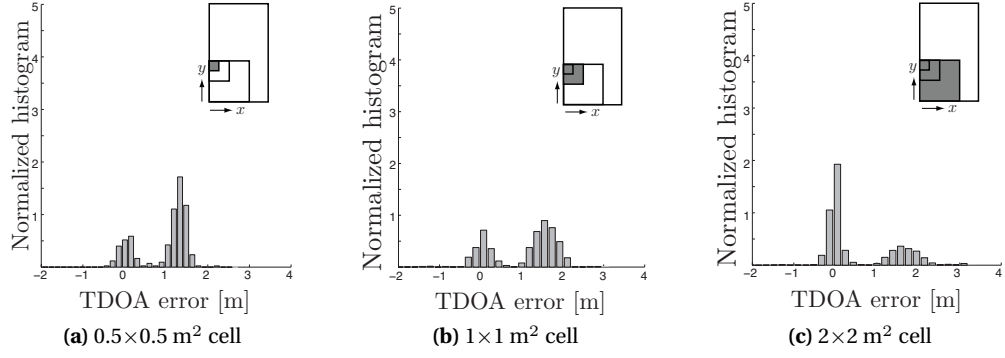
$$p(\Delta\tau_{uv,n,t}; \theta_{uv,a}) = p(\Delta\tau_{uv,n,t}; m_{uv}(\mathbf{x}_{n,t})). \quad (5.3)$$

This concludes our problem formulation. The following text will elaborate how we develop the formula for the probability density function  $p$ .

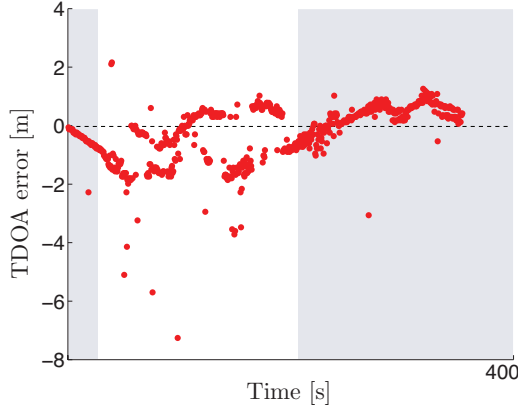
## 5.2 The UWB Error Model

In this section, we develop our UWB error model. As the reader will remember, a UWB measurement is based on time-of-flight measurements. A direct range measurement thus corresponds to a TOA measurement. We previously highlighted, however, that TOA systems are potentially cumbersome due to the required synchronization of emitter and receiver boards. As a consequence, the implementation of TDOA systems is common and popular. Hence, our modeling approach also addresses TDOA systems by extending our baseline TOA error model. The following sections will display our thought process and derivations, leading to a final UWB TDOA error model that is subsequently used throughout our work.





**Figure 5.2:** Histograms of TDOA error data for base station pair  $\langle \mathcal{B}_1, \mathcal{B}_2 \rangle$ . The bars are normalized so that the total area is equal to 1. The data is collected over a (a)  $0.5 \times 0.5 \text{ m}^2$  large square, (b)  $1 \times 1 \text{ m}^2$  large square, and (c)  $2 \times 2 \text{ m}^2$  large square. The areas of data collection are indicated by the shaded cells in the schematized arena, in the top right corner of each panel.



**Figure 5.3:** Example of the multimodal nature of TDOA error data. The data was collected by a robot driving through our experimental setup. The data points in the white (non-shaded) area exhibit a multimodal behavior.

### 5.2.1 Preliminaries

Before proceeding with the development of a formal model, let us take a look at what raw TDOA error data looks like by considering two experiments. Figure 5.2 shows normalized histograms of TDOA error data for a base station pair. We performed the data collection in three overlapping areas of different sizes. Figure 5.3 shows an example of TDOA error data that was collected by a robot driving through our experimental setup (in a straight line). By observing the plots in both figures, we can make the following three conclusions: (i) the error behavior is multimodal, (ii) the multimodal characteristic is preserved over different scales, and (iii) the error behavior is different at different locations in space. These observations will motivate the reasonings in our model development.

Another factor that we include in our reasonings is the requirement to capture mixed LOS and NLOS propagation scenarios. Indeed, without a highly detailed and complete model of our environment, we cannot know for sure what circumstances affect the signal propagation at any given time, and it is best to assume a model that accommodates the influence of both propagation types. Indeed, how exactly the mixing property is defined is one of the challenges tackled by our model, and will be discussed later.

Finally, our baseline model will refer to the main particularity of NLOS propagation (illustrated in Figure 4.3): the introduction of a *positive* bias in the range measurement. Importantly, we will consider this artifact in addition to standard measurement noise. For the sake of simplicity, we will start off by considering a model based on direct range measurements (TOA), and then incrementally arrive at our final model based on range-difference measurements (TDOA).

### 5.2.2 General UWB Measurement Model

For the sake of brevity, our following derivations omit the subscript  $t$ , time. We will begin by detailing a general model  $p^*(\Delta\tau_{uv,n}; \cdot)$ , which, however, will turn out to be analytically non-tractable. The subsequent subsection then develops an approximation to this general model, producing the closed-form model  $p(\Delta\tau_{uv,n,t}; \theta_{uv,a})$  (introduced in Section 5.1).

Our baseline error model for the range<sup>1</sup> between a base station  $\mathcal{B}_u$  and a target node (robot  $\mathcal{R}_n$ ) at position  $\mathbf{x}_n$  is

$$\hat{r}_{u,n} \triangleq r_u(\mathbf{x}_n) + \epsilon + Yb_u \quad (5.4)$$

where  $r_u(\mathbf{x}_n)$  represents the true distance,  $b_u$  is a non-negative distance bias introduced by a NLOS signal propagation, and  $\epsilon \sim p_{\mathcal{N}}(0, \sigma_{\mathcal{N}}^2)$  is a zero-mean Gaussian measurement noise with variance  $\sigma_{\mathcal{N}}^2$ , common to all base stations. The random variable  $Y$  qualifies the occurrence of a NLOS signal path and follows a Bernoulli distribution. Explicitly, it takes the value 1 with probability  $(1 - P_{L_u})$  and the value 0 with probability  $P_{L_u}$ , where  $P_{L_u}$  is the probability of measuring a LOS path, and correspondingly,  $(1 - P_{L_u})$  is the probability of measuring a NLOS path.

Despite the complexity of NLOS error patterns, current work discusses the suitability of a variety of statistical models with exponential behavior, supported on the semi-infinite interval  $(0, \infty)$  [1, 100]. In particular, Alsindi et al. [1] show in a comprehensive measurement campaign that the log-normal distribution best characterizes the NLOS error behavior. Thus, we resort to a bias  $b_u$  that is modeled as a log-normal random variable  $b_u \sim p_{\text{ln}\mathcal{N}}(\mu_u, \sigma_u)$ , supported on the semi-infinite interval  $(0, \infty)$ , and which is associated uniquely to a base station  $\mathcal{B}_u$ . For a range error defined as

$$\Delta r_{u,n} \triangleq \hat{r}_{u,n} - r_u(\mathbf{x}_n) \quad (5.5)$$

the TOA measurement model  $p_u$  describes the likelihood of  $\Delta r_{u,n}$  occurring when a robot measures a certain range distance  $\hat{r}_{u,n}$  from a base station  $\mathcal{B}_u$  at an actual position  $\mathbf{x}_n$  with a nominal (actual) range  $r_u(\mathbf{x}_n)$ . Thus, the probability density of an error  $\Delta r_u$ , occurring in a NLOS event  $\bar{L}_u$ , can be written as

$$p_u(\Delta r_{u,n} | \bar{L}_u) = (p_{\text{ln}\mathcal{N},u} * p_{\mathcal{N}})(\Delta r_{u,n}) \quad (5.6)$$

which is the convolution of the probability density function of the bias value, with the probability density function of the Gaussian noise value. Correspondingly, we can write the probability

<sup>1</sup>The terms TOA and TDOA are used interchangeably with the terms range and range difference, respectively, as they differ only by a constant factor (signal propagation speed).

density of an error  $\Delta r_{u,n}$ , occurring in a LOS event  $L_u$ , as

$$p_u(\Delta r_{u,n}|L_u) = p_{\mathcal{N}}(\Delta r_u) \quad (5.7)$$

Finally, with use of the total probability theorem, we combine the above equations to obtain the probability density of  $\Delta r_{u,n}$  as

$$p_u(\Delta r_{u,n}) = p_u(\Delta r_u|L_u) \cdot P_{L_u} + p_u(\Delta r_u|\bar{L}_u) \cdot (1 - P_{L_u}). \quad (5.8)$$

An example of this probability density function is shown in Figure 5.4a (on page 49), for two base stations.

In practice, TOA systems are rarely implemented due to the complexity induced by the required synchronization of a mobile node with the base stations. Instead, it is a common choice to implement TDOA systems which are significantly more practical, since only the synchronization among base stations is required. Thus, the direct range measurement between a mobile node and a base station is replaced by the difference between two individual range measurements each taken at a different base station. Extending the TOA formalism shown above, we define the difference range value (i.e. TDOA) between two base stations  $\mathcal{B}_u$  and  $\mathcal{B}_v$  to a target node as

$$\hat{\tau}_{uv,n} \triangleq \hat{r}_{u,n} - \hat{r}_{v,n} \quad (5.9)$$

and then easily model the TDOA error  $\Delta\tau_{uv}$  as previously shown in Equation (5.1). Simultaneously, we can describe the TDOA error as the difference between the range errors occurring at the individual base stations  $\mathcal{B}_u$  and  $\mathcal{B}_v$  as described in Equation (5.4), resulting in

$$\Delta\tau_{uv,n} = \Delta r_{u,n} - \Delta r_{v,n}. \quad (5.10)$$

Finally, we describe the probability density of a given TDOA measurement error  $\Delta\tau_{uv,n}$  as the probability density of the subtraction of two independent random variables drawn from the probability densities describing the TOA error models of the two respective base stations. We use the results of Equations (5.8) and (5.10) to model this resulting probability density as

$$p^*(\Delta\tau_{uv,n}) \triangleq (p_u * p_v^-)(\Delta\tau_{uv,n}) \quad (5.11)$$

which is a convolution of the probability density of the range error  $\Delta r_{u,n}$  and the mirrored probability density of  $\Delta r_{v,n}$  (i.e.,  $p_v^-(\Delta r_{v,n}) = p_v(-\Delta r_{v,n})$ ).

### 5.2.3 Efficient TDOA Measurement Model

Although numerical implementations for the TDOA measurement model of Equation (5.11) are easily found, they imply nested integrals which may incur a substantial computational overhead when deploying the model on a real embedded platform for real-time operation. Also, the model itself is analytically non-tractable, which causes difficulties when deriving viable estimators. For these reasons, we perform a closed-form approximation to simplify the TDOA measurement model of Equation (5.11).

Using basic algebraic properties of the convolution and inserting Equation (5.8) into Equa-

tion (5.11), we have

$$p^*(\Delta\tau_{uv,n}) = \left( P_{L_u} P_{L_v} (p_{\mathcal{N}} * p_{\mathcal{N}}^-) + P_{L_u} (1 - P_{L_v}) (p_{\mathcal{N}} * p_{\mathcal{N}}^- * p_{\ln\mathcal{N},v}^-) + \right. \\ \left. P_{L_v} (1 - P_{L_u}) (p_{\mathcal{N}} * p_{\mathcal{N}}^- * p_{\ln\mathcal{N},u}) + \right. \\ \left. (1 - P_{L_u}) (1 - P_{L_v}) (p_{\mathcal{N}} * p_{\mathcal{N}}^- * p_{\ln\mathcal{N},u} * p_{\ln\mathcal{N},v}^-) \right) (\Delta\hat{\tau}_{uv,n}). \quad (5.12)$$

It is well-known that  $p_{\mathcal{N}} * p_{\mathcal{N}}^- = p_{\sqrt{2}\mathcal{N}}$ , where  $p_{\sqrt{2}\mathcal{N}}$  is the density of a normal distribution  $\mathcal{N}(0, 2\sigma_{\mathcal{N}}^2)$ . Hence, Equation (5.12) can be rewritten as

$$p^*(\Delta\tau_{uv,n}) = \left( P_{L_u} P_{L_v} p_{\sqrt{2}\mathcal{N}} + P_{L_u} (1 - P_{L_v}) (p_{\sqrt{2}\mathcal{N}} * p_{\ln\mathcal{N},v}^-) + \right. \\ \left. P_{L_v} (1 - P_{L_u}) (p_{\sqrt{2}\mathcal{N}} * p_{\ln\mathcal{N},u}) + \right. \\ \left. (1 - P_{L_u}) (1 - P_{L_v}) (p_{\sqrt{2}\mathcal{N}} * p_{\ln\mathcal{N},u} * p_{\ln\mathcal{N},v}^-) \right) (\Delta\tau_{uv,n}). \quad (5.13)$$

UWB measurement campaigns have shown that  $\sigma_{\mathcal{N}} \ll 1$  [1, 97]. Thus  $p_{\sqrt{2}\mathcal{N}} * p_{\ln\mathcal{N}} \approx p_{\ln\mathcal{N}}$ , since the standard deviation of  $p_{\ln\mathcal{N}}$  is much larger than  $\sigma_{\mathcal{N}}$ . Furthermore, as we will numerically verify later in this section,  $p_{\ln\mathcal{N},u} * p_{\ln\mathcal{N},v}^-$  can be approximated by the density function  $p_{\tilde{\mathcal{N}}}$  of a normal distribution  $\mathcal{N}(\tilde{\mu}, \tilde{\sigma}^2)$ . The parameters  $\tilde{\mu}, \tilde{\sigma}^2$  are obtained by matching the moments (and thus minimizing the Kullback-Leibler divergence) as follows.

Let  $X_u \sim \ln\mathcal{N}(\mu_u, \sigma_u^2)$  and  $X_v \sim \ln\mathcal{N}(\mu_v, \sigma_v^2)$  be independent. For the mean  $\tilde{\mu}$  and the variance  $\tilde{\sigma}^2$ , the Kullback-Leibler divergence is minimized if  $\tilde{\mu} = \mathbb{E}[X_u - X_v]$  and  $\tilde{\sigma}^2 = \text{Var}(X_u - X_v)$ . This leads to

$$\tilde{\mu} = \mathbb{E}[X_u - X_v] = e^{\mu_u + \sigma_u^2/2} - e^{\mu_v + \sigma_v^2/2} \\ \tilde{\sigma}^2 = \text{Var}(X_u) + \text{Var}(-X_v) = e^{2\mu_u + \sigma_u^2} (e^{\sigma_u^2} - 1) + e^{2\mu_v + \sigma_v^2} (e^{\sigma_v^2} - 1). \quad (5.14)$$

Finally, using the results obtained above, we further simplify Equation (5.13) and redefine the density  $p^*$ , our TDOA error model, in closed-form as a sum of four terms:

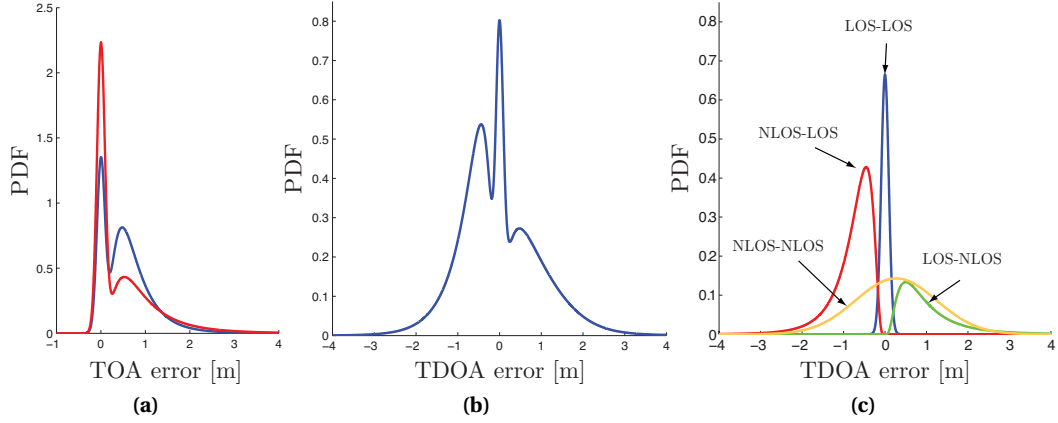
$$p(\Delta\tau_{uv,n}; \boldsymbol{\theta}_{uv,a}) = \left( P_{L_u} P_{L_v} p_{\sqrt{2}\mathcal{N}} + P_{L_u} (1 - P_{L_v}) p_{\ln\mathcal{N},v}^- + \right. \\ \left. P_{L_v} (1 - P_{L_u}) p_{\ln\mathcal{N},u} + (1 - P_{L_u}) (1 - P_{L_v}) p_{\tilde{\mathcal{N}}} \right) (\Delta\tau_{uv,n}) \quad (5.15)$$

where we introduce the parameter vector  $\boldsymbol{\theta}_{uv,a}$  as

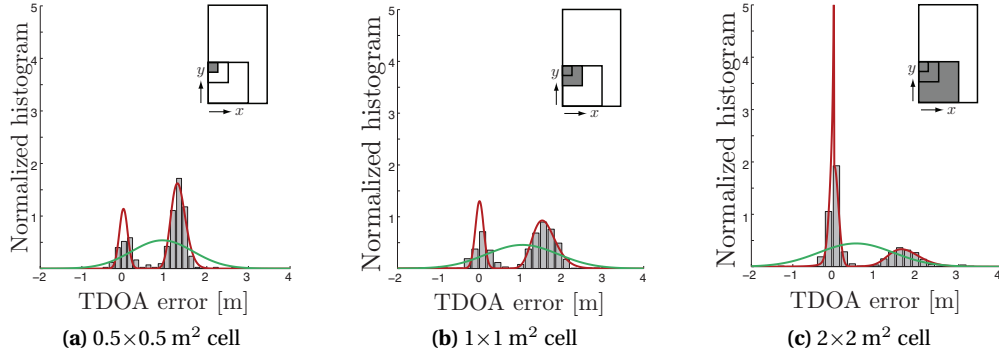
$$\boldsymbol{\theta}_{uv,a} = [\mu_u, \sigma_u, \mu_v, \sigma_v, P_{L_u}, P_{L_v}]^T \quad (5.16)$$

and  $\mu_u, \mu_v \in \mathbb{R}$ ,  $\sigma_u, \sigma_v \in \mathbb{R}^+$ , and  $P_{L_u}, P_{L_v} \in [0, 1]$ . The parameters  $\tilde{\mu}$  and  $\tilde{\sigma}$  are defined as in Equation (5.14) to complete the model (5.15).

We note that in the final form of our model (5.15), each of the four possible signal propagation configurations for a base station pair—LOS-LOS, NLOS-LOS, LOS-NLOS, and NLOS-NLOS—is represented by a term of its own. Figure 5.4b shows an example of the model in Equation (5.15). Figure 5.4c illustrates how each of the four terms of Equation (5.15) represents one of the four possible propagation modes. Figure 5.5 shows the same data as previously seen, in Figure 5.2.



**Figure 5.4:** We consider a base station pair  $\langle \mathcal{B}_1, \mathcal{B}_2 \rangle$ . The plots show an example of the probability density functions describing (a) the TOA error (Eq. (5.8)) for the two base stations and (b) the TDOA error (Eq. (5.15)). Plot (c) illustrates the four modes which form the complete multimodal probability density function shown in (b). The model parameters are set to:  $\mu_1 = -0.43$ ,  $\mu_2 = -0.2$ ,  $\sigma_1 = 0.6$ ,  $\sigma_2 = 0.7$ ,  $P_{L_1} = 0.3$ ,  $P_{L_2} = 0.5$ .



**Figure 5.5:** Histograms of TDOA error data for base station pair  $\langle \mathcal{B}_1, \mathcal{B}_2 \rangle$ . The bars are normalized so that the total area is equal to 1. We fit a Gaussian (in green) onto the data, as well as our proposed error model of Equation (5.15) (in red). The data is collected over a (a)  $0.5 \times 0.5 \text{ m}^2$  large square, (b)  $1 \times 1 \text{ m}^2$  large square, and (c)  $2 \times 2 \text{ m}^2$  large square. The areas of data collection are indicated by the shaded cells in the schematized arena, in the top right corner of each panel.

To demonstrate the efficiency of our model, we fit the parameters (5.16) to the underlying data, and superimpose the probability density onto the histogram plot. As a reference, we also superimpose a Gaussian probability density function (with mean and standard deviation calibrated on the underlying data). We note that our model provides a qualitatively very good fit. In the following section, we will quantitatively evaluate our approximation. Later in this part of the manuscript, we embed our model into a localization algorithm, and test its validity by measuring the localization performance.

Model parameter	$\mu_u$	$\sigma_u$	$P_{L_u}$
Parameter range	$[-3,0]$	$[0.2,0.8]$	$[0.01,0.99]$

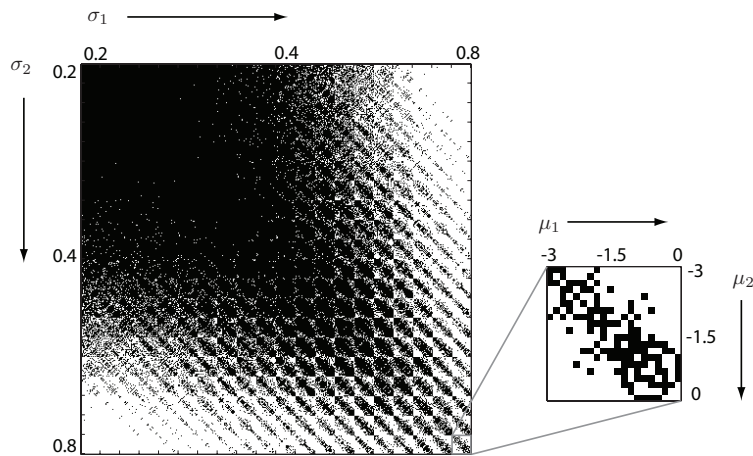
**Table 5.1:** Model parameter ranges for a base station  $\mathcal{B}_u$

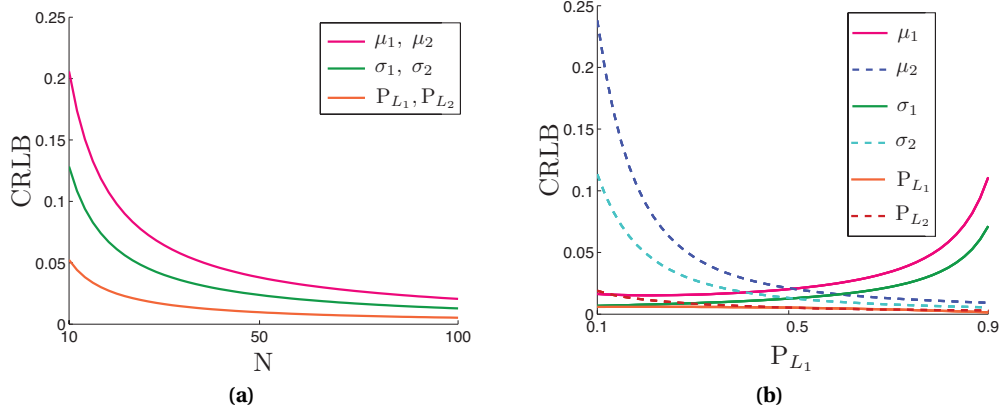
### 5.2.4 Evaluation of Approximation

In order to validate our approach, we performed a statistical comparison between the approximated and the general model. In this work, we will assume that any set of true UWB model parameter values lies in the ranges reported in Table 5.1 (an assumption supported by the experimental measurement campaigns performed in [1] and [97]). We note that these parameter ranges can be interpreted physically, as the resulting density functions quantify the occurrence of actual error values (e.g., a system with a maximum range of 10 m will not have errors beyond that value). In other words, the parameter ranges can be estimated by considering the size of the space in which the UWB system operates, and the duration of the frames within which incoming signals can be detected, and can be narrowed down for any given setup and system implementation. Ultimately, as we will see later in this manuscript, the existence of such parameter ranges facilitates the task of calibrating our models.

We first compare the approximation of  $p_{\ln \mathcal{N},u} * p_{\ln \mathcal{N},v}^-$  with the normal density function  $p_{\tilde{\mathcal{N}}}$  (defined by  $\tilde{\mu}$  and  $\tilde{\sigma}^2$  as in Equation (5.14)). For a base station pair  $\langle \mathcal{B}_u, \mathcal{B}_v \rangle$ , we vary the parameters  $\mu_u, \mu_v$  and  $\sigma_u, \sigma_v$  of the respective log-normal distributions within the ranges shown in Table 5.1 so that 80'000 tests are performed. Figure 5.6 illustrates the acceptance results of the Kolmogorov-Smirnov test at a 5% significance level. The approximation clearly performs well for small  $\sigma_u, \sigma_v$  for all  $\mu_u, \mu_v$ , with an acceptance rate of 98.5% in the parameter range  $\sigma_u, \sigma_v = [0.2, 0.55]$ . Furthermore, for two base stations with similar NLOS behavior  $|\sigma_u - \sigma_v| \leq 0.065$ , the acceptance rate is 85.5% for all  $\sigma_u, \sigma_v, \mu_u, \mu_v$ . Secondly, we compare the full model (5.11) with the approximated model (5.15) over 800'000 tests measuring the Kolmogorov-Smirnov distance, for parameters  $\sigma_u, \sigma_v, \mu_u, \mu_v, P_{L_u}$  and  $P_{L_v}$  in the ranges as reported in Table 5.1. The average Kolmogorov-Smirnov distance over all tests is 0.07, showing a good match of the densities.

**Figure 5.6:** Acceptance results of the Kolmogorov-Smirnov test at a 5% significance level, for  $400^2$  parameter variations (20 values for both  $\mu_u$  and  $\sigma_u$ , in the ranges reported in Table 5.1). Accepted tests are shown in black, rejected tests are shown in white.





**Figure 5.7:** CRLB for (a) a varying number of data samples  $N$  and (b) for varying LOS probability  $P_{L1}$ , for default parameter values  $\mu_{1,2} = -0.4$ ,  $\sigma_{1,2} = 0.6$ , and  $P_{L1,2} = 0.5$ .

### 5.2.5 Cramér Rao Lower Bound

To conclude this chapter on modeling, we will present the CRLB for the efficient TDOA measurement model of Equation (5.15). The CRLB provides a lower bound on the variance of unbiased estimators and thereby provides a useful tool to assess the quality of algorithms which estimate the parameters of our model. Also, the facility of calibrating a given model has an important influence on the model's overall efficiency, and it is useful to understand how the lower estimation bound reacts as a function of the model parameters.

Let us denote by  $\hat{\tau}_{uv}^{(k)}$  the  $k$ -th measured TDOA value, and by  $N$  the total number of measured values. Given the associated true values  $\tau_{uv}^{(k)}$ , our data is  $(\Delta\tau_{uv}^{(1)}, \Delta\tau_{uv}^{(2)}, \dots, \Delta\tau_{uv}^{(N)})$ . In this derivation we consider the parameters of our measurement model

$$\boldsymbol{\theta} = [\mu_u, \sigma_u, \mu_v, \sigma_v, P_{L_u}, P_{L_v}]^T \quad (5.17)$$

as previously seen in Equation (5.16). In order to determine the CRLB, we need to calculate the Fisher Information Matrix (FIM)  $\mathbf{I}(\boldsymbol{\theta})$ . Since the observations  $\hat{\tau}_{uv}^{(k)}$  are independent, the FIM is then, by definition, the  $6 \times 6$  matrix with entries

$$[\mathbf{I}(\boldsymbol{\theta})]_{i,j} := - \sum_{k=1}^N \mathbb{E} \left[ \frac{\partial^2 \log p(\Delta\tau_{uv}^{(k)}; \boldsymbol{\theta})}{\partial \theta_i \partial \theta_j} \right]. \quad (5.18)$$

Since the observations are identically distributed, we have

$$\begin{aligned} [\mathbf{I}(\boldsymbol{\theta})]_{i,j} &= -N \mathbb{E} \left[ \frac{\partial^2 \log p(\Delta\tau_{uv})}{\partial \theta_i \partial \theta_j} \right] \\ &= -N \int_{-\infty}^0 \frac{\partial^2 \log p(x)}{\partial \theta_i \partial \theta_j} p(x) dx - N \int_0^{\infty} \frac{\partial^2 \log p(x)}{\partial \theta_i \partial \theta_j} p(x) dx. \end{aligned} \quad (5.19)$$

Recall that for fixed  $x \in (0, \infty)$  we have

$$p(x) = P_{L_u} P_{L_v} p_{\sqrt{2}\mathcal{N}}(x) + P_{L_v} (1 - P_{L_u}) p_{\ln \mathcal{N}, u}(x) + (1 - P_{L_u}) (1 - P_{L_v}) p_{\tilde{\mathcal{N}}}(x) \quad (5.20)$$

and for  $x \in (-\infty, 0)$  we have

$$p(x) = P_{L_u} P_{L_v} p_{\sqrt{2}\mathcal{N}}(x) + P_{L_u} (1 - P_{L_v}) p_{\ln \mathcal{N}, v}(-x) + (1 - P_{L_u}) (1 - P_{L_v}) p_{\tilde{\mathcal{N}}}(x). \quad (5.21)$$

Using Equation (5.21) for the first term and Equation (5.20) for the second term of Equation (5.19), it is now straightforward to deduce a closed-form expression for its integrand. Then, due to the complicated structure of this integrand, we are not able to perform the integration analytically but (for given  $\theta$ ) only numerically. Finally, for any unbiased estimator  $\hat{\theta} = (\hat{\theta}_1, \dots, \hat{\theta}_6) = (\hat{\mu}_u, \dots, \hat{P}_{L_v})$  for  $\theta$ , we have the CRLB

$$\text{Var}(\hat{\theta}_i) \geq [\mathbf{I}(\theta)^{-1}]_{i,i}, \quad (5.22)$$

which is the inversion of the  $6 \times 6$  matrix found in (5.19), numerically evaluated at  $\hat{\theta}$ .

Figure 5.7 shows the CRLB for our model and two base stations  $\langle \mathcal{B}_1, \mathcal{B}_2 \rangle$ , (a) for a varying number of data samples  $N$ , and (b) for a varying LOS probability  $P_{L_1}$ . As is to be expected, the CRLB decreases for increasing  $N$ . Also, we see that as  $P_{L_1}$  increases, and base station  $\mathcal{B}_1$  is more likely to be in LOS, it becomes harder to estimate the log-normal parameters  $\mu_1$  and  $\sigma_1$  associated to its NLOS bias. Conversely, as  $P_{L_1}$  decreases, and base station  $\mathcal{B}_1$  is more likely to be in NLOS, it becomes harder to estimate the log-normal parameters  $\mu_2$  and  $\sigma_2$  associated to the NLOS bias of  $\mathcal{B}_2$ . These insights are supported by Figure 5.2c: if we assume we ‘know’ if  $\mathcal{B}_1$  is in LOS or NLOS, we can determine the modes that we are trying to estimate. Overlapping modes create more uncertainty. Indeed, the NLOS-NLOS mode is the hardest to estimate, also because it depends on the largest amount of free parameters ( $\mu_1, \mu_2, \sigma_1, \sigma_2$ , and  $P_{L_2}$ ).

### Summary

In this chapter, we derived our UWB error model. Based on our assumptions, we showed how to approximate an analytically intractable solution to produce a closed-form, efficient model, which can easily be implemented on resource constrained platforms. The design of our model is based on basic NLOS propagation characteristics, which consistently induce a positive bias in the measurement value. Furthermore, the final form of our model reflects the four possible signal propagation configurations that a base station pair can be subjected to (LOS-LOS, NLOS-LOS, LOS-NLOS, and NLOS-NLOS). Our analyses have shown that the approximative model hardly deviates from the analytically complete model. Finally, we have discussed the estimation properties of our proposed model by deriving its Cramér Rao Lower Bound.



## 6 Algorithms

A good model by itself does not necessarily fulfill a sensor's use case. Often, what follows the development of a model, is the development of efficient and robust calibration methods, and ways to embed the model in run-time algorithms. This chapter focuses on those two aspects. Firstly, we consider two alternatives for model calibration: a batch (offline) algorithm, and an online algorithm. We show how these estimation methods are embedded in a mapping function, ultimately producing a *spatial* UWB measurement model. Secondly, we motivate our localization framework, which is based on a particle filter, and show how we embed the measurement model into this routine. This chapter is concluded with an evaluation of our system's localization performance.

### 6.1 General Localization Framework

There are various ways in which to approach a localization problem algorithmically. In a nutshell, the ultimate goal is to provide an estimate, or belief, of the state (pose/position) of a mobile target, given sensor measurements that are taken along its way. Localization algorithms integrate these sensor measurements in such a way that a *posterior* belief of the state is formulated. In the following sections, we aim to motivate our choice of localization algorithm given the particular context.

#### 6.1.1 Preliminaries

Localization methods can be separated into two broad categories—those which describe the posterior belief parametrically, and those which don't. The Kalman filter is a prominent state estimation algorithm which has found a strong foothold in localization literature. The Kalman filter (KF), as well as its widely used non-linear counterpart, the extended Kalman filter (EKF), represent the posterior belief of a target's pose with a Gaussian, where state space is continuous. Assumptions, such as Gaussian noise and an initial position with Gaussian uncertainty, make it a viable choice for position tracking problems. The extended multi-hypotheses KF relaxes restrictions on the posterior belief by using mixtures of Gaussians. This allows for a multi-hypotheses representation of the posterior belief. For both the KF and EKF filters, in order to accommodate the assumption of Gaussian distributed sensor noise, raw sensor data is

processed to produce low-dimensional features. This may implicate that relevant data is discarded during the process of feature extraction. Another well-known parametric approach is grid-based localization, which uses a histogram filter to approximate the posterior over a tessellated state space (fine-grained grids or topological maps). The belief is thus represented by piecewise constant functions. Here, assumptions are yet again relaxed since sensor noise may or may not be assumed Gaussian. Finally, Monte Carlo Localization (MCL) [119], which is based on particle filter algorithms, arguably represents the most popular implementation in the class of non-parametric methods. Posterior distributions are represented by sets of weighted samples, and thus approximate a universal density. This method has proven very successful to date, the main reasons for this being (i) accommodation of arbitrary sensor noise, (ii) representation of arbitrary pose distributions, and (iii) adaptability to computer resources.

Given the context of a UWB sensor modality with non-Gaussian characteristics, our method of choice is the MCL framework. MCL, as many other localization techniques, relies on the formalism of the Bayes filter. The Bayes filter is an algorithm which estimates the probability density of a posterior state, conditioned on data collected by the robot. Let us from hereon consider a robot  $\mathcal{R}_n$ . At time  $t$ , after a sequence of motion control actions  $u_{n,t}$  and a sequence of observations  $z_{n,t}$ , the filter is denoted

$$\text{Bel}(\mathbf{x}_{n,t}) = p(\mathbf{x}_{n,t} | z_{n,0..t}, u_{n,0..t}) \quad (6.1)$$

where  $\text{Bel}(\mathbf{x}_{n,t})$  estimates of the posterior state  $\mathbf{x}_{n,t}$  and is called a *belief*. Then, derived from the Bayes rule and given the Markov assumption, the Bayes filter can be expressed in the form of the update equation

$$\text{Bel}(\mathbf{x}_{n,t}) = \eta p(z_{n,t} | \mathbf{x}_{n,t}) \int p(\mathbf{x}_{n,t} | \mathbf{x}_{n,t-1}, u_{n,t-1}) \text{Bel}(\mathbf{x}_{n,t-1}) d\mathbf{x}_{n,t-1} \quad (6.2)$$

with  $\eta$  such that the cumulative belief over all possible states  $\mathbf{x}_{n,t}$  amounts to 1. In this case,  $\eta = p(z_{n,t} | u_{n,t-1}, \dots, z_{n,0})^{-1}$ . The implementation of this formula requires the knowledge of  $p(z_{n,t} | \mathbf{x}_{n,t})$ , the measurement model, and  $p(\mathbf{x}_{n,t} | \mathbf{x}_{n,t-1}, u_{n,t-1})$ , the motion model. The measurement model describes the probability of making a certain observation at a certain position. The way this model is constructed is inherently dependent on the considered sensor type, as it attempts to reproduce the way sensor measurements are generated in a given environment. The motion model describes the state transitions of a mobile robot, and can, for instance, be based on velocity or odometry information. Typically, the model will also integrate noise parameters to take into account for the uncertainty in the actual robot translation and rotation quantities.

The main idea of MCL lies in the way the belief is represented—samples, or particles, are drawn from the posterior probability distribution of the robot pose to form a set of particles. In contrast to other methods (for example Kalman filtering), the advantage of this form of representation is that it can approximate probability densities of any shape. Given this flexibility, MCL is also able to accommodate arbitrary sensor characteristics and noise distributions. Lastly, by tuning the total number of particles used, MCL can adapt to variable computational capacities, at the cost of accuracy and precision.

### 6.1.2 The Particle Filter

As described above, the posterior distribution of MCL is represented by a set of particles. By weighting these particles one obtains a discrete probability function that approximates the continuous belief  $\text{Bel}(\mathbf{x}_{n,t})$ , and hence we have

$$\text{Bel}(\mathbf{x}_{n,t}) \sim \{\langle \mathbf{x}_{n,t}^{[i]}, w_{n,t}^{[i]} \rangle | i = 1, \dots, M\} = X_{n,t} \quad (6.3)$$

where  $M$  is the number of particles,  $\mathbf{x}_{n,t}^{[i]}$  is a sample of the random variable  $\mathbf{x}_{n,t}$ , the state of the robot  $\mathcal{R}_n$  ( $\mathbf{x}_{n,t} = [x_{n,t}, y_{n,t}, \psi_{n,t}]^T$ , where  $x_{n,t}$  and  $y_{n,t}$  are Euclidean coordinates and  $\psi_{n,t}$  is the orientation), and  $w_{n,t}^{[i]}$  is its weight. The symbol  $X_{n,t}$  refers to the set of particles  $\langle \mathbf{x}_{n,t}^{[i]}, w_{n,t}^{[i]} \rangle$  at time  $t$  belonging to robot  $\mathcal{R}_n$ . At the beginning of a localization exercise, the set of particles  $X_{n,0}$  is initialized: if the initial pose is known, the particles are drawn from a probability distribution centered around the given pose, and if the initial pose is unknown, the particles are drawn from a uniform distribution covering the whole environment in which the robot is supposed to operate. Then, in order to estimate the robot's pose at each time step, a recursive update is made. Stated briefly, this update is composed of three parts:

- (1) sample particle  $\mathbf{x}_{n,t-1}^{[i]}$  from proposal distribution  $\text{Bel}(\mathbf{x}_{n,t-1})$
- (2) sample  $\mathbf{x}_{n,t}^{[i]}$  according to  $\mathbf{x}_{n,t}^{[i]} \sim p(\mathbf{x}_{n,t} | \mathbf{x}_{n,t-1}^{[i]}, u_{n,t-1})$
- (3) allocate weight  $w_{n,t}^{[i]}$  to sample  $\mathbf{x}_{n,t}^{[i]}$  with  $w_{n,t}^{[i]} = p(z_{n,t} | \mathbf{x}_{n,t}^{[i]})$ .

where (2) consists of the application of the motion model, essentially propagating the particle, and (3) consists of the application of the measurement model. These steps, when performed for the whole set of  $M$  particles, can be formulated as an algorithm: Algorithm 1 (as seen in [119]) performs this procedure for a given robot  $\mathcal{R}_n$ . In our work, we implement the sampling routine in line 8 with the low-variance sampling algorithm (described in [119]). Although many resampling algorithms exist, this particular implementation is interesting for particle filters, as it avoids particle depletion: the first particle is drawn according to a single random number, and all subsequent draws are dependent on the first draw (as well as the individual particle weights). This method ensures that given a uniformly weighted particle set, the resulting

---

**Algorithm 1**  $\text{MCL}(X_{n,t-1}, u_{n,t}, z_{n,t})$ 


---

```

1:  $\bar{X}_{n,t} = X_{n,t} = \emptyset$ 
2: for  $i = 1$  to  $M$  do
3:    $\mathbf{x}_{n,t}^{[i]} \leftarrow \text{Motion\_Model}(u_{n,t}, \mathbf{x}_{n,t-1}^{[i]})$ 
4:    $w_{n,t}^{[i]} \leftarrow \text{Measurement\_Model}(z_{n,t}, \mathbf{x}_{n,t}^{[i]})$ 
5:    $\bar{X}_{n,t} \leftarrow \bar{X}_{n,t} \cup \langle \mathbf{x}_{n,t}^{[i]}, w_{n,t}^{[i]} \rangle$ 
6: end for
7: for  $i = 1$  to  $M$  do
8:    $\mathbf{x}_{n,t}^{[i]} \leftarrow \text{Sampling}(\bar{X}_{n,t})$ 
9:    $X_{n,t} \leftarrow X_{n,t} \cup \langle \mathbf{x}_{n,t}^{[i]}, \frac{1}{M} \rangle$ 
10: end for
11: return  $X_{n,t}$ 
    
```

---

sampled particle set remains unchanged. The computational complexity (as a function of the number of particles) of Algorithm 1 is  $O(M)$ .

### 6.1.3 Localization using UWB Measurements

In chapter 5, we established the UWB error model  $p(\Delta\tau_{uv,n,t}; \theta_{uv,a})$ . The reader will recall from Section 5.1 that, in addition to a UWB measurement model, our modeling approach includes a set of maps  $\mathcal{M}$ . For now, let us assume that the set of maps  $\mathcal{M}$  is known. Thus, for a given position  $\mathbf{x}_{n,t}$  and a base station pair  $\langle \mathcal{B}_u, \mathcal{B}_v \rangle$ , we apply our mapping function of Equation (5.2) to retrieve the parameter vector  $\theta_{uv,a}$  used to formulate the probability density function  $p(\Delta\tau_{uv,n,t}; \theta_{uv,a})$ . Hence, for a set of TDOA measurements  $T_{n,t}$  taken at time  $t$ , the likelihood of a given pose sample  $\mathbf{x}_{n,t}^{[i]}$  is

$$P(\mathbf{x}_{n,t}^{[i]}) = \eta \cdot \prod_{\langle \hat{\tau}_{uv,n,t}, \langle \mathcal{B}_u, \mathcal{B}_v \rangle \rangle \in T_{n,t}} p(\Delta\tau_{uv}(\hat{\tau}_{uv,n,t}, \mathbf{x}_{n,t}^{[i]}; m_{uv}(\mathbf{x}_{n,t}^{[i]})), \quad (6.4)$$

where  $\eta$  is a normalization factor. This likelihood function is formulated algorithmically in Algorithm 2: the weight of the particle  $i$  is updated to take into account the new set of measurements.

Algorithm 3 shows a complete localization algorithm accommodating UWB measurements. Line 3 shows the application of the motion model, where  $u_{n,t}$  represents dead-reckoning information. Line 4 shows the application of the measurement model where  $T_{n,t}$  represents the TDOA data and  $\mathcal{M}$  is the map. This measurement model replaces the template measurement model of Algorithm 1, on line 4. Practically, however, the application of our new measurement model is preceded by a mapping step, which *builds* the maps in  $\mathcal{M}$ . We will discuss how to construct these maps later, in Section 6.3.

---

**Algorithm 2** Measurement\_Model\_Map( $T_{n,t}, \mathcal{M}, \mathbf{x}_{n,t}^{[i]}, w_{n,t-1}^{[i]}$ )

---

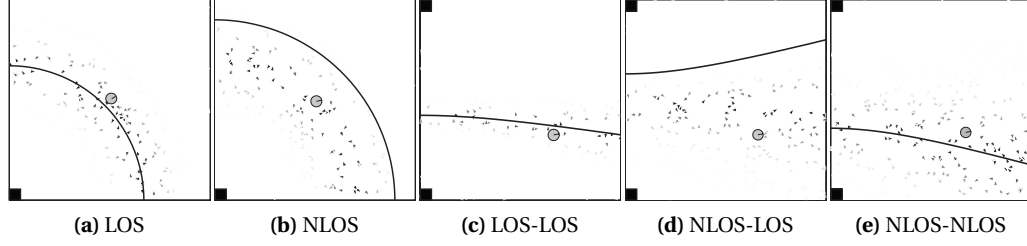
- 1:  $w_{n,t}^{[i]} \leftarrow w_{n,t-1}^{[i]} \cdot P(\mathbf{x}_{n,t}^{[i]})$
  - 2: return  $w_{n,t}^{[i]}$
- 

---

**Algorithm 3** UWB\_MCL( $X_{n,t-1}, u_{n,t}, T_{n,t}$ )

---

- 1:  $\bar{X}_{n,t} = X_{n,t} = \emptyset$
  - 2: **for**  $i = 1$  to  $M$  **do**
  - 3:  $\mathbf{x}_{n,t}^{[i]} \leftarrow \text{Motion\_Model}(u_{n,t}, \mathbf{x}_{n,t-1}^{[i]})$
  - 4:  $w_{n,t}^{[i]} \leftarrow \text{Measurement\_Model\_Map}(T_{n,t}, \mathcal{M}, \mathbf{x}_{n,t}^{[i]}, w_{n,t-1}^{[i]})$
  - 5:  $\bar{X}_{n,t} \leftarrow \bar{X}_{n,t} \cup \langle \mathbf{x}_{n,t}^{[i]}, w_{n,t}^{[i]} \rangle$
  - 6: **end for**
  - 7: **for**  $i = 1$  to  $M$  **do**
  - 8:  $\mathbf{x}_{n,t}^{[i]} \leftarrow \text{Sampling}(\bar{X}_{n,t})$
  - 9:  $X_{n,t} \leftarrow X_{n,t} \cup \langle \mathbf{x}_{n,t}^{[i]}, \frac{1}{M} \rangle$
  - 10: **end for**
  - 11: return  $X_{n,t}$
-



**Figure 6.1:** Illustration of TOA and TDOA measurement models within a particle filter. The base stations are schematically marked in the panel corners. The robot body shows the actual robot position. The pose estimates (particles) are represented by triangles with increasing transparency for decreasing weights. The curved lines represent the noisy/biased TOA range measurements in (a)-(b) and noisy/biased TDOA measurements in (c)-(e). The application of the TOA measurement model for a single base station is illustrated in (a) for a LOS condition and (b) for a NLOS condition. The application of the TDOA measurement model for a single base station pair is shown in (c) for both base stations in LOS conditions, (d) for the bottom base station in NLOS, the top base station in LOS conditions, and (e) for both base stations in NLOS conditions.

Figure 6.1 illustrates the application of the measurement model on a set of particles, first for the more intuitive case of TOA (6.1a and 6.1b), and then for TDOA (6.1c—6.1e). Figures 6.1a and 6.1b show an application of the UWB TOA error model, weighting particles in (a) a LOS scenario and (b) a NLOS scenario, for a single base station. Figures 6.1c—6.1e show an application of the UWB TDOA error model, weighting particles in (c) a LOS condition for both base stations, (d) a LOS condition for the top base station and a NLOS condition for the bottom base station and (e) a NLOS condition for both base stations. We observe that the models produce particles with high weights in the area of the actual robot position, even when the measurement is biased. Even though the models are able to capture all possible conditions, information on the actual path condition (LOS/NLOS) is essential to obtain good localization. Finally, we note that with the addition of measurements obtained from other base station (pairs), the set of particles with high weights is additionally confined. A summarized position estimate can then be made by considering a predefined metric (e.g., the center of mass, or the particle with the highest weight).

## 6.2 Estimation of Model Parameters

Due to its efficiency in achieving the CRLB for data set sizes tending to infinity, our approach to estimate  $\theta_{uv,a}$  is based on Maximum Likelihood Estimation (MLE). Although several methods can be used to obtain the maximum likelihood estimate, we implement an EM approach. Indeed, for our model, the closed-form solution to the maximum likelihood problem does not exist. The following derivations will show how our model is reformulated as a mixture model, and how this reformulation leads to an efficient implementation of the EM algorithm. In our particular case, we will show that our EM formalism produces a compact, elegant closed-form expression. This constitutes one of our most important results, as the existence of such a compact formalism ultimately enables the portability of our method onto computationally constrained devices. In the following, we will derive an online as well as a batch (offline) expression.

### 6.2.1 Derivation

For brevity, we omit the robot index  $n$  in the following derivations.  $K$  denotes the number of observations, and  $k$  denotes the observation index. Note that here (and in what follows) there is no dependency on  $k$ , since the observations are i.i.d. The maximum likelihood estimator for our model is defined by

$$\hat{\theta}_{uv} = \arg \max_{\theta_{uv,a}} \frac{1}{K} \sum_{k=1}^K \log p(\Delta\tau_{uv,k}; \theta_{uv,a}). \quad (6.5)$$

First, let us simplify our model (5.15) by reformulating its four terms as

$$\begin{aligned} f_1(x; \theta_{uv,a}) &= p_{\sqrt{2}\mathcal{N}}(x; \theta_{uv,a}) = p_{\sqrt{2}\mathcal{N}}(x) \\ f_2(x; \theta_{uv,a}) &= p_{\ln\mathcal{N},v}^-(x; \theta_{uv,a}) = p_{\ln\mathcal{N},v}^-(x; \mu_v, \sigma_v) \\ f_3(x; \theta_{uv,a}) &= p_{\ln\mathcal{N},u}(x; \theta_{uv,a}) = p_{\ln\mathcal{N},u}(x; \mu_u, \sigma_u) \\ f_4(x; \theta_{uv,a}) &= p_{\tilde{\mathcal{N}}}(x; \theta_{uv,a}) = p_{\tilde{\mathcal{N}}}(x; \tilde{\mu}, \tilde{\sigma}). \end{aligned} \quad (6.6)$$

By defining  $\alpha_1 = P_{L_u}P_{L_v}$ ,  $\alpha_2 = P_{L_u}(1 - P_{L_v})$ ,  $\alpha_3 = P_{L_v}(1 - P_{L_u})$ , and  $\alpha_4 = (1 - P_{L_u})(1 - P_{L_v})$ , we can rewrite the model  $p$  in the form of a standard mixture model

$$p(x; \theta_{uv,a}) = \sum_{j=1}^4 \alpha_j f_j(x; \theta_{uv,a}). \quad (6.7)$$

Thus, by assuming that every observation  $\Delta\tau_{uv,k}$  originates from an  $f_j$ , we postulate the existence of a latent variable  $\mathbf{Z} = (Z_1, \dots, Z_N)$  where  $Z_k = j$  with probability  $\alpha_j$  and  $j \in \{1, 2, 3, 4\}$ .  $Z_k$  specifies which  $f_j$  the  $k$ -th observation corresponds to, thus, given  $Z_k = j$ , the observation  $\Delta\tau_{uv,k}$  has density  $f_j$ . In other words, any given data sample corresponds to the probability density defined by two base stations in either LOS-LOS, NLOS-NLOS, LOS-NLOS, or NLOS-LOS configuration. For any  $k$ , the likelihood of  $(\Delta\tau_{uv,k}, Z_k)$  is

$$f(x, z; \theta_{uv,a}) = \sum_{j=1}^4 \alpha_j \delta_{zj} f_j(x; \theta_{uv,a}) \quad (6.8)$$

with  $\delta_{zj} = 1$  if  $z = j$ , and  $\delta_{zj} = 0$  otherwise. Our new formulation is a mixture model, with  $z$  specifying the mixture component that a data point belongs to.

Given our new formulation, we can conveniently leverage the EM framework to estimate our parameter vector  $\theta_{uv,a}$ . The following notations follow the conventions introduced by Cappé et al. [19]. We are now able to write  $f$  in exponential family form:

$$f(x, z; \theta_{uv,a}) = h(x, z) \exp \left( \sum_{j=1}^{11} S_j(x, z) \phi_j(\theta_{uv,a}) \right) \quad (6.9)$$

where  $h(x, z) = 0$  if  $z = 2$  and  $x \geq 0$ , or  $z = 3$  and  $x \leq 0$ , and  $h(x, z) = 1$  otherwise.

$S = (S_1, \dots, S_{11})$  and  $\phi = (\phi_1, \dots, \phi_{11})$  are defined as follows:

$$\begin{aligned}
 S(x, z) = & [\delta_{z1}, x^2 \delta_{z1}, \chi_{(-\infty, 0)}(x) \delta_{z2}, \chi_{(-\infty, 0)}(x) \log(-x) \delta_{z2}, \\
 & \chi_{(-\infty, 0)}(x) \log(-x)^2 \delta_{z2}, \chi_{(0, \infty)}(x) \delta_{z3}, \chi_{(0, \infty)}(x) \log(x) \delta_{z3}, \\
 & \chi_{(0, \infty)}(x) \log(x)^2 \delta_{z3}, \delta_{z4}, x \delta_{z4}, x^2 \delta_{z4}]^T \\
 [\phi_1, \phi_2]^T(\theta_{uv,a}) = & \left[ \log(\alpha_1) - \log(2\sqrt{\pi}\sigma), -\frac{1}{4\sigma_{\mathcal{N}}^2} \right]^T \\
 [\phi_3, \phi_4, \phi_5]^T(\theta_{uv,a}) = & \left[ -\log(2\sqrt{\pi}\sigma_v) + \log(\alpha_2) - \frac{\mu_v^2}{2\sigma_v^2}, -1 + \frac{\mu_v}{\sigma_v^2}, -\frac{1}{2\sigma_v^2} \right]^T \\
 [\phi_6, \phi_7, \phi_8]^T(\theta_{uv,a}) = & \left[ -\log(2\sqrt{\pi}\sigma_u) + \log(\alpha_3) - \frac{\mu_u^2}{2\sigma_u^2}, -1 + \frac{\mu_u}{\sigma_u^2}, -\frac{1}{2\sigma_u^2} \right]^T \\
 [\phi_9, \phi_{10}, \phi_{11}]^T(\theta_{uv,a}) = & \left[ -\log(2\sqrt{\pi}\tilde{\sigma}) + \log(\alpha_4) - \frac{\tilde{\mu}^2}{2\tilde{\sigma}^2}, \frac{\tilde{\mu}}{\tilde{\sigma}^2}, -\frac{1}{2\tilde{\sigma}^2} \right]^T
 \end{aligned} \tag{6.10}$$

where  $\chi_{[\cdot]}$  is the indicator function for a given interval. We define weights  $\bar{w}_j(x; \theta_{uv,a})$  obtained through Bayes' theorem as

$$\bar{w}_j(x; \theta_{uv,a}) := \frac{\alpha_j f_j(x; \theta_{uv,a})}{\sum_{m=1}^4 \alpha_m f_m(x; \theta_{uv,a})} \tag{6.11}$$

which is equal to the probability that  $Z_k = j$  given  $\Delta\tau_{uv,k} = x$ . Then for  $\bar{s}(x; \theta_{uv,a}) := E_{\theta_{uv,a}}[S(\Delta\tau_{uv,k}, Z_k) | \Delta\tau_{uv,k} = x]$  we have

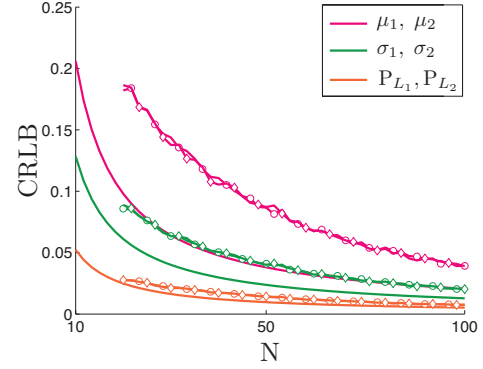
$$\begin{aligned}
 \bar{s}(x; \theta_{uv,a}) = & [\bar{w}_1(x; \theta_{uv,a}), \bar{w}_1(x; \theta_{uv,a})x^2, \bar{w}_2(x; \theta_{uv,a}), \\
 & \bar{w}_2(x; \theta_{uv,a}) \log(-x), \bar{w}_2(x; \theta_{uv,a}) \log(-x)^2, \\
 & \bar{w}_3(x; \theta_{uv,a}), \bar{w}_3(x; \theta_{uv,a}) \log(x), \bar{w}_3(x; \theta_{uv,a}) \log(x)^2, \\
 & \bar{w}_4(x; \theta_{uv,a}), \bar{w}_4(x; \theta_{uv,a})x, \bar{w}_4(x; \theta_{uv,a})x^2]^T.
 \end{aligned} \tag{6.12}$$

Finally, for a given vector  $s \in \mathbb{R}^{11}$  we define the function  $l(s; \theta_{uv,a}) = \sum_{j=1}^{11} s_j \phi_j(\theta_{uv,a})$ . Through straightforward calculation we can derive conditions on  $s$  so that we can define the function  $\bar{\theta}_{uv,a}(s) := \arg \max_{\theta_{uv,a}} l(s; \theta_{uv,a})$ , which leads to

$$\begin{aligned}
 \bar{\theta}_{uv,a}(s) = & \left[ \frac{s_7}{s_6}, \sqrt{\frac{-s_7^2 + s_6 s_8}{s_6^2}}, \frac{s_4}{s_3}, \sqrt{\frac{-s_4^2 + s_3 s_5}{s_3^2}}, \frac{s_{10}}{s_9}, \sqrt{\frac{-s_{10}^2 + s_9 s_{11}}{s_9^2}}, \right. \\
 & \left. \frac{s_1 + s_3}{s_1 + s_3 + s_6 + s_9}, \frac{s_1 + s_6}{s_1 + s_3 + s_6 + s_9} \right]^T.
 \end{aligned} \tag{6.13}$$

We note that Equations (6.11) and (6.12) relate to the E-step and that Equation (6.13) relates to the M-step of a standard, offline EM algorithm. As we will see in the next two paragraphs, the implementations of the batch and online estimation algorithms now only require the evaluation of these closed-form vectors.

**Figure 6.2:** CRLB for a varying number of data samples  $N$  for default parameter values  $\mu_{1,2} = -0.4$ ,  $\sigma_{1,2} = 0.6$ , and  $P_{L1,2} = 0.5$ . Additionally, the panel shows the average variance of the batch estimator, evaluated over 1000 runs. Symbols  $\diamond$  and  $\circ$  show parameter estimates for base station  $\mathcal{B}_1$  and  $\mathcal{B}_2$ , respectively.



### 6.2.2 Batch Estimation Algorithm

Using the notations introduced above, for  $N$  data samples, the  $k+1$ -th parameter estimate  $\hat{\theta}^{(k+1)}$  in the batch EM algorithm is given by

$$\hat{\theta}_{uv,a}^{(k+1)} = \bar{\theta} \left( \frac{1}{N} \sum_{i=1}^N \bar{s}(\Delta\tau_{uv}^{(i)}; \hat{\theta}_{uv,a}^{(k)}) \right). \quad (6.14)$$

Figure 6.2 shows the CRLB, as seen previously in Figure 5.7. Additionally, we superimpose the average variance of the batch estimator, evaluated over 1000 runs. As expected, the variance tends towards the CRLB for  $N \rightarrow \infty$ , confirming its efficiency for growing  $N$ .

### 6.2.3 Online Estimation Algorithm

For some applications, it is beneficial to use an estimation algorithm that produces estimates in real-time. For this reason, we resort to the results of Cappé et al. [19], who present an online formalism for the EM problem. For observations  $(\Delta\tau_{uv,1}, \dots, \Delta\tau_{uv,k})$ , the  $k+1$ -th parameter estimate  $\hat{\theta}_{uv,a}^{(k+1)}$  in the online EM algorithm takes the form

$$\begin{aligned} \hat{s}^{(k+1)} &= \hat{s}^{(k)} + \gamma^{(k+1)} \left[ \bar{s}(\Delta\tau_{uv,k}; \hat{\theta}_{uv,a}^{(k)}) - \hat{s}^{(k)} \right] \\ \hat{\theta}_{uv,a}^{(k+1)} &= \bar{\theta}_{uv,a}(\hat{s}^{(k+1)}) \end{aligned} \quad (6.15)$$

where  $\gamma^{(k+1)}$  is a user-defined step size. Variations of  $\gamma^{(k)} = 1/k$  have shown to produce good convergence speed—typically, the choice of  $\gamma$  defines the trade-off between adaptability and stability of the estimate (the work by [19] demonstrates that this baseline online EM framework almost surely converges). In particular, bounding  $\gamma$  from below can ensure continuous adaptation of the model. We note that since the second and third term of our mixture model are defined by densities that are only supported on a semi-infinite interval, we do not update the entries of  $\hat{s}_{3,4,5}^{(k+1)}$  when  $\Delta\tau_{uv}^{(k+1)} \in (0, \infty)$  and similarly of  $\hat{s}_{6,7,8}^{(k+1)}$  when  $\Delta\tau_{uv,k+1} \in (-\infty, 0)$ . Finally, we refer the reader to Algorithm 4 on page 64, where line 6 corresponds to the application of Equation (6.15). The real-time estimation of UWB error maps constitutes an example application of the online EM algorithm.



### 6.2.4 Performance

We perform several tests to assess the efficiency of our estimation algorithms. Our first test consists of a qualitative comparison of the online and batch estimation algorithms, on a set of 2000 data points gathered in our experimental setup. Figure 6.3 illustrates the convergence of the online EM parameter estimates. In this example, base station  $\mathcal{B}_1$  is in moderate NLOS, with  $P_{L_1} = 0.56$ , and base station  $\mathcal{B}_2$  is in LOS with  $P_{L_2} = 0.98$ . Due to the large spread of NLOS data points, parameters associated to base station  $\mathcal{B}_1$  undergo larger oscillations than those associated to base station  $\mathcal{B}_2$ . Figure 6.4 summarizes the resulting estimated probability density functions, and superimposes the normalized histogram of the collected data points. The panel shows a qualitatively good fit to the underlying samples.

Figure 6.5 quantifies the efficiency of the online and batch algorithms. We performed a set of 1000 simulations, where for each simulation, true parameters  $\theta_{uv,a}$  were sampled randomly in the intervals as reported in Table 5.1, and used in the non-approximated model, Equation (5.11), to generate a sample size of  $N = 1000$ . For the online algorithm, the final estimate of  $\hat{\theta}_{uv,a}$  is found by averaging the last 50 estimates. The results confirm good estimation of the model parameters. The Kolmogorov-Smirnov distance shown in Figure 6.5b indicates that both estimation algorithms produce a good model of the underlying data, and that the batch algorithm is slightly more efficient than the online algorithm.

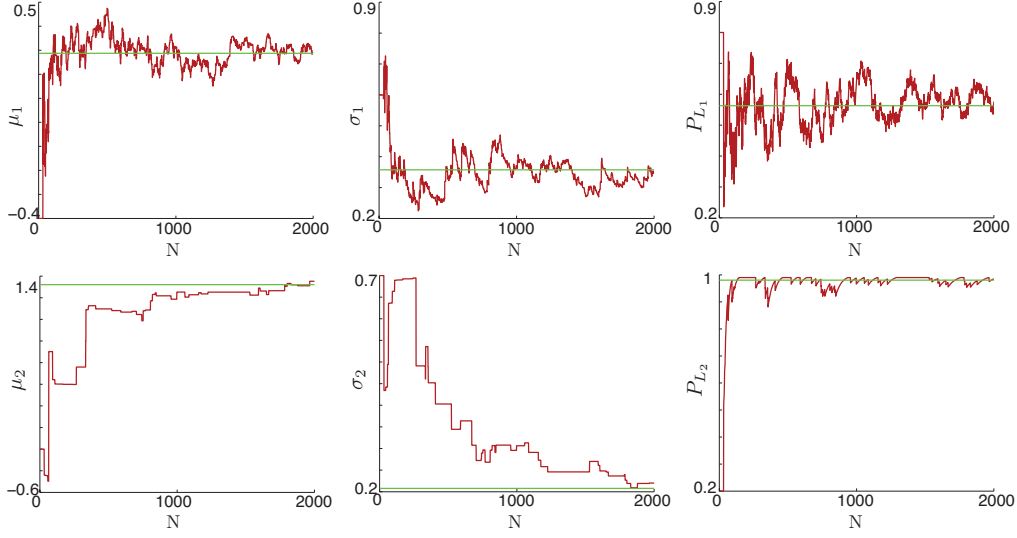
In order to assess the computational efficiency of our batch estimation algorithm, we perform a test that compares its performance to that of a standard numerical optimization algorithm. As above, we sample true parameters  $\theta_{uv,a}$  in the intervals as reported in Table 5.1, and draw a sample size of  $N = 1000$ . Figure 6.6 shows the comparison of our method with a Sequential Quadratic Programming (SQP) algorithm, evaluated over 100 simulations. Figure 6.6a shows that, for an equal number of maximum iterations, the batch estimation algorithm produces significantly lower Kolmogorov-Smirnov distance values. Moreover, Figure 6.6b shows that running the batch EM is significantly faster than running the SQP algorithm.

## 6.3 Mapping UWB Error Models

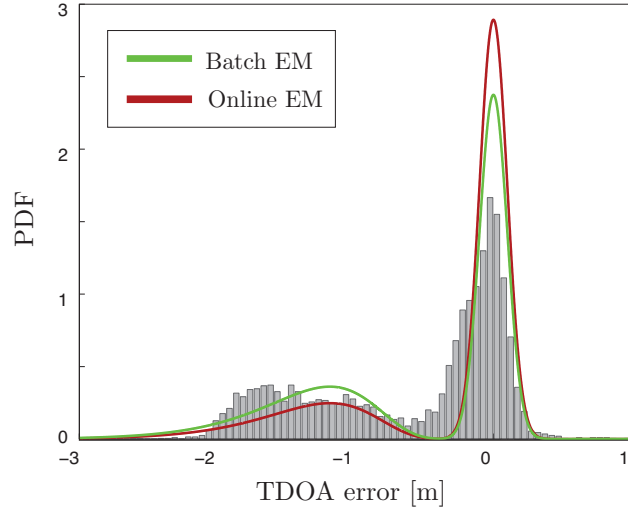
Now that we are able to calibrate UWB error models as a function of underlying data, let us consider the problem of creating the maps in  $\mathcal{M}$ , which include multiple, spatially anchored error models. As stated in our problem formulation (Section 5.1), our mapping approach consists of tessellating space into a fixed number of  $N_A$  areas. For simplicity, we will consider a fixed grid tessellation such as the one depicted in Figure 6.7. Each area is associated to an error model, which is initialized with parameters randomly sampled from the ranges reported in Table 5.1. The models can be calibrated using either our online method or our batch method.

### 6.3.1 Online Mapping

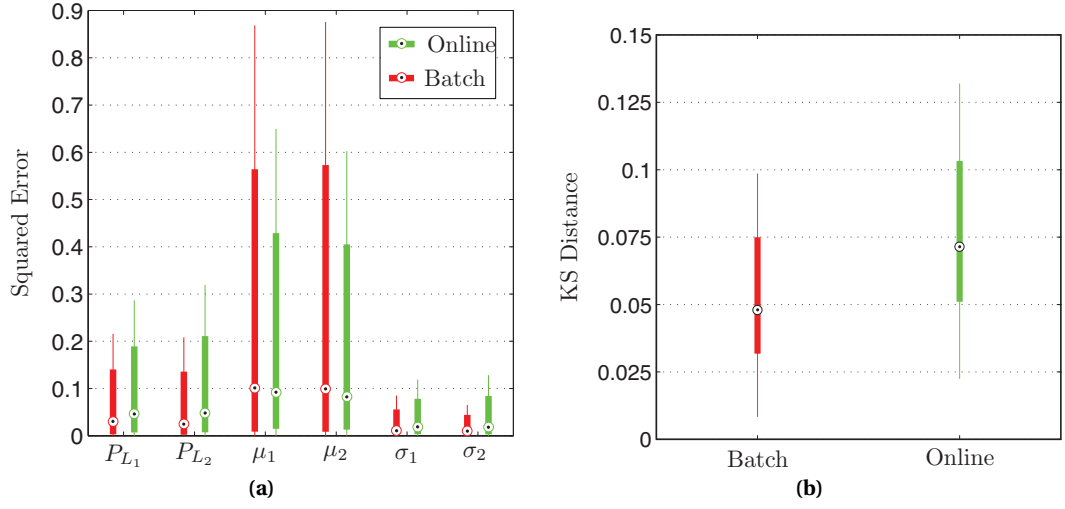
At each pass through an area, a robot samples data points which are then used to refine the associated error model in real-time. When a robot passes through an area, its measurements contribute to online updates of the model parameters. Every subsequent time the robot passes through the same area and takes measurements, the estimation algorithm picks up the



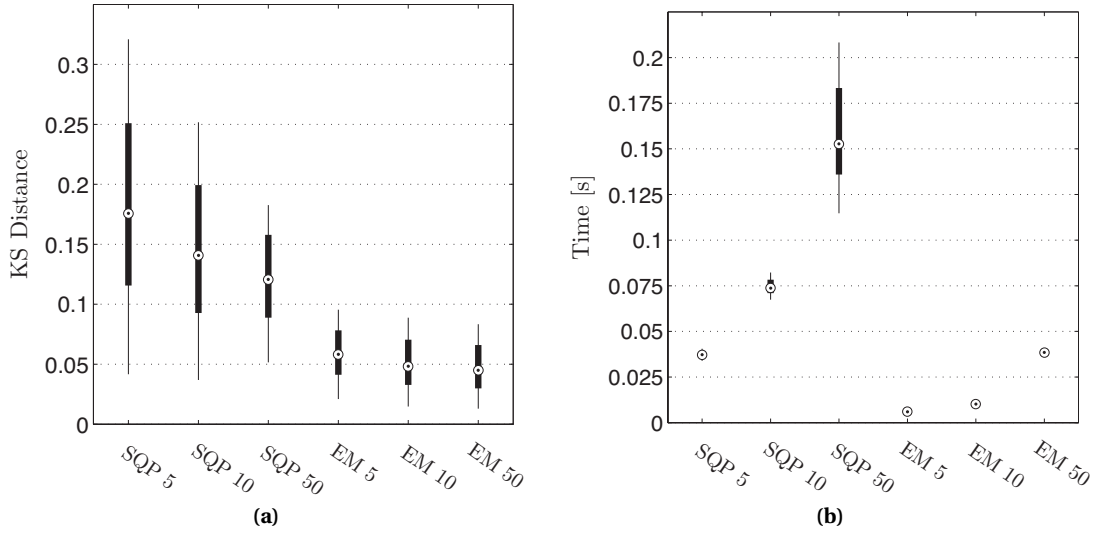
**Figure 6.3:** Online EM estimation of model parameters for a real base station pair  $\langle \mathcal{B}_2, \mathcal{B}_1 \rangle$ , for a number of  $N = 2000$  data points sampled in a  $1\text{m} \times 1\text{m}$  area  $A_1$  in our experimental setup, and  $\gamma = 1/k^{0.65}$ . The batch estimates (in green) are:  $\mu_1 = 0.28$ ,  $\mu_2 = 1.32$ ,  $\sigma_1 = 0.36$ ,  $\sigma_2 = 0.21$ ,  $P_{L_1} = 0.56$ , and  $P_{L_2} = 0.98$ . Parameters  $\tilde{\mu}$  and  $\tilde{\sigma}$  (not shown here) are defined as in Equation (5.14) to form the full model.



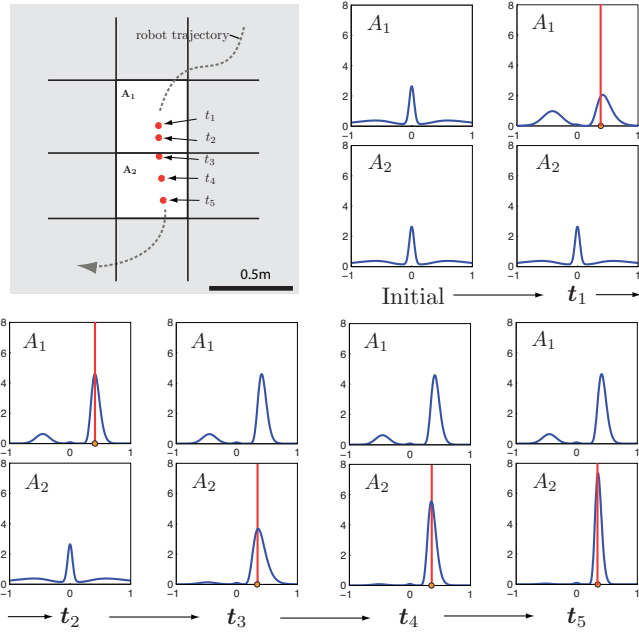
**Figure 6.4:** Normalized histogram of the data points used in Figure 6.3. We superimpose the probability density functions of the estimated models. For the online estimation, we average the 100 last estimates to provide our final estimate.



**Figure 6.5:** Comparison of batch and online EM estimation algorithms, evaluated over 1000 simulations. For the online algorithm, the final estimate of  $\hat{\theta}_{uv,a}$  is found by averaging the last 50 estimates. (a) Squared-error  $(\theta_i - \hat{\theta}_i)^2$ , with  $\hat{\theta}$  resulting from the batch and the online estimation algorithms. (b) Kolmogorov-Smirnov statistic for the online and batch estimation algorithms. The boxplots mark the median, and the 25th and 75th percentiles.



**Figure 6.6:** Comparison of a standard numerical optimization algorithm and our proposed batch EM estimation algorithm, evaluated over 100 simulations. The standard numerical optimization is performed using SQP based on the non-approximated model of Equation (5.11). The maximum number of iterations are marked by a number following the algorithm acronym (e.q., for a maximum of 5 iterations: EM 5 and SQP 5). (a) Kolmogorov-Smirnov distance of true and resulting estimated models. (b) Computational time for one optimization. The boxplots show the median, 25th and 75th percentiles.



**Figure 6.7:** Illustration of the online estimation algorithm. An excerpt of the robot path is shown, passing through areas  $A_1$  and  $A_2$ , which belong to an evenly tessellated map. Five measurements (vertical red lines) from base stations  $\langle \mathcal{B}_u, \mathcal{B}_v \rangle$  are taken during the time interval  $(t_1, t_5)$ , which are succeeded by online updates of the error models  $p(\Delta\tau_{uv}; m_{uv}(\mathbf{x}_{t_1})), \dots, p(\Delta\tau_{uv}; m_{uv}(\mathbf{x}_{t_5}))$  (blue curves).

model in the state when the robot last exited that area, and applies the new updates to this model. Figure 6.7 illustrates a simple example of how two models are updated. Let us describe this process formally. The mapping function in Equation (5.2) returns the parameter vector  $\theta_{uv,a}$  for a given base station pair  $\langle \mathcal{B}_u, \mathcal{B}_v \rangle$  and  $\mathbf{x}_n$ . In order to estimate the entries of  $\theta_{uv,a}$ ,  $K$  measurements  $\hat{\tau}_{uv,k}$  are gathered at times  $t_k, k = 1, \dots, K$ , with  $\mathbf{x}_{t_k} \in A_a$ . Thus, for every new data point  $\Delta\tau_{uv,k} \triangleq \Delta\tau_{uv}(\hat{\tau}_{uv,k}, \mathbf{x}_{t_k})$ , we update the feature  $\langle A_a, \theta_{uv,a} \rangle$  of map  $\mathcal{M}_{uv}$  according to our online estimation algorithm. This routine is illustrated in Algorithm 4. Line 6 refers to the online estimation algorithm, which we elaborated in detail in the previous paragraphs (and showed in Equation (6.15)).

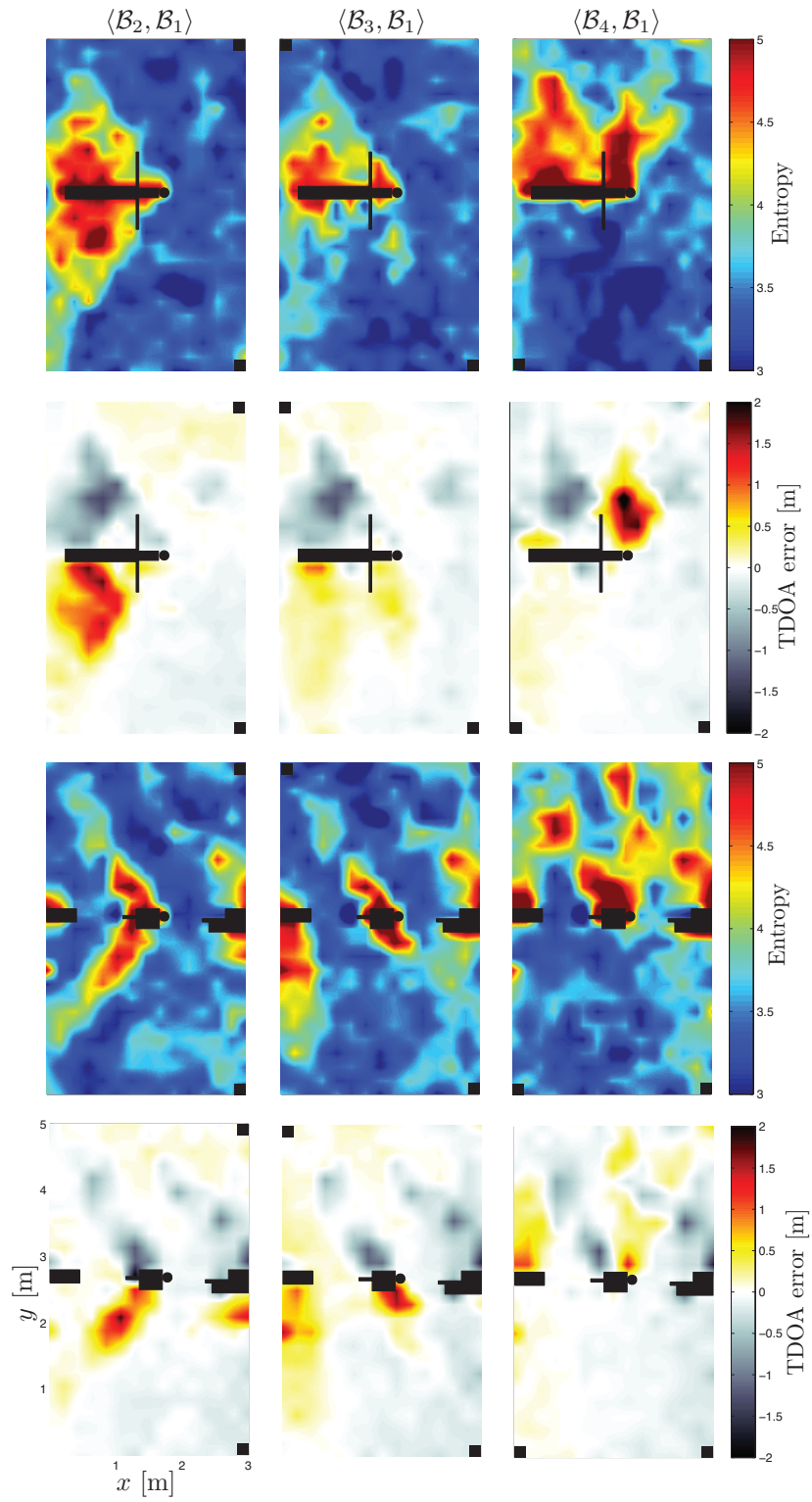
Annex B discusses the potential application of this online estimation algorithm within a simultaneous localization and mapping routine. Although some work still needs to be done for this integration to be fully efficient, some preliminary results indicate the possibility of creating the error maps simultaneously, thus removing the need of building error maps a priori.

---

**Algorithm 4** Update\_Map( $\mathbf{x}_{t_k}, T_t, \mathcal{M}_{t-1}$ )

---

- 1:  $\mathcal{M}_t = \mathcal{M}_{t-1}$
  - 2: **for all**  $\langle \hat{\tau}_{uv,k}, \langle \mathcal{B}_u, \mathcal{B}_v \rangle \rangle \in T_t$  **do**
  - 3:    $\mathcal{M}_t \leftarrow \mathcal{M}_t \setminus \mathcal{M}_{uv}$
  - 4:    $\langle A_a, \bar{\theta}_{uv,a} \rangle \leftarrow \text{Find}(\langle A_a, m_{uv}(\mathbf{x}_{t_k}) \rangle \in \mathcal{M}_{uv})$
  - 5:    $\mathcal{M}_{uv} \leftarrow \mathcal{M}_{uv} \setminus \langle A_a, \bar{\theta}_{uv,a} \rangle$
  - 6:    $\theta_{uv,a} \leftarrow \text{Online\_EM}(\bar{\theta}_{uv,a}, \Delta\tau_{uv,k})$
  - 7:    $\mathcal{M}_{uv} \leftarrow \mathcal{M}_{uv} \cup \langle A_a, \theta_{uv,a} \rangle$
  - 8:    $\mathcal{M}_t \leftarrow \mathcal{M}_t \cup \mathcal{M}_{uv}$
  - 9: **end for**
  - 10: **return**  $\mathcal{M}_t$
-



**Figure 6.8:** Evaluation of experimental data as a function of space. The obstacle placements as well as the base station placements are schematically marked by black shapes.

### 6.3.2 Offline Mapping

The offline mapping approach differs from the online mapping approach in that it estimates the error models a posteriori, using the batch EM algorithm. Let us discuss this approach on hand of a practical example. To this means, we will consider two variant obstacle configurations: configuration **A**, as depicted in Figure 4.5, and configuration **B**, which rearranges the obstacle modules to form three separate obstacles, with one obstacle in the center, and two obstacles aligned with the arena walls, creating two “doorways”.

Since our experimental setup uses three base station pairs ( $\langle \mathcal{B}_2, \mathcal{B}_1 \rangle$ ,  $\langle \mathcal{B}_3, \mathcal{B}_1 \rangle$ ,  $\langle \mathcal{B}_4, \mathcal{B}_1 \rangle$ ),  $\mathcal{M}$  is composed of three maps  $\mathcal{M}_{21}$ ,  $\mathcal{M}_{31}$ , and  $\mathcal{M}_{41}$ . We collect a data set comprising TDOA values  $\hat{\tau}_{uv,t}$  for each base station pair  $\langle \mathcal{B}_u, \mathcal{B}_v \rangle$  (over 58'000 values for configuration **A**, and over 69'000 values for configuration **B**, per base station pair), and record the associated ground truth positions  $\mathbf{x}_{n,t}$  guaranteeing full coverage of our experimental arena. For each TDOA measurement  $\hat{\tau}_{uv,n,t}$  we then calculate the ground truth TDOA value  $\tau_{uv}(\mathbf{x}_{n,t})$ , which, in turn, allows us to compute the TDOA error value  $\Delta\tau_{uv,n,t} = \Delta\tau_{uv}(\hat{\tau}_{uv,n,t}, \mathbf{x}_{n,t})$  (see Section 5.1). For each cell  $A_a$ ,  $a \in \{1, \dots, N_A\}$  defined by our grid map, and for each of the three base station pairs, we associate all data points  $\Delta\tau_{uv,n,t}$  to the cell area they were taken in, and define the set

$$\mathcal{T}_{uv,a} = \{\Delta\tau_{uv,n,t} \mid \forall \mathcal{R}_n, \text{ with } \mathbf{x}_{n,t} \in A_a\}. \quad (6.16)$$

For all cells  $A_a$ , we can now estimate the model parameters  $\hat{\theta}_{uv,a}$  that define a unique TDOA error model per cell, using the batch EM algorithm (Equation (6.14)).

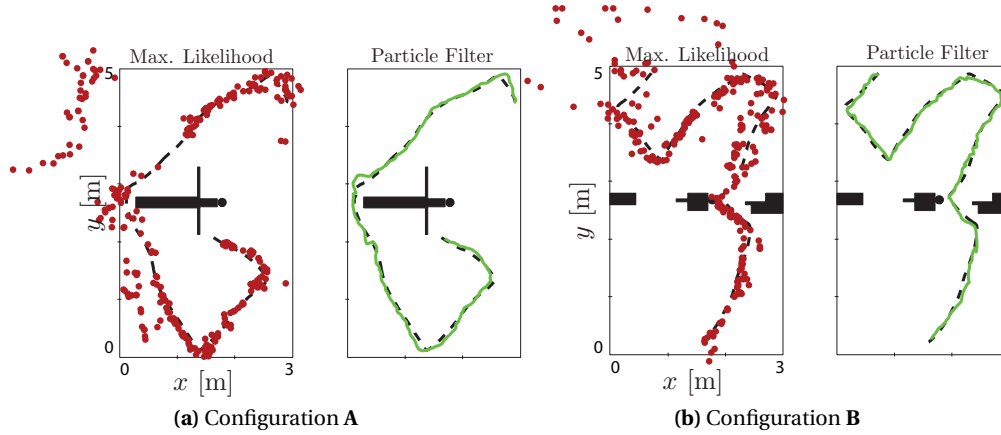
Figure 6.8 shows the collected data sets for configurations **A** (top half) and **B** (bottom half), in two variant visualizations as a function of space. The panels represent an overhead view of the experimental arena, for the three base station pairs (we perform 2D smoothing with a Gaussian kernel on a grid map of  $150 \times 250$  cells). The black shapes in the center correspond to the obstacle placements and the filled squares in the panel corners schematically indicate the placements of the base stations with respect to the layout of the experimental arena. For a given grid cell  $A_a$ , the two visualization variants show (i) the calculated average TDOA error values (rows 2 and 4),

$$\Delta\bar{\tau}_{uv,a} = \frac{1}{|\mathcal{T}_{uv,a}|} \cdot \sum_{\Delta\tau_{uv,n,t} \in \mathcal{T}_{uv,a}} \Delta\tau_{uv,n,t}, \quad (6.17)$$

and (ii) calculated entropy values (rows 1 and 3),

$$H_{uv,a} = \sum_{\Delta\tau_{uv,n,t} \in \mathcal{T}_{uv,a}} p(\Delta\tau_{uv,n,t}; \hat{\theta}_{uv,a}) \log \frac{1}{p(\Delta\tau_{uv,n,t}; \hat{\theta}_{uv,a})}. \quad (6.18)$$

In (i), we observe the higher bias averages of up to  $\pm 2$  m in the vicinity of the obstacle (note that TDOA errors are positive as well as negative). Also, we note that although this figure helps us identify strongly biased areas, the plotted average error values do not accurately represent the full multimodal error behavior. We now look at the entropy maps (ii), which provide an alternate interpretation of the data. Indeed, by computing the entropy values for each grid cell, we gain an insight on how areas are affected by the signal path: low entropy values indicate areas with high measurement predictability, physically related to LOS measurements, and likewise, high entropy values indicate areas with low measurement predictability, physically



**Figure 6.9:** The panels show a 160 s excerpt of the trajectory of one robot. Maximum likelihood position estimates (calculated with the raw TDOA values) are plotted in red, and the center of mass of the robot trajectory generated by our particle filter (using  $\mathcal{M}_{M,375}$ ) is plotted in green. The ground truth trajectory is shown by a dashed black line.

related to NLOS measurements. In this particular experimental scenario, we clearly see the qualitative correspondence with the average TDOA errors. This points to the fact that for both configurations **A** and **B** there are significant modes of the error distributions not centered around zero, and that the distributions in NLOS areas are either clearly multimodal, or have a very large spread (i.e., the distribution has a high entropy). In both of our scenarios, we also note the relation of the geometry of the setup to the NLOS patterns: higher absolute errors and higher entropy tend to lie in areas where direct LOS rays are occluded by the obstacles.

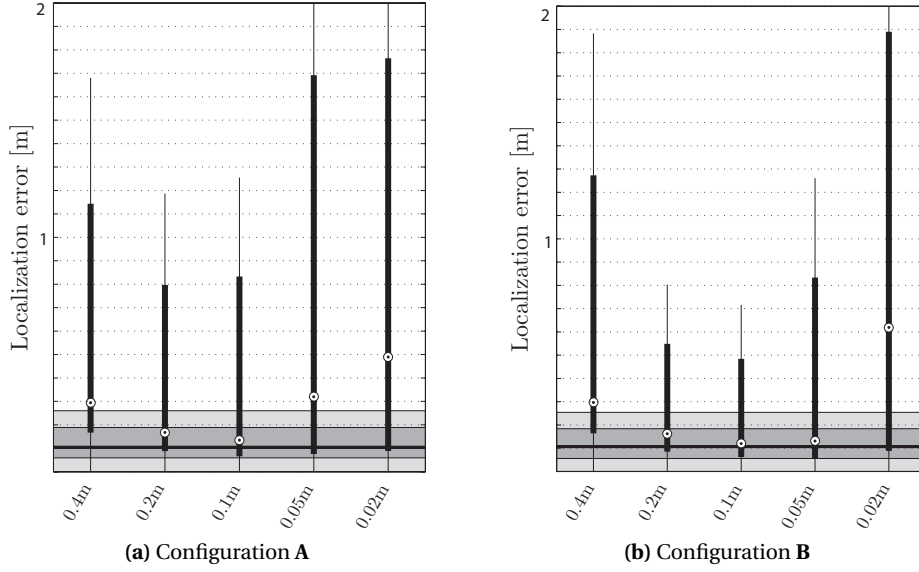
## 6.4 Performance

With all the elements in place, we now move on to evaluate our localization algorithm. In order to discuss the localization performance in terms of the absolute positioning error (distance to the ground truth position), we define a performance metric. For a given robot  $\mathcal{R}_n$  at time  $t$ , the positioning error is the distance from the center of mass of all its particles to the true position  $\mathbf{x}_{n,t}$ :

$$\mathcal{E} = \left\| \left( \frac{1}{M} \sum_i^M \mathbf{x}_{n,t}^{[i]} \right) - \mathbf{x}_{n,t} \right\|. \quad (6.19)$$

We note that other performance metrics (such as the root-mean-squared-error of all particles' positions with respect to the true position) could equally have been considered. Our chosen metric relates to the evaluation criteria of *accuracy*, as introduced in Section 1.1.3.

The algorithm is evaluated repeatedly (over 10 iterations) on a data set for each of the two configurations **A** and **B**, comprising a 40 minute experiment involving one robot, initially randomly placed in the arena. The robot runs Algorithm 3 with 50 particles. Apart from the modalities described so far (including UWB and odometry), the robot uses no other sensors to localize.



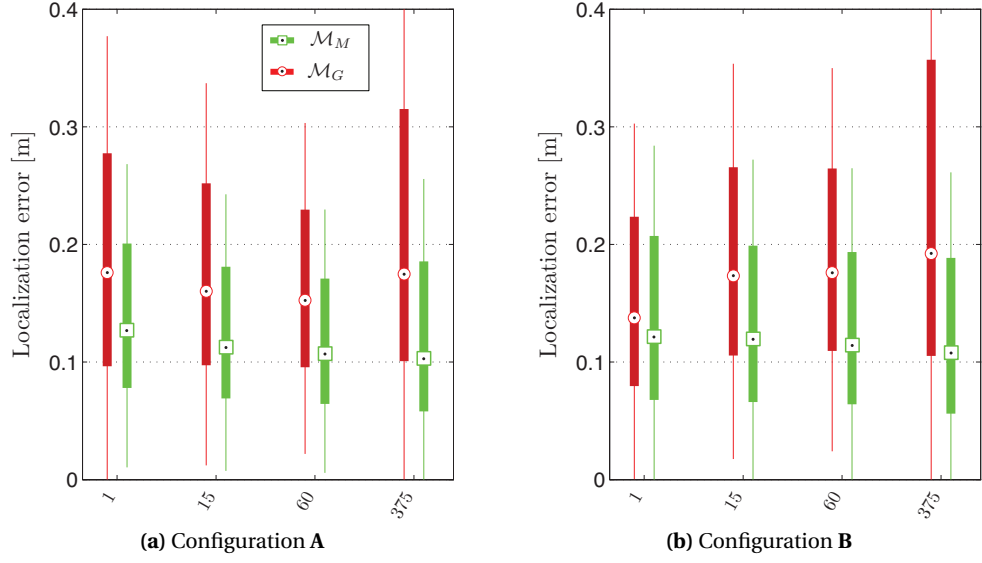
**Figure 6.10:** Localization performance using  $\mathcal{M}_{H,375}$ , with varying bin-sizes (5 boxplots along the x-axis), as well as  $\mathcal{M}_{M,375}$  (1 boxplot spanning the x-axis, shaded in grey). The boxplots show the median, 25th and 75th percentile of all localization errors throughout the experiment.

Figure 6.9 shows two excerpts of the robot trajectory for both experimental configurations, and superimposes (i) the maximum likelihood trilateration estimates (computed using only TDOA measurements), and (ii) the center of mass of the set of particles resulting from our localization algorithm. The particle filter uses a set of maps consisting of 375 square cells, 0.2 m in size. The images illustrate that, despite the occasionally very erroneous UWB measurements, the particle filter is able to maintain an accurate trajectory.

As a way of validating our UWB model, we calculate two additional, alternative sets of maps: (i) one composed of Gaussian distributions instead of our proposed error model (Equation (5.15)), and (ii), one composed of normalized, discrete histogram models with a fixed bin-size. We label our three map types according to the underlying *model*, i.e.,  $\mathcal{M}_M$  (multimodal),  $\mathcal{M}_G$  (Gaussian), and  $\mathcal{M}_H$  (histogram). Also, to test the effect of map granularity on the localization performance, we use four different grid maps: (a)  $N_A = 1$  with one  $3 \times 5$  m<sup>2</sup> cell, (b)  $N_A = 15$  with  $1 \times 1$  m<sup>2</sup> cells, (c)  $N_A = 60$  with  $0.5 \times 0.5$  m<sup>2</sup> cells, and (d)  $N_A = 375$  with  $0.2 \times 0.2$  m<sup>2</sup> cells. These two map parameters are combined, with resulting maps referred to as  $\mathcal{M}_{model, N_a}$  (e.g.,  $\mathcal{M}_{M,375}$  corresponds to a map with 375 multimodal models).

We assess the relevance of our multimodal model employed in the maps  $\mathcal{M}_{M,\cdot}$ , and compare its resulting localization performance with that of the Gaussian model employed in  $\mathcal{M}_{G,\cdot}$ , and with that of the histogram model employed in  $\mathcal{M}_{H,\cdot}$ . Figure 6.10 shows a comparison of the localization performance obtained for  $\mathcal{M}_{M,375}$  and  $\mathcal{M}_{H,375}$  with five different bin sizes (0.4 m, 0.2 m, 0.1 m, 0.05 m, 0.02 m). The plots show results for configuration A and B. We see that for both configurations, for  $\mathcal{M}_{H,375}$ , an optimal bin size can be found in the vicinity of 0.1 m. This median performance almost matches that of the multimodal model. However, importantly, we note that by decreasing the bin-size (i.e., increasing the granularity of the histogram model), a performance improvement is not necessarily achieved, as the histogram model begins to overfit the data and the overall localization error increases. Furthermore, we





**Figure 6.11:** Localization performance using  $\mathcal{M}_M$  and  $\mathcal{M}_G$  with four grid resolutions  $N_a$  (along the x-axis). The boxplots show the median, 25th and 75th percentile of all localization errors throughout the experiment.

note the significantly larger spread of localization errors produced by the histogram model, which is likely due to the discontinuity of the histogram model.

Figure 6.11 shows a comparison of the localization performance obtained for  $\mathcal{M}_M$  and  $\mathcal{M}_G$ , for all four grid resolutions,  $N_a = \{1, 15, 60, 375\}$ . We see that the multimodal model systematically produces better results, and for an increasing map resolution, tends to reduce the localization errors. In contrast, the Gaussian model does not systematically reduce localization errors for an increasing map granularity. This indicates that the Gaussian model produces a fundamental mismatch of the underlying data. In average and across all map resolutions, the multimodal model provides a 34% performance improvement over the Gaussian models for configuration **A**, and a 31% performance improvement for configuration **B**.

For a final comparison, we note that the error of the maximum likelihood trilateration estimates (evaluated over the whole 40 minute experimental run, calibrated with a standard deviation of 0.1 m) amounts to 0.18 m (median) and 0.48 m (mean) for configuration **A**, and 0.2 m (median) and 0.45 m (mean) for configuration **B**.

### Summary

In this chapter, we developed algorithms that estimate the parameters of our UWB models, as well as algorithms that apply these models in real-time localization routines. We introduced our general localization framework, the particle filter, and showed how we use it in our context by using a measurement model specific to UWB TDOA measurements. We went on to define a mapping strategy that creates spatially anchored UWB models that can be summarized as a map covering the experimental space. Importantly, this chapter showed how our UWB models can be calibrated very efficiently through an Expectation Maximization algorithm, given real data, in both an online mode as well as in a batch mode. We applied our mapping strategy to our experimental setup, and showed that the resulting values relate to the physical placements of obstacles. Our experimental results have shown that our algorithm (in combination with our multimodal model) outperforms alternative, standard methods, with a performance improvement of over 30%, and an overall localization performance with median errors in the range of 10-13 cm.

## 7 Conclusion

**I**N conclusion to our work on UWB localization in single-robot systems, we will review related work and place our results and achievements into a more global context. This chapter aims to distill a few main insights of our work and provide the reader with a high-level understanding of our contributions.

### 7.1 Related Work

Recently, UWB has received some attention within the robotics community. Hollinger et al. [50] use UWB range data from five fixed base stations to track a mobile robot, which carries an UWB emitter. Their underlying UWB error model is based on the Gaussian distribution, and is tested in three forms: an offset Gaussian, Gaussian Processes, and a mixture of Gaussians. The experimental tracking scenario tests ranging through walls, and does not rely on the fusion of additional odometry measurements for localization. The final results show room-level accuracy. The study performed by Gonzalez et al. [40] develops a probabilistic model for biased UWB range measurements which is based on an offset Gaussian. This offset (ranging bias) is modeled within an augmented state particle filter that does not take LOS/NLOS path conditions into account explicitly. Short experiments (of about 6 minutes duration) are performed using a significantly large particle set (15'000 particles), employing a mobile robot carrying an UWB transceiver and three external fixed UWB transceivers. LOS experiments yield an accuracy of 5 cm, and NLOS experiments yield an accuracy of 20 cm. A similar approach is taken by Jourdan et al. [57], where the ranging bias is also modeled in an augmented state particle filter. The bias is sampled from a fixed-width uniform distribution at regular intervals. Their experimental setup employs a mobile agent carrying an UWB emitter, one real receiver station, and one simulated receiver station, and uses the computed range data with odometry data. The authors demonstrate a performance of roughly 50 cm accuracy in a NLOS office environment. Lastly, Segura et al. [112] develop their own UWB positioning system based on TDOA measurements. Their system is composed of an UWB receiver board mounted on the robot, and three external UWB emitter beacons at fixed locations. The authors compute the robot position directly via constrained least squares minimization of the TDOA positioning equations, and without fusing robot odometry. The positioning accuracy is tested (statically) at five different locations in the experimental space, with errors in the order of 20 cm.

In contrast to these methods, we account for NLOS biases with a log-normal component in our probabilistic model, and we account for spatiality by including individual models in a map. Moreover, by defining our map granularity at will, we have a much higher control over the resulting accuracy. Also, we have shown that a set of 50 particles is sufficient to achieve reliable localization in a space of  $3 \times 5\text{m}$  in size. Importantly, since we quantify the UWB error as a function of space (with measures of the average TDOA error, as well as the entropy), we are able to assess our localization performance in relation to these values.

Our work considers a basic way of integrating odometry measurements with UWB measurements. However, a number of variant sensor fusion design strategies can be found, in particular in the domain of outdoor navigation [20, 125]. Our system relates to typical outdoor navigation systems, since we similarly use an absolute positioning component as well as a dead-reckoning component. In classical GPS literature, the fusion with Inertial Navigation System (INS) data is termed according to a naming convention that comprises four different integration options: loosely coupled, closely coupled, tightly coupled, and deeply coupled [42]. These four integration options quantify the depth of interaction of the sensor data and the amount of information that is shared between the respective sensory systems. In a loosely coupled system, the measurements from the GPS are processed separately in an independent filter before being fed to a main filter which outputs a final estimate. In contrast, a tightly coupled system considers a feedback loop from the main estimation filter to the GPS receiver. This feedback loop enables adaptive calibration and prediction of critical system variables (at the GPS level). At the most extreme level, a deeply integrated scheme is implemented directly on the hardware level (and thus, the composition actually constitutes one new sensor). The work in [104] considers the fusion of UWB and Micro Electro Mechanical Sensors (MEMS) in a closely coupled system. Our method compares best to that of a loosely coupled GPS/INS system. The benefits of performing tighter coupling still remain to be explored.

## 7.2 Discussion

Our results have shown that an effective solution can be found that leads to accurate localization with UWB, even in potentially cluttered indoor environments. In comparison with state-of-the-art works employing UWB on robots [40, 50, 57], in terms of localization accuracy, our method performs very well while bearing a small computational load. Additionally, we show how our method compares quantitatively with a baseline technique (maximum likelihood).

In terms of localization performance, our proposed multimodal model has shown clear benefits. Its general form avoids overfitting (as seen with the histogram model), yet it provides enough information to clearly distinguish itself from a basic Gaussian model. An instance of the multimodal model requires the determination of six parameters. The total memory needed to store a complete map is then obtained by multiplying this number by the number of base station pairs times the number of cells defined by a given tessellation. These parameters can be configured easily through our online estimation algorithm, which is incremental in its nature and can effectively be implemented on resource constrained platforms. It remains to be questioned whether other physically motivated models which have not yet been explored may result in better fits. Also, although we have empirically shown that for initial conditions sampled in predefined, physically feasible parameter ranges, our estimation algorithm always

converges, exceptions may happen and need to be taken into account. Finally, in this work we have considered two alternative models to which we compared our multimodal model: a histogram model and a Gaussian model. Other approaches (such as those belonging to the class of non-parametric models) could have been considered in place, and may have provided other insights. However, we have chosen those two models in particular because they are efficient and can be implemented easily on resource constrained platforms, thus representing viable alternatives. Indeed, we consider that the set of non-parametric models do not present viable alternatives (given the large number of data points and our computational restrictions), unless heavily constrained.

The main trade-off of our method is that it requires a one-time, a priori mapping step: in order to solve this, a mobile device with reliable localization capabilities (other than UWB) needs to profile the entire space of interest. A similar approach is suggested by Quigley et al. [101], where the indoor environment is profiled a priori to build probabilistic sensor models associated to space. Analogously, in particular in the domain of radio localization, a priori profiling is commonly known as *fingerprinting* [70]. The more UWB data is collected, the finer the resolution of the tessellation, and the better the resulting map. On the one hand, the return on investment provided by our method is very straight forward. On the other hand, it is to be assumed that low resolution maps may be equally good, given that the cell separations faithfully separate LOS (with mostly unimodal error distributions) from NLOS areas (with mostly multimodal error distributions), as well as separate differing NLOS cells from each other. Further, the validity of our model still needs to be investigated for highly dynamic environments. On the one hand, one could assume that moderately busy environments would only produce very sporadic anomalies, and that the overall localization quality would be maintained. On the other hand, in highly dynamic environments, it would be of great interest to investigate additional methods (such as collaboration in multi-agent systems, or fusion with additional exteroceptive sensors). Moreover, the development of adaptive calibration methods (based on our online estimation algorithm) may be able to contribute to a dynamic set of models that are faithful to the underlying error statistics over longer periods of time.

## 7.3 Conclusion

Although, in theory, UWB localization has the potential of providing centimeter level accuracy, in practice, sophisticated strategies are necessary to mitigate the effect of NLOS biases. We have taken a high-level approach based on the existence of time-of-flight measurement errors. This led us to develop measurement models as well as the localization algorithms that employ them in real-time. Empirical tests indicated that the error behavior of UWB measurements is dependent on the configuration of the environment, and thus, can be modeled as a function of space. Duly, we integrated our measurement models into a spatial framework to provide the possibility of configuring multiple models based on measurement localities. Our experimental results confirmed that our approach contributes to improved localization performance, and showed that accurate localization in cluttered environments is possible. The degree of accuracy is configurable at a user level, and a desired performance grade can be achieved methodically. Our resulting framework is efficient, and can be deployed on resource constrained embedded platforms.

### Summary

We conclude our work on UWB localization methods for single-robot systems by rounding up our contributions, and putting them into a global context. Our work stands out in particular by providing models and algorithms that aim to exploit the full benefits of UWB in difficult environments. To the best of our knowledge, we are among the first to present a framework that models UWB errors as a function of space. Moreover, we take care to develop efficient methods which can run on simple embedded platforms. Our discussion points out that the main trade-off of our method is its dependence on an a priori mapping (calibration) phase. However, our results support the insight that the final, overall localization performance relates directly to the effort put into this phase. A user is thus able to adjust his a priori involvement with respect to given performance requirements. A final item that still remains to be addressed is the impact of a highly dynamic environment.

# **Multi-Robot Localization Part III**





## 8 Introduction

*Doing better in a team.*

THE real world is full of scenarios where multiple technical devices co-exist, and possibly, co-work in the same environment. Indeed, the aim of collaboration is to add value to a system, at the individual as well as group level. In the following chapters, we will look at how to devise a collaboration strategy between multiple independent devices, that ultimately benefits the system as a whole. We will keep in mind that our final goal is to improve localization performance. In particular, we will ultimately investigate the value of collaborative strategies in the context of UWB localization. This being stated, this part of the manuscript will focus on the basic underlying mechanisms we employ for collaborative localization, leaving the combination of collaboration with UWB for the last part of the manuscript.

### 8.1 Motivation

Let us begin by assuming that systems composed of multiple entities may rely on strategies of distributed or collaborative problem solving. Such strategies will include some form of inter-entity interaction that is meant to augment the value of the system as a whole (e.g., by improving its outcome and/or performance). By acting on the inter-entity interactions, the system is transformed into a *team* (of entities). From a technological point of view, solving a problem with a team of computing devices is very attractive, especially when the task at hand is intrinsically already distributed (spatially or temporally), or simply too complex for a single unit to solve by itself. The research domain of multi-robot systems has tasked itself with the questions that revolve around such problems. Although there are many flavors of multi-robot systems—tele-operated, autonomous, networked, distributed and modular—the main incentive to collaborate is prevalent [113].

Bridging the gap to our topic of localization, we notice that a variety of tasks performed by multi-robot systems, such as search and rescue [56, 58], environmental monitoring [18, 86], and construction of real structures [69, 128], actually *need* accurate localization to succeed.

However, due to the intrinsic nature of such tasks, the individual agents are often confined to a small size and weight, which sets hard limits on on-board resources. Simultaneously, a large portion of the robot's resources may be dedicated to the task at hand, especially when this task requires high-frequency perception-to-action loops, leaving little room for solving the localization problem. These compounding problems pose the challenge of designing systems and algorithms that can flexibly accommodate given restrictions, without compromising performance.

Despite the large amount of work already done in this domain, to date, the solution to the localization problem for fully decentralized, large-scale multi-robot systems is still rather an open question. We contribute to this particular problem outline by presenting a concise solution that builds upon the ability to perform robot-to-robot observations. Various strategies have been followed in past works on collaborative localization—our work distinguishes itself by respecting the following design goals:

- **Low-cost:** The time/energy spent on the localization algorithm must be inferior to that spent on the actual task at hand. Thus, we try to minimize the overall complexity of our algorithm, and simultaneously relax the communication requirements.
- **Full decentralization:** Each robot carries responsibility for its own localization, and runs an independent localization algorithm on-board.
- **Any-time relative observations:** Robot-to-robot observations can be made asynchronously, at any given time. This simultaneously means that there are no connectivity constraints on the robot team, and that the computational time of fusing relative observations with proprioceptive sensing is bounded.
- **Mobility:** Since our system is decoupled and decentralized, we do not constrain mobility by making use of any methods that rely on motion agreements among the robots.
- **Independence of the environment:** In order for our method to be equally suited for indoor and outdoor applications, in structured as well as unstructured environments, it should be self-contained and robust. Thus, we rely only on inter-robot relative sensing, and on the possibility of an initial localization (of one of the robots).

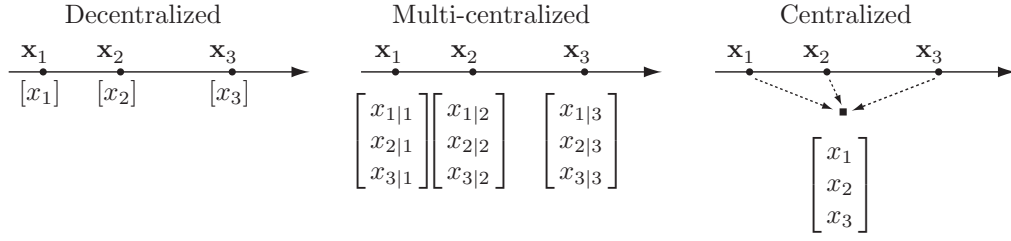
In the two chapters that follow this introduction (Chapters 9 and 10), we will elaborate the components of our method, and demonstrate the utility of our low-cost localization algorithm on groups of up to ten real mobile robots.

## 8.2 Centralized, Multi-Centralized or Decentralized?

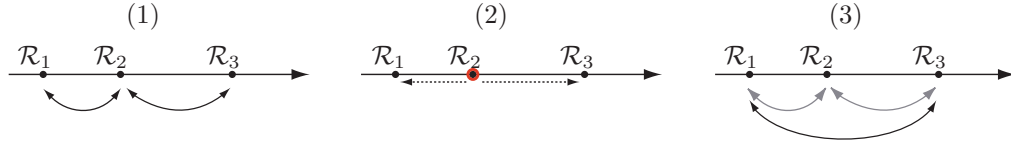
The first questions to be posed in designing a multi-robot localization framework are: Will the work load of estimating the state be distributed over the individual team members, and if yes, how? And what requirements does this choice pose on the information-sharing infrastructure?

Let us consider a simple example. We have a system of three robots,  $\mathcal{R}_n$ ,  $n = 1, 2, 3$ , and each robot has a state  $\mathbf{x}_n = [x_n]$  in one-dimensional space. At one extreme, a *centralized* estimation scheme would consider one state  $X = [x_1, x_2, x_3]^T$  to track all three robots at once.

## 8.2. Centralized, Multi-Centralized or Decentralized?



**Figure 8.1:** Three robots in one-dimensional space. The panel shows three variant estimation schemes for a multi-robot localization framework: decentralized, multi-centralized, and centralized. The schemes define the way in which a robot tracks its state, and also how it interacts with the other robots through information-sharing mechanisms.



**Figure 8.2:** Correlation of robots. (1) Robots  $\mathcal{R}_1$  and  $\mathcal{R}_3$  are correlated to robot  $\mathcal{R}_2$ , but without mutual correlations. (2) Robot  $\mathcal{R}_2$  makes an observation and exchanges this newly obtained information with robots  $\mathcal{R}_1$  and  $\mathcal{R}_3$ , allowing them to update their state information. (3) Robots  $\mathcal{R}_1$  and  $\mathcal{R}_3$  become correlated since they are both correlated to robot  $\mathcal{R}_2$ .

At the other extreme, a decentralized estimation scheme would consider three separate states  $\mathbf{x}_1 = [x_1]$ ,  $\mathbf{x}_2 = [x_2]$ , and  $\mathbf{x}_3 = [x_3]$ , where each robot tracks its own state. In a *multi-centralized* scheme, each robot tracks the state of the whole system, and thus we could formulate the state of each robot as  $X_n = [x_{1|n}, x_{2|n}, x_{3|n}]^T$ . Figure 8.1 illustrates this example.

As a matter of fact, only the **multi-centralized** and **centralized** solutions are exact: When a robot pair collaborates, it exchanges information on the state and state uncertainty. The newly exchanged information is then used and integrated into robots' state estimates. By doing this, the robots become correlated. This correlation can only be kept track of by a complete system state estimate, such as present with the centralized or multi-centralized schemes. Figure 8.2 shows an example of how correlation arises. Due to the necessity of 'total' information sharing, the centralized schemes require a fully connected communication infrastructure, which, for  $N_{\mathcal{R}}$  robots, is in the order of  $N_{\mathcal{R}}^2$ . Also, any interruptions or delays in the communication channels may lead to inconsistencies at the system level. The **decentralized** approach relaxes these requirements. By factoring the system into individual robot states, inter-robot correlations are ignored, and all robots are assumed independent. This approximation is convenient, as it allows each robot to perform state updates locally: in the absence of observations, this amounts to running an independent localization filter on each robot. In the presence of observations, the additional information is used to refine the local estimates. The disadvantage is, however, that an unlucky series of observations may lead to overconfident (and possibly false) state estimates.

Choosing any one of these three localization schemes has its own consequences. A centralized or multi-centralized scheme will provide an exact solution, provided the underlying communication infrastructure is infallible. Also, given that a full system estimate is being maintained (on-board each robot, or off-board, on a centralized machine), the computational and communication overhead is significant (a brief discussion of related work in Chapter 11.1 shows

how the computational complexity scales at least quadratically with the number of robots). Hence, the challenge is to find frameworks that are formally centralized, but that minimize issues due to computational and communications bottlenecks. The decentralized scheme provides a tolerant alternative to the centralized schemes. It imposes very few requirements. Moreover, given its local nature, the underlying run-time algorithm can be tuned to any team member's individual computational resources. Here, the challenge is to devise frameworks that minimize the impact of the approximative computations. In conclusion, the trade-offs between the various schemes are very clear, and the final choice remains to be defined based on actual (real) platform characteristics and desired performance.

**Centralized** multi-robot localization schemes compute full-system (global) state estimates. The computations take place on a single device (i.e., an additional off-board device, or one of the robots), that simultaneously serves as a sink for all communications originating from the robot system.

**Multi-Centralized** multi-robot localization schemes compute a full-system (global) state estimate on-board each of the individual robots. In order to perform estimation updates, each robot needs to receive relevant information updates in a timely manner. The system relies on peer-to-peer communications.

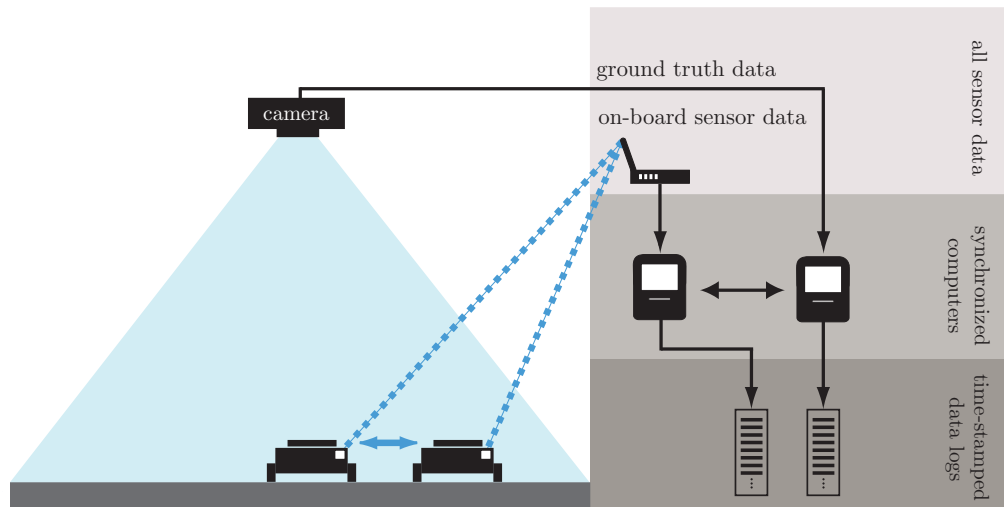
**Decentralized** multi-robot localization schemes compute a partial (local) system estimate on-board each of the individual robots. Peer-to-peer communications serve as a means to provide information updates among the individual robots.

### 8.3 Experimental Setup

The studies in this part focus on collaborative localization methods. In order to test and validate our proposed methods, we use an experimental setup that includes multiple, networked robots, that are endowed with relative observation sensors. We have already established a baseline experimental system in Section 2.2.2—this system is now easily extended to take into account more than one robot. Figure 8.3 illustrates our new setup.

The Khepera III robots used in our experiments are all equipped with a 802.11b WiFi card (that provides max. 11 Mbits/s), and communicate with each other over a wireless router that manages the local network. The messages are sent via User Datagram Protocol (UDP). Each robot possesses a unique ID, which is used to associate a sender/receiver pair to each message (none of our proposed methods rely on message broadcasts, as will become evident in the later chapters). Although we do not monitor the number of dropped packets in this setup, for some of our experiments, we remove information from the data logs a posteriori to vary the rate of received messages. Finally, we note that at start-up, we stagger the activation of each robot in a pre-defined order. As each pair of LED markers is detected, the overhead tracking system allocates the correct ID, and is thus able to monitor the group of uniquely identified robots.

In our work, we will consider various experimental configurations using this setup. We vary the robot team sizes (performing experiments involving 2, 4 as well as 10 robots), we vary the initial conditions (localized versus unlocalized), and we vary the duration of our experimental



**Figure 8.3:** The experimental infrastructure is composed of two elements: a vision-based ground truth positioning system, and several mobile robots. The robots are equipped with relative positioning modules (sensor data is collected by the robots themselves). The overhead camera monitors the uniquely identified robots, which drive around in a delimited area. The information from the subsystems is collected and time-stamped by synchronized computers to produce data logs.

runs (from approx. 4 min. to 40 min.). Notably, as well as evaluating our algorithms a posteriori on collected data, we will perform experimental runs on-board the robots, in real-time, to test the portability of our algorithms onto the actual embedded platform.

## 8.4 Infrared-based Relative Positioning

As introduced in Section 8.1, our collaborative method builds upon robot-to-robot observations. There are numerous ways in which this could be implemented in an experimental setup. To name a few, Fox et al. [35] use a camera for vision-based detection, Howard et al. [53] use a scanning laser range-finder and a pair of retro-reflective totem-poles, and Roumeliotis et al. [107] use the centralized information from an overhead tracking system. In all cases, the resulting sensor measurements are processed to produce a relative range and a relative bearing measurement. Possible variations include relative range only, relative bearing only, or combined relative range and bearing sensors—of which only the combined version is able to produce a full relative position estimate. Nevertheless, several authors have dealt with relative localization when either range [78] or bearing [38], but not both measurements, are available.

Recently, the usage of infrared-based range and bearing sensors has gained in popularity: for commercially available robots [99], for open robot architectures [43], as well as for customized robots [106]. Such infrared-based sensors are particularly popular for their cheap, off-the-shelf components, which can also be produced at a very small scale. Moreover, the signals have the potential of being fast, which allows for very fast update rates.



**Figure 8.4:** Fleet of ten Khepera III robots (here without the top-layer LED tracking module). The robots are equipped with an inter-robot, relative range and bearing module, which is composed of a ring of 16 infrared LEDs.

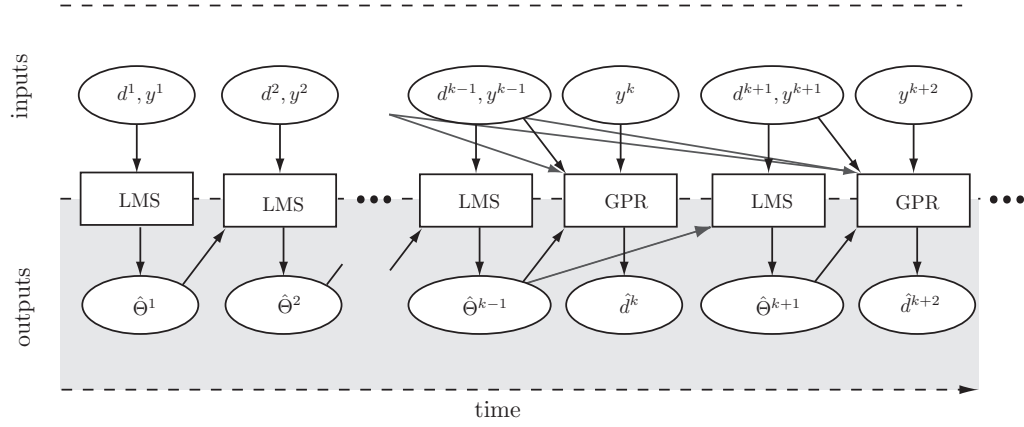
### 8.4.1 Hardware Platform

The relative positioning hardware used in our setup was previously developed in our research laboratory by Pugh et al. [99], and is based on infrared technology. The module consists of a circular board that has sixteen infrared Light Emitting Diodes (LEDs), evenly spaced on its perimeter, and eight infrared receivers capable of measuring the RSSI of the incoming infrared light. The infrared light emissions are modulated. The measured raw dimensionless RSSI is proportional to the actual magnitude of the emitted light. This platform is also able to broadcast low bit rate communication packets, thus enabling association of RSSI values with a specific emitter (robot). It is set to provide measurements at a maximum frequency of 2 Hz (but can operate robustly up to a frequency of 100 Hz in a system of two robots). Given our dense multi-robot setup, we set its maximum detection range to be 1.75 m (the actual maximum is reported to be in the range of 3.5 m). Figure 8.4 shows a group of ten Khepera III robots each equipped with a range and bearing module.

### 8.4.2 Calibration

Unfortunately, infrared-based relative positioning modules have several disadvantages: *(i)* infrared receivers have variable sensitivity, *(ii)* infrared emitters have variable emitting powers, and *(iii)* environmental conditions (i.e. background noise) have a high impact on their performance. Moreover, these issues can be exacerbated when using cheap off-the-shelf components, and, thus, one can often not fully rely on a priori physical models to calibrate such sensors. Nevertheless, appropriate software can be developed to remedy these issues. For these reasons, and because the calibration of infrared-based range and bearing modules is poorly represented in literature, we developed a dedicated method to address our needs.

Our method is composed of two phases. The first phase of our method consists in applying a standard online stochastic-gradient method, least-mean-squares (LMS). This step results in a fairly good estimation  $\hat{\Theta}$  of the parameters  $\Theta$  of our underlying physical model (which, for simplicity, we will not show here, but can be found in our published work [41]). The downside of using LMS is that it cannot easily adapt to changes and is limited to the given physical



**Figure 8.5:** Illustration of an online example of our two-phase calibration procedure. At a time step  $k$ , a receiver gathers an RSSI measurement  $y^k$ , and possibly also a ground truth range value  $d^k$ . Training and estimation are interleaved, depending on whether a ground truth value is available. LMS outputs an estimate of the model parameters,  $\hat{\Theta}^k$ . GPR uses previous input value pairs, as well as the previous model estimate, and outputs an estimate of the range value  $\hat{d}^k$  given the most recent RSSI measurement.

model. We overcome these limitations by adding a second phase, which consists in refining this first estimate by adapting the underlying physical model to the present observations. To do this, we use the parameters  $\hat{\Theta}$  computed by the LMS method, and bootstrap a Gaussian Process Regression (GPR). On the one hand, the advantage of GPR is that it can account for characteristics present in the data that are not visible in the given explicit model. Thus, it is able to find the nonlinearities present at the level of the receiver sensor. On the other hand, finding optimal values for  $\Theta$  using GPR alone is cumbersome, as it requires a numerical optimization procedure that needs to invert a covariance matrix the size of the number of data points collected. This motivates our combined two-phase method. We note that previous work by Fink et al. [33] has demonstrated the feasibility of using a pure GPR approach, however, at the cost of running a relatively powerful and well equipped mobile platform. Finally, we note that our general calibration method is extended to also account for the incident angle, thus enabling the determination of the bearing value (with a detailed elaboration in [41]).

This two-phase combination addresses the above-mentioned issues, and also implicitly adapts the a priori underlying physical model. By utilizing the results of the first calibration phase and reducing the search space of the second, computationally more complex phase, the methodology is purposefully designed to run on resource-constrained robots.

Our method can be used in an online as well as an offline fashion. Figure 8.5 shows a schematized example of the two-phase method used in an online fashion. Practically, however, we perform an offline calibration. We place our robots into our experimental arena and let them move randomly for ten minutes. Training points are gathered using the range and bearing module in conjunction with the overhead camera tracking system. The final model estimates are then hard-coded into the robot software providing it with the means to compute range and bearing values on-board. Finally, our published work [41] shows how our calibration method is able to improve over a standard, manual calibration by over 20% for bearing measurements, and over 30% for range measurements.

### Summary

In this chapter, we introduced the concept of using a multi-robot team for collaborative localization. Our chosen approach distinguishes itself from prior work by emphasizing low-cost, full decentralization, asynchrony, unrestricted mobility, and independence of the environment. Further, we discussed the trade-offs of decentralized versus centralized estimation architectures, and pointed out the strengths and weaknesses of each. We conclude our chapter by characterizing our multi-robot experimental setup. The robots possess a custom relative range and bearing hardware module. A brief paragraph detailed how we calibrated the board for use in our experiments.



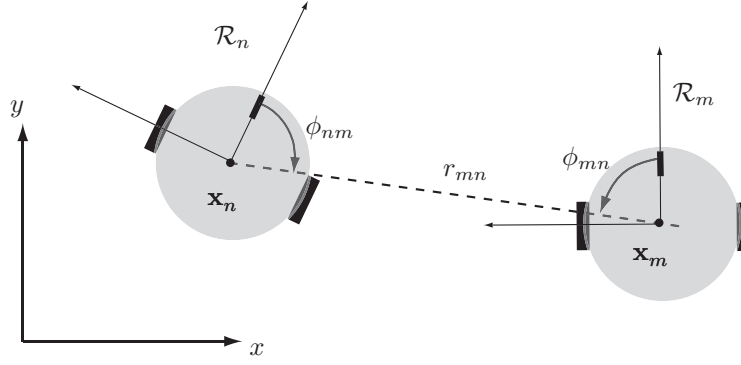
## 9 Sensor Model

**T**HE previous chapter established relative range and bearing values as the main quantities defining the interactions between individual robots in our multi-robot system. Here, we will develop a probabilistic robot detection model that integrates these values to output an updated belief of the detected robot's position. Our method will build upon the general localization framework introduced in Chapter 6: the particle filter.

### 9.1 Problem Formulation

Our collaboration strategy exploits associated, inter-robot relative range and bearing observations, which are evaluated by a dedicated detection model to form position estimates. Let us consider a multi-robot system of  $N_{\mathcal{R}}$  robots (the number  $N_{\mathcal{R}}$  does not necessarily need to be known by the robots), deployed with noisy relative range and bearing sensors. Figure 9.1 illustrates such a system, indicating the nominal (error-free) range quantity  $r_{mn}$ , and the nominal bearing quantity  $\phi_{mn}$ . We use the notion of neighborhoods: at time  $t$ , a robot  $\mathcal{R}_m$  is in the set of neighbors  $\mathcal{N}_{n,t}$  of robot  $\mathcal{R}_n$  if robot  $\mathcal{R}_m$  is able to take a range measurement  $\hat{r}_{mn,t}$  and bearing measurement  $\hat{\phi}_{mn,t}$  of robot  $\mathcal{R}_n$ . Thus, at every moment in time, the neighborhood topology is defined by the physical constraints given by the relative observation sensors deployed on the robots. Also, if  $\mathcal{R}_m \in \mathcal{N}_{n,t}$ , we make the assumption that the robot  $\mathcal{R}_m$  can communicate with the robot  $\mathcal{R}_n$  to send detection data. The range and bearing measurement errors for robots  $\mathcal{R}_n$  and  $\mathcal{R}_m$  are defined as the difference between the nominal range and bearing values at the actual robot positions  $\mathbf{x}_{n,t}$  and  $\mathbf{x}_{m,t}$  and the measured values  $\hat{r}_{mn,t}$  and  $\hat{\phi}_{mn,t}$

$$\begin{aligned}\Delta r(\hat{r}_{mn,t}, \mathbf{x}_{m,t}, \mathbf{x}_{n,t}) &= \hat{r}_{mn,t} - r(\mathbf{x}_{m,t}, \mathbf{x}_{n,t}) \\ \Delta \phi(\hat{\phi}_{mn,t}, \mathbf{x}_{m,t}, \mathbf{x}_{n,t}) &= \hat{\phi}_{mn,t} - \phi(\mathbf{x}_{m,t}, \mathbf{x}_{n,t}),\end{aligned}\tag{9.1}$$



**Figure 9.1:** System of  $N_{\mathcal{R}} = 2$  robots  $\mathcal{R}_n$  and  $\mathcal{R}_m$ , at positions  $\mathbf{x}_n$  and  $\mathbf{x}_m$ , respectively. The figure depicts the relative robot range  $r_{mn} = r_{nm}$ , and the relative bearing values  $\phi_{mn}$  and  $\phi_{nm}$ .

where range and bearing are described as functions of positions  $\mathbf{x}_{m,t}$  and  $\mathbf{x}_{n,t}$ , representing a transformation from Euclidean to polar coordinates:

$$\begin{bmatrix} r_{mn,t} \\ \phi_{mn,t} \end{bmatrix} \triangleq \mathbf{T}_e^p(\mathbf{x}_m, \mathbf{x}_n) = \begin{bmatrix} r(\mathbf{x}_{m,t}, \mathbf{x}_{n,t}) \\ \phi(\mathbf{x}_{m,t}, \mathbf{x}_{n,t}) \end{bmatrix} = \begin{bmatrix} \sqrt{(x_n - x_m)^2 + (y_n - y_m)^2} \\ \text{atan2}((y_n - y_m), (x_n - x_m)) - \psi_m \end{bmatrix}. \quad (9.2)$$

In the context of the particle filter, we evaluate these difference values for a given particle  $\mathbf{x}_m^{[j]}$  belonging to the belief of the detecting robot  $\mathcal{R}_m$ . We define

$$\Delta r_{mn,t}^{[j]} \triangleq \Delta r(\hat{r}_{mn,t}, \mathbf{x}_m^{[j]}, \mathbf{x}_{n,t}) \quad (9.3)$$

$$\Delta \phi_{mn,t}^{[j]} \triangleq \Delta \phi(\hat{\phi}_{mn,t}, \mathbf{x}_m^{[j]}, \mathbf{x}_{n,t}). \quad (9.4)$$

Thus, we define a probability density  $q^{[j]}$  that depends on a parameter set  $\xi$ , and which describes the likelihood of position  $\mathbf{x}_{n,t}$  for a single particle  $\mathbf{x}_m^{[j]}$  in the belief of robot  $\mathcal{R}_m$ , given measurements  $\hat{r}_{mn,t}$ ,  $\hat{\phi}_{mn,t}$  as

$$q^{[j]}(\Delta r_{mn,t}^{[j]}, \Delta \phi_{mn,t}^{[j]}; \xi). \quad (9.5)$$

Finally, by considering all particles belonging to robot  $\mathcal{R}_m$ , we define the robot detection model  $q$  which describes the probability that robot  $\mathcal{R}_m$  detects robot  $\mathcal{R}_n$  at position  $\mathbf{x}_{n,t}$  as

$$q(\Delta r_{mn,t}, \Delta \phi_{mn,t}; \xi) = \eta \cdot \sum_{\langle \mathbf{x}_m^{[j]}, w_{m,t}^{[j]} \rangle \in X_{m,t}} q^{[j]}(\Delta r_{mn,t}^{[j]}, \Delta \phi_{mn,t}^{[j]}; \xi) \cdot w_{m,t}^{[j]} \quad (9.6)$$

where  $\eta$  is a normalization factor, and  $\Delta r_{mn,t} = \{\Delta r_{mn,t}^{[j]} | j = 1, \dots, M\}$  is the sets of all the range difference values, and  $\Delta \phi_{mn,t} = \{\Delta \phi_{mn,t}^{[j]} | j = 1, \dots, M\}$  is the set of all the bearing difference values, for all particles of robot  $\mathcal{R}_m$ .

The scenario described above is the default case, and we shall generally refer to Equation (9.6) as the detection model when discussing our experimental work. For the sake of completeness, however, the following formulas will detail the case where robot  $\mathcal{R}_n$  reciprocally detects robot  $\mathcal{R}_m$  (in addition to  $\mathcal{R}_m$  detecting  $\mathcal{R}_n$ ). When this occurs, robot  $\mathcal{R}_n$  can use the information of its own relative observation to determine the orientation difference  $\Delta \psi_{mn,t}^{[j]}$  with the  $j$ th

particle of robot  $\mathcal{R}_m$ , which is defined by the following geometric relation

$$\Delta\psi_{mn,t}^{[j]} = \pi + \psi_{m,t}^{[j]} - \psi_{n,t} + \hat{\phi}_{mn,t} - \hat{\phi}_{nm,t}. \quad (9.7)$$

Analogous to the development above, we can then define

$$q^{[j]}(\Delta r_{mn,t}^{[j]}, \Delta\phi_{mn,t}^{[j]}, \Delta\psi_{mn,t}^{[j]}; \tilde{\xi}) \quad (9.8)$$

that includes the orientation difference, and

$$q(\Delta r_{mn,t}, \Delta\phi_{mn,t}, \Delta\psi_{mn,t}; \tilde{\xi}) = \eta \cdot \sum_{\langle \mathbf{x}_{m,t}^{[j]}, w_{m,t}^{[j]} \rangle \in X_{m,t}} q^{[j]}(\Delta r_{mn,t}^{[j]}, \Delta\phi_{mn,t}^{[j]}, \Delta\psi_{mn,t}^{[j]}; \tilde{\xi}) \cdot w_{m,t}^{[j]} \quad (9.9)$$

that denotes the extended probability density function. We point out that the set of parameters  $\tilde{\xi}$  are not identical to the ones in Equation (9.6), denoted by  $\xi$ . The following chapter will discuss the actual implementation of this model in more depth.

## 9.2 Range & Bearing Model

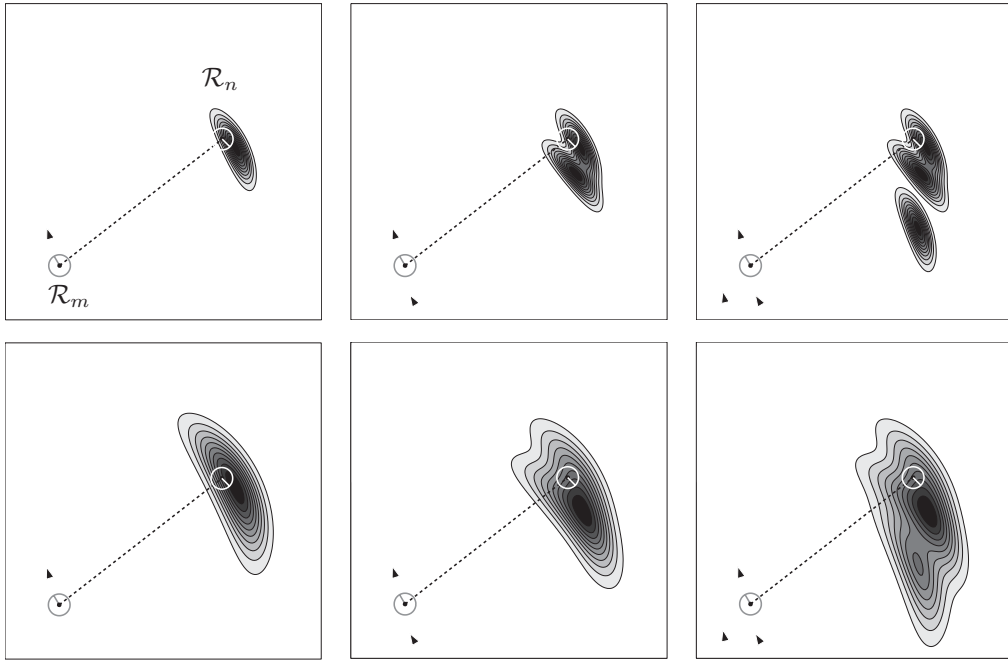
The detection model  $q(\Delta r_{mn,t}, \Delta\phi_{mn,t}; \xi)$  of Equation (9.6) describes the probability that robot  $\mathcal{R}_m$  detects robot  $\mathcal{R}_n$  at position  $\mathbf{x}_{n,t}$ . For such a collaboration to take place, robot  $\mathcal{R}_m$  needs to communicate its range and bearing measurements  $\hat{r}_{mn,t}$ ,  $\hat{\phi}_{mn,t}$  as well as its set of particles  $X_{m,t}$  to robot  $\mathcal{R}_n$  (confer Equations (9.1)—(9.5)). Thus, a communication message composed as  $d_{mn,t} = \langle \hat{r}_{mn,t}, \hat{\phi}_{mn,t}, X_{m,t} \rangle$  is sent from robot  $\mathcal{R}_m$  to robot  $\mathcal{R}_n$ . If several robots in a neighborhood  $\mathcal{N}_{n,t}$  communicate with robot  $\mathcal{R}_n$ , the received information is the set of all relative observations made by those robots at time  $t$ , as well as the belief representations  $X_{m,t}$  of all detecting robots  $\mathcal{R}_m \in \mathcal{N}_{n,t}$ . We denote this data set as  $D_{n,t} = \{d_{mn,t} | \mathcal{R}_m \in \mathcal{N}_{n,t}\}$ . The collaborative aspect of this formalism thus lies in the integration of robot  $\mathcal{R}_m$ 's belief into that of robot  $\mathcal{R}_n$ .

The probability density function  $q$  is applied to the ensemble of particles in the belief of robot  $\mathcal{R}_n$ , in order to adjust their weights to current relative observations. Given the nature of relative observations, we use for each particle  $j$  in  $X_{m,t}$  a Gaussian probability density function in polar coordinates centered at  $\mathbf{x}_{m,t}^{[j]}$ :

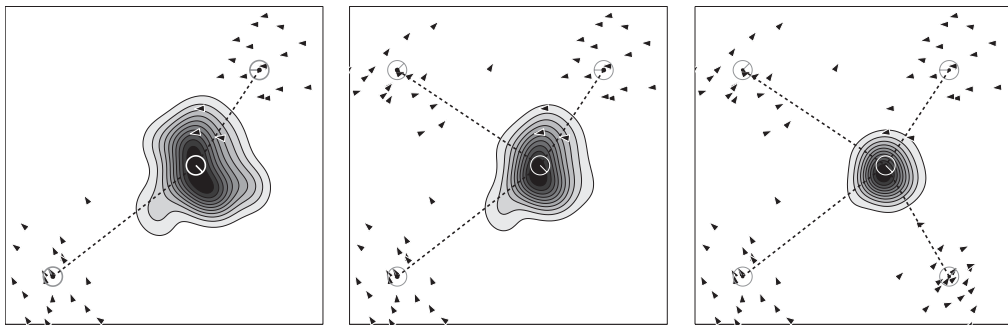
$$q^{[j]}(\Delta r_{mn,t}^{[j]}, \Delta\phi_{mn,t}^{[j]}; \xi) = \Phi \left( \mathbf{T}_e^p(\mathbf{x}_{m,t}^{[j]}, \mathbf{x}_{n,t}); \begin{bmatrix} \hat{r}_{mn} \\ \hat{\phi}_{mn} \end{bmatrix}, \xi \right) \quad (9.10)$$

where  $\Phi(\cdot; \boldsymbol{\mu}, \xi)$  is the Gaussian probability density function with mean  $\boldsymbol{\mu}$  and covariance matrix  $\xi$ . In practice, we use a homogenous set of calibrated hardware sensors, and thus, we use one value for the range standard deviation  $\sigma_r$ , and one value for the bearing standard deviation  $\sigma_\phi$ . The covariance  $\xi$  is

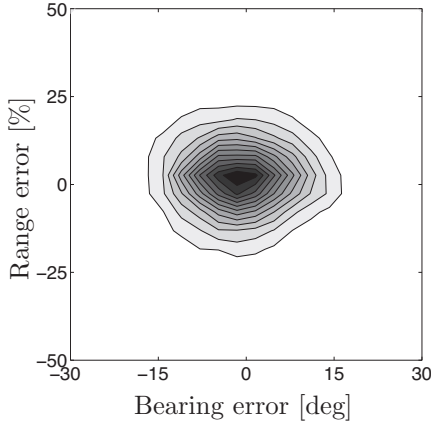
$$\xi = \begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\phi^2 \end{bmatrix} \quad (9.11)$$



**Figure 9.2:** Example of the range and bearing model. Robot  $\mathcal{R}_n$  (in white) applies the model on data received from robot  $\mathcal{R}_m$ . We consider three particles sets, composed of 1, 2, or 3 particles (from left to right), represented by oriented triangles in the vicinity of the detecting robot. The model's probability density is superimposed on the detected robot. The dotted line and the orientation of the robots show the actual relative range and bearing. The top row employs noise values  $\sigma_r = 0.03$  and  $\sigma_\phi = 0.1$ , and the bottom row multiplies these values by a factor of 2,  $\sigma_r = 0.06$  and  $\sigma_\phi = 0.2$ .



**Figure 9.3:** Example of an application of the detection model for 2, 3 and 4 robots. Here, we use a fixed set of 20 particles, represented by oriented triangles superimposed over the detecting robots. The particle positions were generated randomly from a normal distribution ( $\sigma_{xy} = 0.2$  m, and  $\sigma_\phi = 0.2$  rad). The model employs noise values  $\sigma_r = 0.15$  and  $\sigma_\phi = 0.15$ .



**Figure 9.4:** We collected 30'000 range and bearing values using four robots over 25 minutes, and calculated the measurement errors given the ground truth. The plot shows the density of the error values as a function the range error and the bearing error. The non-skewed shape of the 2-dimensional density indicates the independence of range and bearing values. Furthermore, we use this data to calibrate our detection model parameters: We have  $\sigma_r = 0.15 \cdot \hat{r}_{mn}$  and  $\sigma_\phi = 0.15$  rad.

where we assume the independence of range and bearing measurements. This assumption is supported by empirical work: Figure 9.4 provides experimental evidence for our platform showing that a range measurement and a bearing measurement behave like two independent Gaussian variables. For our extended model of Equation (9.9), we adapt the covariance to

$$\tilde{\xi} = \begin{bmatrix} \sigma_r^2 & 0 & 0 \\ 0 & \sigma_\phi^2 & -\sigma_\phi^2 \\ 0 & -\sigma_\phi^2 & 2\sigma_\phi^2 \end{bmatrix}. \quad (9.12)$$

We note that, for the purpose of our case-study, we use a simple Gaussian in polar coordinates, but all reasonings are valid for completely arbitrary distributions, which are equally well accommodated by our presented formalism. Indeed, since we use a particle filter, we can keep the same framework for any possible underlying range and bearing hardware not fulfilling the Gaussian noise assumption. Figure 9.2 illustrates how the detection model is constructed for one, two, and three particles in the detecting robot's belief, for two sets of noise values. This example demonstrates the multi-modal shape of the detection model (here, we observe up to three modes, one for each particle in the detecting robot's belief). Also, we see that the larger the spread of the particles, the larger the spread of the density function.

For a set of relative observations  $D_{n,t}$  taken at time  $t$ , the likelihood of a given pose sample  $\mathbf{x}_{n,t}^{[i]}$  is

$$Q(\mathbf{x}_{n,t}^{[i]}) = \eta \cdot \prod_{d_{mn} \in D_{n,t}} q(\Delta r_{mn,t}, \Delta \phi_{mn,t}; \xi) \quad (9.13)$$

where  $\eta$  is a normalization factor. The detection model incorporating the detection data from multiple detecting robots can be formulated as the update equation shown in Algorithm 5.

---

**Algorithm 5** `Detection_Model`( $D_{n,t}, \mathbf{x}_t^{[i]}, \bar{w}_t^{[i]}$ )

---

- 1:  $w_{n,t}^{[i]} \leftarrow \bar{w}_{n,t}^{[i]} \cdot Q(\mathbf{x}_{n,t}^{[i]})$
  - 2: **return**  $w_{n,t}^{[i]}$
-

Figure 9.3 illustrates the probability density function resulting from the detection model, for two, three, and four detecting robots. We see that when detection data from multiple robots is integrated into the range and bearing model, the detection precision increases. We note that for  $M$  particles, Algorithm 5 has a complexity of  $O(|\mathcal{N}_n|M)$  by itself, and when encapsulated in our global algorithm (which is shown in the following chapter), leads to a complexity of  $O(|\mathcal{N}_n|M^2)$ . This can be prohibitively costly for a large number of particles. Also, communication constraints may make sending large particle sets infeasible. Our next chapter will take a look at these issues and will elaborate a clustering strategy to mitigate these effects.

### Summary

In this chapter we introduced a robot detection model based on range and bearing measurements, and detailed how it is used within the particle filter framework that we defined in our work. The model's essential characteristic is that a *detected* robot applies the model using data received from the robots that made the detection. Also, it assumes underlying Gaussian noise (in polar coordinates) for the range and bearing measurements. While we only consider this particular model implementation, the localization framework (particle filter) allows for alternative (non-Gaussian) implementations, and thus maintains the generalizable nature of our approach.

# 10 Algorithms

**W**E contribute to the problem of multi-robot localization by proposing a low-cost method: in this chapter, we describe a fully decentralized algorithm, particularly designed for resource-limited robotic platforms in large-scale systems. The key of our approach is that we consider two novel algorithmic components which are used in conjunction with a baseline collaborative particle filter. In the following sections, we will elaborate these components and demonstrate the utility of our low-cost localization algorithm on groups of up to ten real mobile robots.

## 10.1 Multi-Robot Localization Framework

In this section, we elaborate our collaborative localization algorithm, which, together with the Monte-Carlo Localization (MCL) method presented by Fox et al. [35], forms the baseline for our work. The following sections will detail the methods we developed on top of this baseline, however, for convenience, the complete localization algorithm is already shown here, in Algorithm 6.

Our collaboration strategy exploits associated, inter-robot relative range and bearing observations. In order to accommodate the noise characteristics of typical relative range and bearing measurements, we use the robot detection model described in the previous chapter (Chapter 9), and introduce it into our localization algorithm. This combination forms the basis of our collaborative paradigm. Given this foundation, a key element and contribution of our approach consists of an additional routine, namely a *reciprocal* particle sampling routine, mainly designed to accelerate the convergence of a robot's position estimate (to the correct value), and to mitigate overconfidence.

A collaborative localization algorithm composed of the aforementioned robot detection model jointly with the reciprocal sampling routine is very efficient with respect to its non-collaborative counterpart. However, due to the computational overhead induced by the detection model and the reciprocal sampling routine (which scale to the square of the number of particles), such an algorithm may run into real-time running constraints. This can turn out to be particularly prohibitive for platforms with hard limits on available resources. For this reason, we further extend our approach with a *particle clustering* method that reduces the complexity of the

overall localization algorithm and also reduces the amount of data to be communicated. This clustering routine is especially designed to accommodate the characteristics of the range and bearing robot detection model, and does not impose an additional computational burden on the localization algorithm as a whole.

Let us from here on consider that a robot  $\mathcal{R}_n$  is detected by robot  $\mathcal{R}_m$ , and simultaneously receives localization information from robot  $\mathcal{R}_m$ . If we make the assumption that individual robot positions are independent (we refer the reader to Section 8.2 for our reasonings), we can formulate the update of the belief of robot  $\mathcal{R}_n$  at time  $t$  with

$$\text{Bel}(\mathbf{x}_{n,t}) = p(\mathbf{x}_{n,t}|u_{n,0..t}) \cdot \int p(\mathbf{x}_{n,t}|\mathbf{x}_{m,t}, r_{mn,t}, \theta_{mn,t}) \text{Bel}(\mathbf{x}_{m,t}) d\mathbf{x}_{m,t} \quad (10.1)$$

where  $u_{n,0..t}$  is the sequence of motion control actions up to time  $t$ . We note that this formalism is analogous to the one previously introduced in Section 6.1.1 by Equation (6.2). Building on the notation defined in Section 9.2, where  $D_{n,t}$  is the detection data, we formulate this update in Algorithm 6 on line 4.

As previously discussed in [35], as a result of the approximative nature of this formalism, there are certain limitations to this approach. Due to the fact that robot  $\mathcal{R}_m$  integrates its position belief into that of robot  $\mathcal{R}_n$  upon detection, subsequent detections would induce multiple integrations of this belief, ultimately leading to an overconfident (and possibly erroneous) belief of the actual pose. Fox et al. remedy this shortcoming by considering two rules: (i) their approach does not consider negative sights (no detection) of other robots, and (ii) they define a minimum travel distance which a robot has to complete before detecting a same robot again. Although rule (i) is a practical consideration, rule (ii) limits the scalability and robustness of the approach. In fact, it does not respect our design goals of full mobility and any-time observations (see Sec. 8.1). We will see in the following sections how our approach aims to alleviate this problem by exploiting a reciprocal sampling method.

---

**Algorithm 6** MultiRob\_Recip\_MCI( $X_{n,t-1}, u_{n,t}, z_{n,t}, D_{n,t}$ )

---

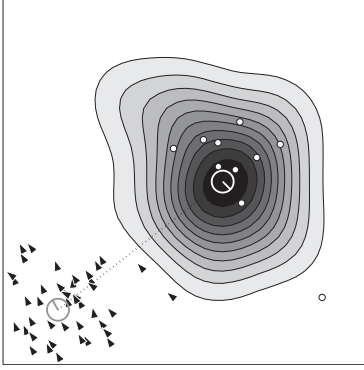
```

1:  $\bar{X}_{n,t} = X_{n,t} = \emptyset$ 
2: for  $i = 1$  to  $M$  do
3:    $\mathbf{x}_{n,t}^{[i]} \leftarrow \text{Motion\_Model}(u_{n,t}, \mathbf{x}_{n,t-1}^{[i]})$ 
4:    $w_{n,t}^{[i]} \leftarrow \text{Detection\_Model}(D_{n,t}, \mathbf{x}_{n,t}^{[i]}, w_{n,t}^{[i]})$ 
5:    $\bar{X}_{n,t} \leftarrow \bar{X}_{n,t} + \langle \mathbf{x}_{n,t}^{[i]}, w_{n,t}^{[i]} \rangle$ 
6: end for
7: for  $i = 1$  to  $M$  do
8:    $r \sim \mathcal{U}(0, 1)$ 
9:   if  $r \leq (1 - \alpha)$  then
10:     $\mathbf{x}_{n,t}^{[i]} \leftarrow \text{Sampling}(\bar{X}_{n,t})$ 
11:   else
12:     $\mathbf{x}_{n,t}^{[i]} \leftarrow \text{Reciprocal\_Sampling}(D_{n,t}, \bar{X}_{n,t})$ 
13:   end if
14:    $X_{n,t} \leftarrow X_{n,t} + \langle \mathbf{x}_{n,t}^{[i]}, \frac{1}{M} \rangle$ 
15: end for
16: return  $X_{n,t}$ 

```

---





**Figure 10.1:** Example of reciprocal sampling. The detecting robot (in black) possesses 50 particles. Ten new (reciprocal) particles are sampled according to the detection model, and will be added (with replacement) to the detected robot's belief (the reciprocal samples are shown by white circles). The model shown here employs noise values  $\sigma_r = 0.15$  and  $\sigma_\phi = 0.15$ .

## 10.2 Reciprocal Sampling

In addition to using a robot detection model for updating the belief representation  $\text{Bel}(\mathbf{x}_{n,t})$ , our approach relies on a *reciprocal* sampling method. Let us refer to the iterative process described in Algorithm 6: instead of sampling a new particle pose  $\mathbf{x}_{n,t}^{[i]}$  from  $\text{Bel}(\mathbf{x}_{n,t-1}^{[i]})$  in line 10, the reciprocal MCL routine in line 12 samples from the detection model  $q$ , according to Eq. (9.6). Thus, samples are drawn at poses which are probable given reciprocal robot observations, and which are independent of the previous belief  $\text{Bel}(\mathbf{x}_{n,t-1})$ . By defining a reciprocal sampling proportion  $\alpha$ , particles are sampled from the robot's own belief with a probability  $1 - \alpha$ , and with a probability of  $\alpha$  from the probability density function proposed by the detection model. The advantages of this procedure are twofold. Firstly, as the reciprocal sampling method exploits the information available in the set of neighboring robots, it creates particles in areas of the pose space which are likely to be significant, and thus it allows for very small particle set sizes. Secondly, by sampling new particles from the detection model, the method introduces a variance proportional to that of the relative detection sensors into the belief of the detected robot (this proportion can be tuned by varying  $\alpha$ ), and effectively mitigates overconfidence.

Algorithm 7 shows the stand-alone routine where line 4 represents the sampling step. There are a multitude of methods which can be applied to sample from a given distribution. In our particular case (multi-modal Gaussians), sampling from the detection model  $q$  is cheap. For more complex probability density functions, sophisticated and efficient methods such as slice sampling [81] can be employed. Figure 10.1 shows an example of reciprocal sampling, where 10 new particles are drawn from the detection model. These new particles subsequently replace ten particles in the detected robot's current belief (not shown).

---

### Algorithm 7 $\text{Reciprocal\_Sampling}(D_{n,t}, \bar{X}_{n,t})$

---

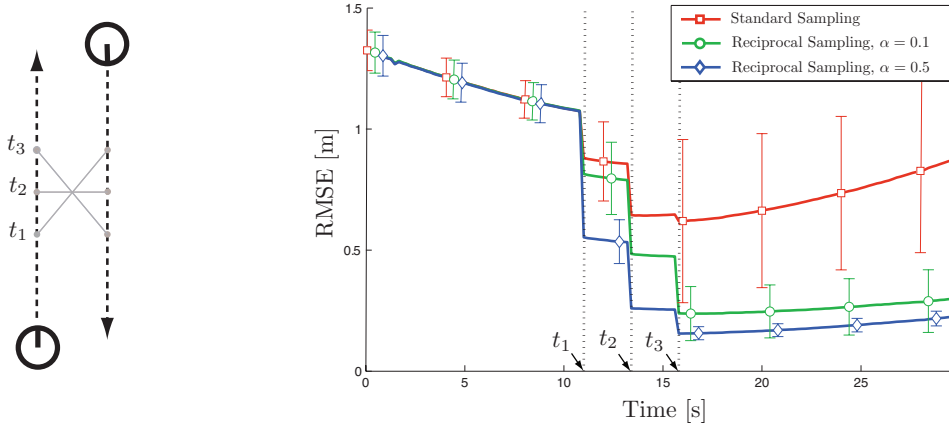
```

1: if  $D_{n,t} = \emptyset$  then
2:    $\mathbf{x} \leftarrow \text{Sampling}(\bar{X}_{n,t})$ 
3: else
4:    $\mathbf{x} \sim Q(\mathbf{x})$ 
5: end if
6: return  $\mathbf{x}$ 

```

---

We illustrate the effect of reciprocal robot detections by performing a short experiment involving two Khepera III robots, one of which is initially localized. Figure 10.2 shows the localization

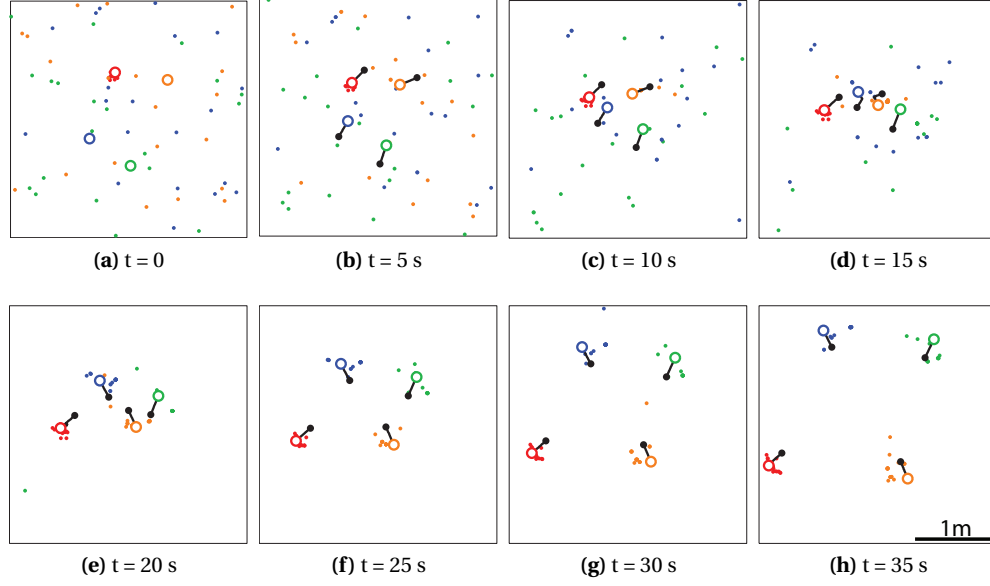


**Figure 10.2:** The panel shows an example of two robots driving past each other. Three detections are made. The results show the localization error for an initially unlocalized robot running our algorithm with three variant reciprocal sampling proportions ( $\alpha = 0$ ,  $\alpha = 0.1$ ,  $\alpha = 0.5$ ). It detects the initially localized robot three times along its path. The standard and reciprocal sampling algorithms (employing 50 particles) are tested 1000 times. The times at which the observations are made are marked by dotted lines (11.2s, 13.6s, 16s). The error bars show the standard deviation.

error for the second, initially unlocalized robot: In comparison to the standard sampling algorithm (Algorithm 6 with  $\alpha = 0$ ), we see that the reciprocal sampling algorithm (Algorithm 6 with  $\alpha > 0$ ) reduces the localization error by taking better advantage of information available at the localized team-member. Additionally, in this case, where the first robot is well localized during this short time span, an increased reciprocal sampling proportion  $\alpha$  brings a greater benefit to the robot that is lost (due to the higher probability of drawing accurate samples). However, in cases where there is no knowledge on whether a given robot is well localized or not, small values for  $\alpha$  produce better results (see Figure A.1d in Appendix A). We note that the error bars are significantly smaller for the reciprocal sampling than for the standard sampling. This result indicates increased robustness and reliability when using reciprocal sampling.

Appendix A performs an analysis of the reciprocal sampling algorithm on hand of a simplified scenario. The conclusions of the analysis can be summarized as follows. For very large particle sets, standard sampling algorithms (with  $\alpha = 0$ ) will perform at least as well, or even better than reciprocal sampling algorithms. However, for small particle sets, and especially when the detection noise values are moderate, the reciprocal sampling algorithm shows a clear acceleration of the convergence (to a position estimate). Moreover, this benefit is maintained for a large interval of values for  $\alpha$ , when the detection noise is moderate. In conclusion, we note that while the noise values are given by the real system at hand, parameters such as  $\alpha$  and the number of particles can be tuned and thus adapted to available computational and communication resources.

Finally, to illustrate the localization process, Figure 10.3 shows eight snapshots based on real data from an experiment performed over an interval of 35 s during which one robot (in red) is initially localized. The snapshots show that, even though the robots employ a very sparse particle set (20 particles per robot over a  $9 \text{ m}^2$  space), the reciprocal sampling algorithm is able to produce particles which coincide with the approximate true locations of the robots.



**Figure 10.3:** The figure shows eight snapshots with 5 s intervals of an experimental run on the team of four Khepera III robots. Each robot employed 20 particles with a reciprocal proportion  $\alpha = 0.1$ . The black lines show the trajectories completed in the time intervals between snapshots, with the filled black dots representing the robot positions at the end of the previous snapshots. The red robot was initially localized.

### 10.3 Particle Clustering

The algorithm complexity of the detection model  $q$  (Eq. (9.6)) leads to  $O(M^2)$  for Algorithm 6 (assuming a neighborhood size  $|\mathcal{N}_n| = 1$ ). This cost can be prohibitive for a large number of particles  $M$  (i.e., large with respect to available computational resources). Also, a multi-robot system may have communication constraints that make sending large particle sets infeasible. Hence, even though the method applied in this paper allows for very small particle sets, we resort to a clustering method to further reduce the computational and communication overhead.

Let us consider a case where robot  $\mathcal{R}_m$  detects robot  $\mathcal{R}_n$ . For better clarity in the following derivations, we will assume that  $|\mathcal{N}_{n,t}| = 1$ . The goal of the clustering method is to reduce the number of operations needed to compute the probability density function  $q$ . Thus, for every detection that it makes, robot  $\mathcal{R}_m$  resorts to a clustering method which summarizes its set  $X_m$  composed of  $M$  particles to a set  $\bar{X}_m$  composed of  $K$  cluster abstractions (or centroids), reducing the overall computational cost to  $O(MK)$  (this clustering routine is detailed later, in Algorithm 8 of Section 10.3.1). The resulting partition of the particle set is denoted  $\mathcal{C}_m$ , with  $|\mathcal{C}_m| = K$ . An individual cluster  $c_m^{[k]} \in \mathcal{C}_m$  is defined as the set of particles

$$c_m^{[k]} = \{\langle \mathbf{x}_m^{[i]}, w_m^{[i]} \rangle \mid f(\langle \mathbf{x}_m^{[i]}, w_m^{[i]} \rangle, \cdot) = k\}, \quad (10.2)$$

where  $f$  is a function mapping a particle to a cluster index. Also, we define  $\bar{c}_m^{[k]}$  as the data

abstraction of cluster  $c_m^{[k]}$ , representing all particles in its set by the tuple

$$\bar{c}_m^{[k]} = \langle \bar{\mathbf{x}}_m^{[k]}, \bar{w}_m^{[k]}, \bar{\boldsymbol{\mu}}_m^{[k]}, \bar{\xi}_m^{[k]} \rangle, \quad (10.3)$$

where  $\bar{\boldsymbol{\mu}}_m^{[k]}$  is a two dimensional vector and  $\bar{\xi}_m^{[k]}$  is a covariance matrix (which we detail later). Thus,  $\bar{X}_m = \{\bar{c}_m^{[k]} \mid c_m^{[k]} \in \mathcal{C}_m\}$  is the set of  $K$  cluster abstractions. Finally, we denote the clustered detection data as  $\bar{d}_{mn} = \langle \hat{r}_{mn}, \hat{\phi}_{mn}, \bar{X}_m \rangle$ , which is sent in place of the unclustered detection data  $d_{mn}$ .

Formally, given the notation introduced above, finding an optimal particle clustering is equivalent to solving the following optimization problem

$$\min_{\bar{d}_{mn}} \mathbf{D}(q(\mathbf{x}_n | d_{mn}) \parallel \bar{q}(\mathbf{x}_n | \bar{d}_{mn})), \quad (10.4)$$

where  $\bar{q}$  is an approximated detection model, and  $\mathbf{D}$  a distance measure between two probability density functions.

To simplify the mathematical developments that will follow, Equation (10.4) makes  $q$  explicitly depend on  $\mathbf{x}_n$  and the detection data  $d_{mn}$ . We combine Equations (9.6) and (9.10), and rewrite our probability density function  $q$  as

$$q(\mathbf{x}_n | d_{mn}) = \eta \cdot \sum_{\langle \mathbf{x}_m^{[i]}, w_m^{[i]} \rangle \in X_m} \Phi \left( \mathbf{T}_e^p(\mathbf{x}_m^{[i]}, \mathbf{x}_n); \boldsymbol{\mu}, \xi \right) \cdot w_m^{[i]}, \quad (10.5)$$

where  $\Phi(\cdot; \boldsymbol{\mu}, \xi)$  is a multivariate normal probability density function with mean  $\boldsymbol{\mu} = [\hat{r}_{mn}, \hat{\phi}_{mn}]^T$ , covariance  $\xi$  (as in Equation (9.11)), and  $\eta$  is a normalization factor.  $\mathbf{T}_e^p$  is the transformation from Euclidean to polar coordinates, as defined in Equation (9.2).

Jain et al. [55] point out that in a typical clustering task, the actual grouping (or **clustering**) and cluster data abstraction (or **cluster representation**) are separate components of the task and are commonly treated sequentially. Hence, we deal with our problem by dividing it into the two following sub-problems: (i) we consider the set of particles  $X_m$  and find an optimal way to create a partition  $\mathcal{C}_m$ , and (ii) we consider an arbitrary cluster  $c_m^{[k]}$  in  $\mathcal{C}_m$  and find an optimal way to determine its cluster abstraction  $\bar{c}_m^{[k]}$ . For a given set  $X_m$ , these two steps together ultimately lead to a set of cluster abstractions  $\bar{X}_m$ , which, instead of  $X_m$ , is included into the detection data tuple  $\bar{d}_{mn}$  for every new detection made. The following paragraphs detail our low-cost clustering approach that aims to meet these specifications.

### 10.3.1 Clustering Algorithm

The optimal, combinatorial solution to the clustering problem of Equation 10.4 requires the evaluation of a very large number of partitions (the number of ways to partition a set of  $M$  data points into  $K$  non-empty clusters is given by the Stirling number of the second kind). Even though efficient approaches have been proposed [62], combinatorial solutions still remain prohibitively expensive. Given the usefulness of clustering in a large range of disciplines, many non-combinatorial clustering approaches have been proposed [55]. Yet, since our goal is to

**Algorithm 8**  $\text{Cluster}(X_{m,t}, K)$ 


---

```

1:  $\overline{X}_m \leftarrow \emptyset$ 
2:  $c_m^{[1]} \leftarrow X_m$ 
3:  $\mathcal{C}_m \leftarrow c_m^{[1]}$ 
4: for  $k = 1$  to  $K - 1$  do
5:    $k_{max}, \text{dim} \leftarrow \text{find\_highest\_variance\_cluster}(\mathcal{C}_m)$ 
6:    $c_m^{[k_{max}]}, c_m^{[k+1]} \leftarrow \text{split\_cluster}(c_m^{[k_{max}]}, \text{dim})$ 
7:    $\mathcal{C}_m \leftarrow \mathcal{C}_m + c_m^{[k+1]}$ 
8: end for
9: for  $k = 1$  to  $K$  do
10:   $\overline{c}_m^{[k]} \leftarrow \text{assign\_data\_abstraction}(c_m^{[k]})$ 
11:   $\overline{X}_m \leftarrow \overline{X}_m + \overline{c}_m^{[k]}$ 
12: end for
13: return  $\overline{X}_m$ 

```

---

reduce the final complexity of our algorithm, the complexity of the actual clustering algorithm must be at most equal to  $O(MK)$ . One of the most commonly used low-cost clustering methods is the k-means algorithm [74]. It starts off with a random initial cluster assignment and iteratively reassigns clusters until a convergence criterion is met or a maximum number of iterations  $L$  is attained. Although the algorithm has a low time complexity  $O(MKL)$ , its main disadvantage is that it is sensitive to the initial cluster assignment. The variant ISODATA algorithm [7] is also an iterative clustering algorithm with a time complexity of  $O(MKL)$ , with the additional capability to split and merge clusters according to predefined threshold values. It is therefore more flexible than the k-means and able to find the optimal partition, provided that the user is able to define correct threshold values. Non-iterative, incremental clustering algorithms have the advantage that they are even less time consuming than iterative algorithms. The leader algorithm [46] is the simplest of that kind. Data points are incrementally assigned to existing clusters based on a distance metric, with new clusters being created if all distance measures exceed a predefined criterion. Yet, given the algorithms incremental nature, the final clustering result is dependent on the order of the assignments made.

Having tested the methods mentioned above, we finally settled on a non-iterative, order-independent, non-parametric approach that produces a predefined number of  $K$  clusters. Our solution resorts to the construction of multidimensional binary trees, and relates to the multi-dimensional divide-and-conquer method described by Bentley [10]. Our method consists of a two-dimensional sorting algorithm which repetitively separates the particle set along the mean of the dimension producing the highest variance, until the predefined maximum number of clusters  $K$  is attained. We note that splitting along the median (as done in [10]) incurs a complexity of  $O(MK \log M)$ , thus, we use the mean instead. A description of this algorithm is shown in Algorithm 8. The function in line 5 has a complexity  $O(M)$ , the function in line 6 has a complexity  $O(|c_m^{[k_{max}]}|)$ , and function in line 10 has a complexity  $O(|c_m^{[k]}|)$ . Hence, the total algorithm cost is  $O(MK)$ .

### 10.3.2 Cluster Abstraction

For an arbitrary cluster  $c_m^{[k]}$ , we have the non-summarized detection data  $d_{mn}^{[k]} = \langle \hat{r}_{mn}, \hat{\phi}_{mn}, c_m^{[k]} \rangle$ . The problem of finding an optimal cluster abstraction  $\bar{c}_m^{[k]}$  can, thus, be formalized as

$$\min_{\bar{c}_m^{[k]}} \mathbf{D}_{\text{KL}}(q(x_n | d_{mn}^{[k]}) || \bar{q}(x_n | \bar{d}_{mn}^{[k]})), \quad (10.6)$$

where  $\mathbf{D}_{\text{KL}}$  is the Kullback-Leibler divergence (other divergence measures can also be used), and  $\bar{d}_{mn}^{[k]} = \langle \hat{r}_{mn}, \hat{\phi}_m, \bar{c}_m^{[k]} \rangle$  is the summarized detection data. Let us consider the following. Given a point  $\bar{\mathbf{x}}_m^{[k]} = [\bar{x}_m^{[k]}, \bar{y}_m^{[k]}, \bar{\psi}_m^{[k]}]^\top$ , and the probability density function

$$\bar{q}(\mathbf{x}_n | \bar{d}_{mn}^{[k]}) = \Phi \left( \mathbf{T}_e^p(\bar{\mathbf{x}}_m^{[k]}, \mathbf{x}_n); \bar{\boldsymbol{\mu}}_m^{[k]}, \bar{\boldsymbol{\xi}}_m^{[k]} \right), \quad (10.7)$$

the Kullback-Leibler divergence between  $q$  and  $\bar{q}$  is minimal if

$$\bar{\boldsymbol{\mu}}_m^{[k]} = \frac{1}{|c_m^{[k]}|} \sum_{\mathbf{x}_m^{[i]} \in c_m^{[k]}} \mathbf{v}_m^{[k,i]}, \quad (10.8)$$

$$\bar{\boldsymbol{\xi}}_m^{[k]} = \frac{1}{|c_m^{[k]}| - 1} \sum_{\mathbf{x}_m^{[i]} \in c_m^{[k]}} \left( \mathbf{v}_m^{[k,i]} - \bar{\boldsymbol{\mu}}_m^{[k]} \right) \left( \mathbf{v}_m^{[k,i]} - \bar{\boldsymbol{\mu}}_m^{[k]} \right)^\top \quad (10.9)$$

are the mean and covariance of  $\mathbf{v}_m^{[k,i]} = \mathbf{T}_e^p(\bar{\mathbf{x}}_m^{[k]}, \tilde{\mathbf{x}}_m^{[i]})$ , with

$$\tilde{\mathbf{x}}_m^{[i]} = \begin{bmatrix} \tilde{x}_m^{[i]} \\ \tilde{y}_m^{[i]} \end{bmatrix} = \begin{bmatrix} x_m^{[i]} + \hat{r}_{mn} \cos(\hat{\phi}_{mn} + \psi_m^{[i]}) \\ y_m^{[i]} + \hat{r}_{mn} \sin(\hat{\phi}_{mn} + \psi_m^{[i]}) \end{bmatrix}. \quad (10.10)$$

Our proof is inspired by the general methodology shown in [48]. Let us use the polar coordinate frame  $\rho, \vartheta$  with its origin at  $\bar{\mathbf{x}}_m^{[k]}$  (cf.  $\mathbf{T}_e^p(\bar{\mathbf{x}}_m^{[k]}, \cdot)$ ). The probability density function  $\bar{q}(\mathbf{x}_n | \bar{d}_{mn}^{[k]})$  is part of the exponential family of functions and can be written

$$\bar{q}(\mathbf{x}_n | \bar{d}_{mn}^{[k]}) = h(\rho, \vartheta) g(\bar{\boldsymbol{\mu}}_m^{[k]}, \bar{\boldsymbol{\xi}}_m^{[k]}) e^{\eta(\bar{\boldsymbol{\mu}}_m^{[k]}, \bar{\boldsymbol{\xi}}_m^{[k]}) \tau(\rho, \vartheta)} \quad (10.11)$$

with  $\tau(\rho, \vartheta) = [\rho, \rho^2, \vartheta, \vartheta^2, \rho\vartheta, 1]^\top$ . The Kullback-Leibler divergence

$$\mathbf{D}_{\text{KL}}(q || \bar{q}) = \int_{-\infty}^{\infty} q(\mathbf{x}_n | d_{mn}^{[k]}) \log \frac{q(\mathbf{x}_n | d_{mn}^{[k]})}{\bar{q}(\mathbf{x}_n | \bar{d}_{mn}^{[k]})} d\mathbf{x}_n \quad (10.12)$$

is minimized by taking its gradient with respect to  $\bar{\boldsymbol{\mu}}_m^{[k]}$  and  $\bar{\boldsymbol{\xi}}_m^{[k]}$ , and setting it to zero:

$$-\nabla \log g(\bar{\boldsymbol{\mu}}_m^{[k]}, \bar{\boldsymbol{\xi}}_m^{[k]}) = \nabla \eta(\bar{\boldsymbol{\mu}}_m^{[k]}, \bar{\boldsymbol{\xi}}_m^{[k]})^\top \mathbb{E}_{p_{mn}} [\tau(\rho, \vartheta)]. \quad (10.13)$$

Note that the integral of  $\bar{q}$  is one, and thus we have

$$g(\bar{\mu}_m^{[k]}, \bar{\xi}_m^{[k]}) \int_{-\pi}^{\pi} \int_0^{\infty} h(\rho, \vartheta) e^{\eta(\bar{\mu}_m^{[k]}, \bar{\xi}_m^{[k]}) \tau(\rho, \vartheta)} d\rho d\vartheta = 1. \quad (10.14)$$

Computing the gradient on the left and right side of the above equation with respect to  $\bar{\mu}_m^{[k]}$  and  $\bar{\xi}_m^{[k]}$  yields

$$-\nabla \log g(\bar{\mu}_m^{[k]}, \bar{\xi}_m^{[k]}) = \nabla \eta(\bar{\mu}_m^{[k]}, \bar{\xi}_m^{[k]})^T \mathbb{E}_{\bar{q}} [\tau(\rho, \vartheta)]. \quad (10.15)$$

Finally, by combining Equations (10.13) and (10.15), we obtain

$$\mathbb{E}_{\bar{q}} [\tau(\rho, \vartheta)] = \mathbb{E}_q [\tau(\rho, \vartheta)] \quad (10.16)$$

From this we conclude that mean and covariance, respectively, of  $q$  and  $\bar{q}$  must be the same in order to minimize their Kullback-Leibler divergence, hence Equations (10.8) and (10.9).

We note that the above equations do not take into account the uncertainty of the range and bearing observations. Thus, we propose a variant detection model  $\bar{q}$  (cf. Equation (9.6)) that explicitly takes into account noise. We have

$$\bar{q}(\mathbf{x}_n | \bar{d}_{mn}) = \eta \cdot \sum_{\bar{c}_m^{[k]} \in \bar{X}_m} \Phi \left( \mathbf{T}_e^p(\bar{\mathbf{x}}_m^{[k]}, \mathbf{x}_n); \bar{\mu}_m^{[k]}, \bar{\xi}_m^{[k]} + \xi \right) \cdot \bar{w}_m^{[k]} \quad (10.17)$$

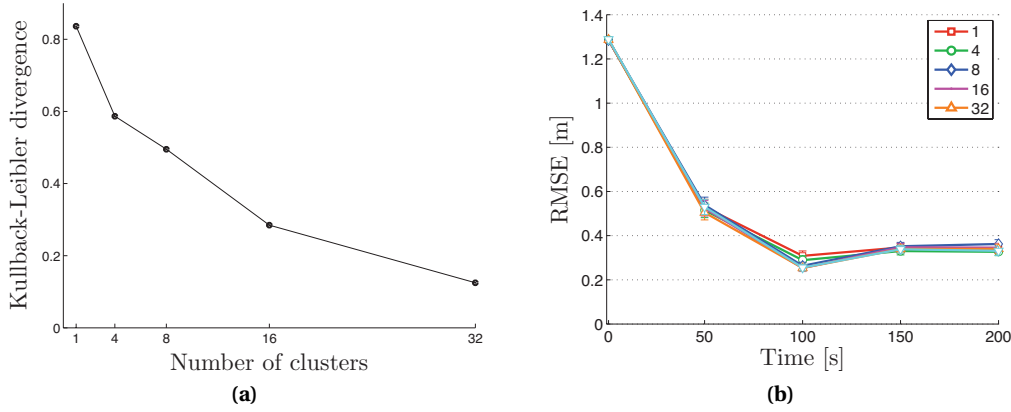
where  $\bar{\mu}_m^{[k]}$  and  $\bar{\xi}_m^{[k]} + \xi$  approximate the true mean and covariance, respectively, in the presence of noise (we remind the reader that  $\xi = \text{diag}([\sigma_r^2, \sigma_\theta^2])$ ). Indeed, finding a closed form solution for the true values is intractable. However, if the set of particles  $c_m^{[k]}$  is densely populated, our approximation is very good. Moreover, if the particle positions coincide, and if for a given cluster  $c_m^{[k]}$  the point  $\bar{\mathbf{x}}_m^{[k]}$  is its center of mass, the solution is optimal. Hence, we complete the data abstraction  $\bar{c}_m^{[k]} = \langle \bar{\mathbf{x}}_m^{[k]}, \bar{w}_m^{[k]}, \bar{\mu}_m^{[k]}, \bar{\xi}_m^{[k]} \rangle$  (cf. Equation (10.3)) with  $\bar{\mathbf{x}}_m^{[k]}$  as the weighted center of mass, and  $\bar{w}_m^{[k]}$  the cumulative weight

$$\bar{\mathbf{x}}_m^{[k]} = \frac{1}{\bar{w}_m^{[k]}} \cdot \sum_{\langle \mathbf{x}_m^{[i]}, w_m^{[i]} \rangle \in c_m^{[k]}} w_m^{[i]} \cdot \mathbf{x}_m^{[i]} \quad (10.18)$$

$$\bar{w}_m^{[k]} = \sum_{\langle \mathbf{x}_m^{[i]}, w_m^{[i]} \rangle \in c_m^{[k]}} w_m^{[i]}. \quad (10.19)$$

Finally, we note that the constraints given by our approximated detection model  $\bar{q}$  motivate the choice of a clustering algorithm which clusters densely located particles into common clusters (a condition which is satisfied by Algorithm 8).

Figure 10.4a shows the Kullback-Leibler divergence between the full and the approximated detection models  $q$  and  $\bar{q}$ , calculated from a data set gathered by ten robots. The more clusters we employ in the clustering method, the smaller the divergence to the true probability density function. This shows that our clustering method produces a valid representation of



**Figure 10.4:** (a) The Kullback-Leibler divergence between the full and approximated detection models, as a function of the number of clusters employed by the clustering method. (b) Average localization error over 100 evaluations. The localization algorithm is tested, employing the clustering method using  $\{1, 4, 8, 16, 32\}$  clusters. The errorbars show 95% confidence intervals.

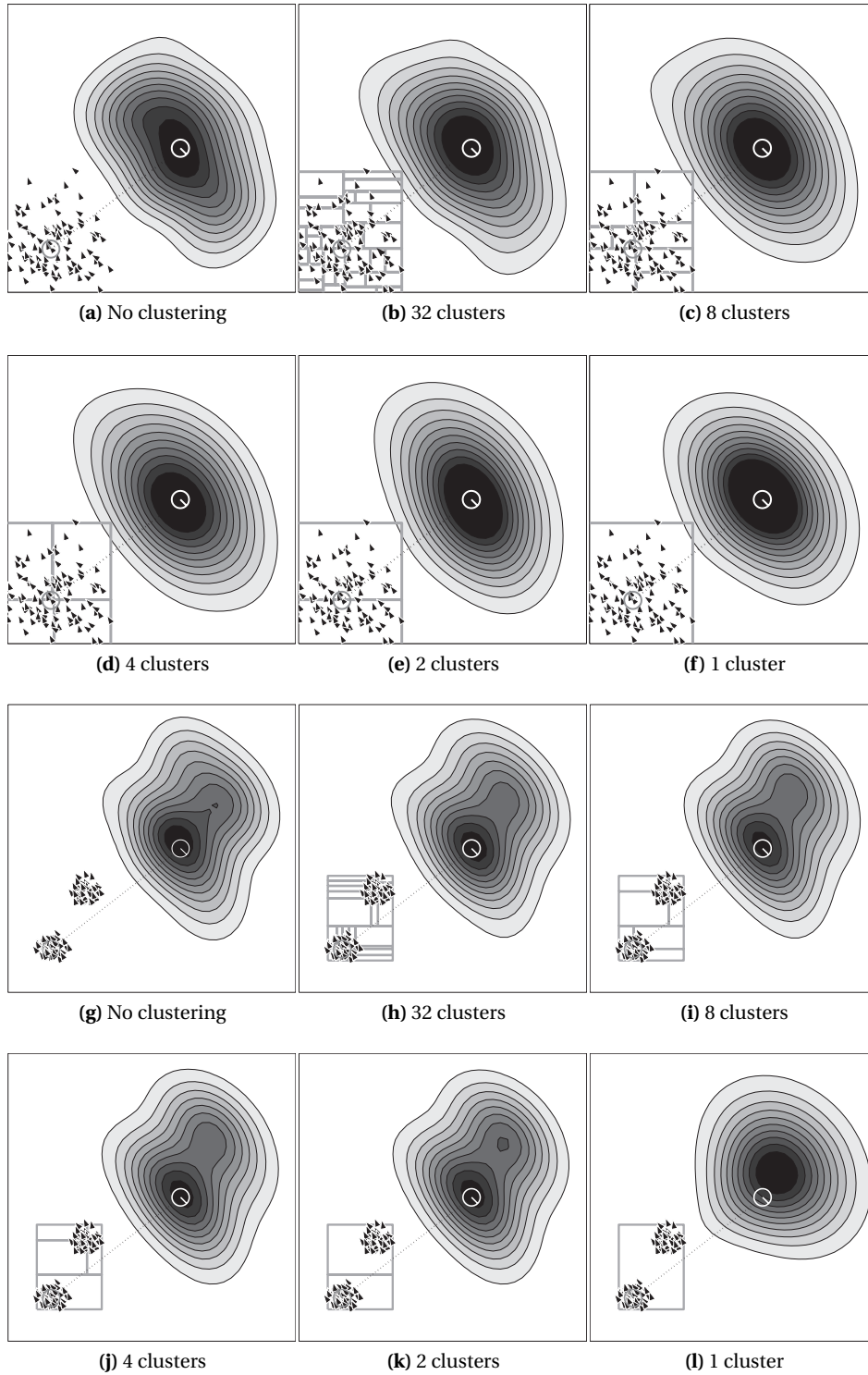
the original probability density functions. Figure 10.4b shows the localization performance when employing the clustering method for a variable number of clusters  $K$ . We note that the difference of performance between maximal clustering ( $K = 1$ ) and modest clustering ( $K = 32$ ) is insignificant.

Figure 10.5 shows examples of final cluster partitions for six different total numbers of clusters, performed on an identical set of 100 particles. We consider two cases. The top two rows (a)–(f) use a particle set sampled from a Gaussian distribution centered at the actual robot position. The bottom two rows (g)–(l) use a particle set sampled from a bimodal Gaussian distribution, with one of the modes centered at the actual robot position (and using 60% of the particles), and the other mode centered at a faulty location. In the first case, we note that for all cluster numbers, including maximal clustering with  $K = 1$ , the detection model is well approximated. In the second case, the bimodal particle distribution is well approximated up to  $K = 2$ , and for  $K = 1$  (l) results in an off-center approximation of the detection model. Indeed, the panels indicate that a  $k$ -modal particle distribution is best captured by  $K \geq k$  clusters. As a general remark, we note that a set of  $M$  particles that is clustered into  $K < M$  clusters will provide more information than a set of just  $K$  particles, (i) because the cluster abstraction contains summarized information (mean and covariance values) that is obtained from the whole particle distribution, and (ii) because the clustering is recalculated at each iteration, for each new particle distribution and each new range and bearing measurement.

## 10.4 Performance

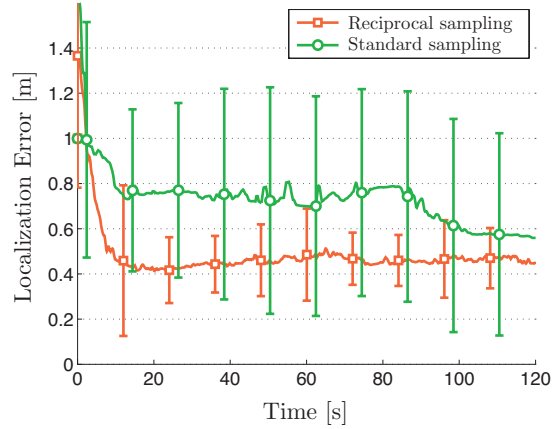
We performed a number of experiments to test our approach. In a first experiment, we ran our algorithm on-board, in real-time on a team of four robots. We considered a scenario where one robot continuously receives position fixes at a frequency of 1 Hz from the overhead camera system, while the three other robots are unlocalized (and rely only on our relative positioning method). To begin, the robots are placed randomly in an arena of reduced size ( $9 \text{ m}^2$ , without obstacles). Each robot has a total of 50 particles, and a reciprocal sampling





**Figure 10.5:** The detection model (here with range and bearing noise  $\sigma_r = 0.15$  and  $\sigma_\theta = 0.15$ ) is projected on the detected robot (in white). Final cluster partitions are visualized on the particles of the detecting robot. Two variant particle distributions are considered, one for (a)—(f), and one for (g)—(l). For each of the two cases, the number of clusters  $K$  employed by the clustering algorithm is: 100, 32, 8, 4, 2, 1, for a total number of particles  $M = 100$ .

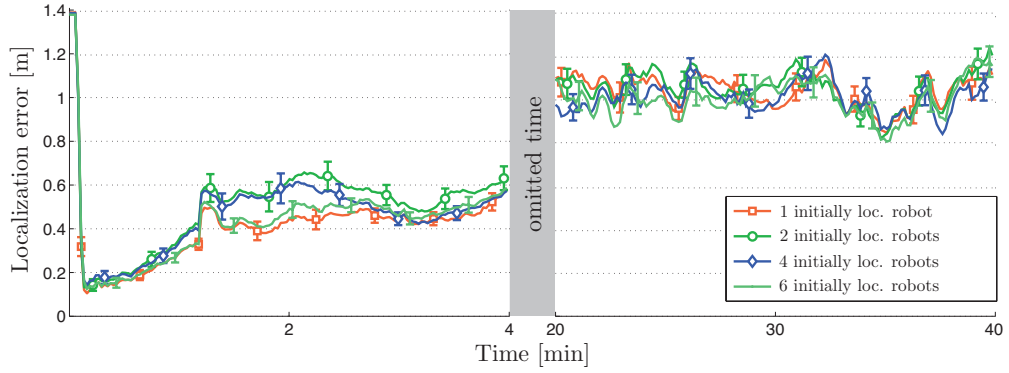
**Figure 10.6:** Average localization error obtained over 10 runs of 2 min duration, performed in real-time on 4 Khepera III robots with 50 particles each, and the clustering routine with  $K = 1$ . The algorithm is tested in two variants, (i) with a reciprocal sampling proportion of  $\alpha = 0.02$ , and (ii) without reciprocal sampling ( $\alpha = 0$ ). One robot is continuously localized, the three other robots are initially unlocalized—the shown error statistic excludes the localized robot. The error bars show the standard deviations.



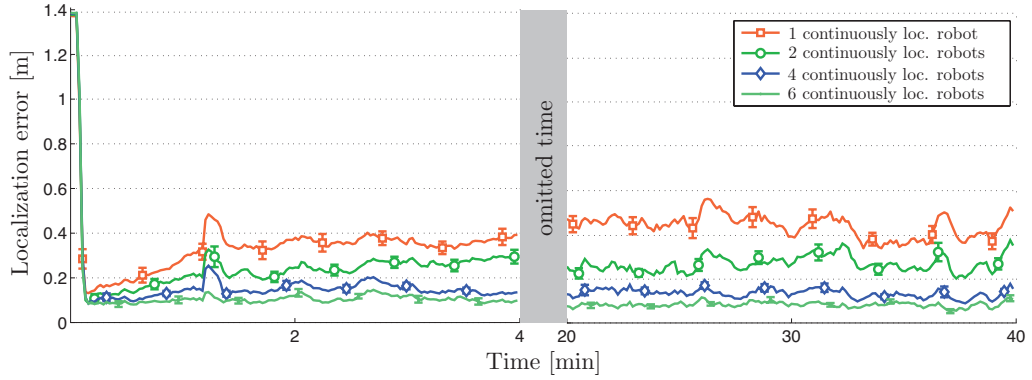
proportion  $\alpha = 0.02$ . We repeated the experiment 10 times, and each run lasts 2 minutes. Figure 10.6 shows the positioning error averaged over the number of runs for the initially unlocalized robots. The results show that, throughout the duration of this experiment, the reciprocal sampling algorithm outperforms the standard sampling algorithm, and is able to reduce the average positioning error to approximately 0.4 m. This experiment shows a successful application of our approach on an actual embedded platform, and also confirms the feasibility of our algorithm with respect to real-time constraints as well as resource constraints.

To test the long-term performance of our systems, we performed an experiment that lasted approximately 40 minutes, involving ten robots in our large  $3 \text{ m} \times 5 \text{ m}$  arena (shown in Figure 4.5, including a cross-shaped obstacle). Figure 10.7 shows results for a variable number of initially localized robots, and Figure 10.8 shows results for a variable number of continuously localized robots. The experimental data is evaluated over 50 iterations, where at each iteration the set of localized robots is randomly allocated, with the remaining robots relying on our relative positioning method for localization. We evaluated the localization error for all robots in the system, and for better clarity, omit 16 minutes worth of data points in the plots. By considering the results in Figure 10.7, we note that without a continuous update of the absolute position, the overall error increases over time. However, the first minute shows the benefits of collaboration, since a single localized robot allows the whole team to localize quite quickly. The results in Figure 10.8 show that by providing at least one robot in the team with continuous absolute position updates, the localization error is bounded and constant over time. Also, we note that by increasing the number of continuously localized robots, the error decreases. In order to assess the worst performance, we compute the maximum error value (over all robots) for every time step, throughout the whole run. These data points are summarized in the form of boxplots in Figure 10.10. The results indicate that by increasing the number of continuously localized robots, the maximum localization error decreases consistently, as we are able to positively affect the performance of all other robots in the team.

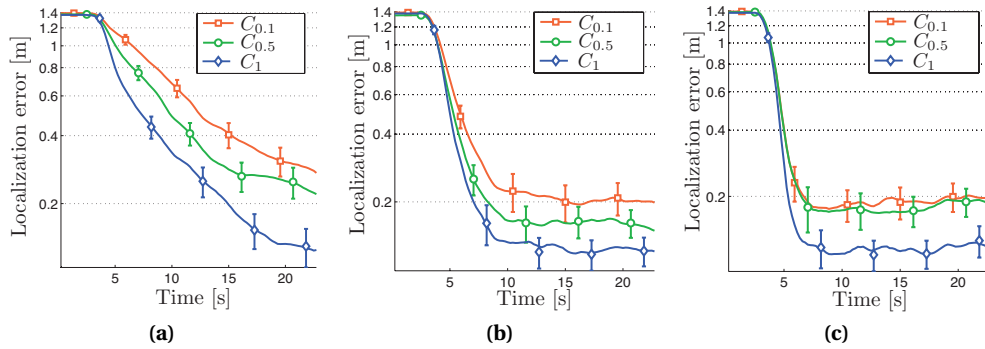
Using the same setup, we performed a 25 s experiment to test the effect of collaboration on the convergence of the position estimate. To this means, we vary the frequency with which the robots exchange detection data. Per default, the robots collaborate at an average frequency of 2 Hz. We remove detection messages from our data log a posteriori, producing three additional data sets with collaboration rates at 1 Hz, 0.5 Hz, 0.1 Hz. These schemes are referred to as  $C_1$ ,  $C_{0.5}$ , and  $C_{0.1}$ , respectively. Figure 10.9 shows the localization performance



**Figure 10.7:** Average localization error (for all robots) obtained over 50 evaluations of a data set for an experiment of approx. 40 min duration, performed on 10 Khepera III robots with 50 particles each, and reciprocal sampling proportion of  $\alpha = 0.02$ . We vary the number of robots that are localized at the start. The error bars show a 95% confidence interval.

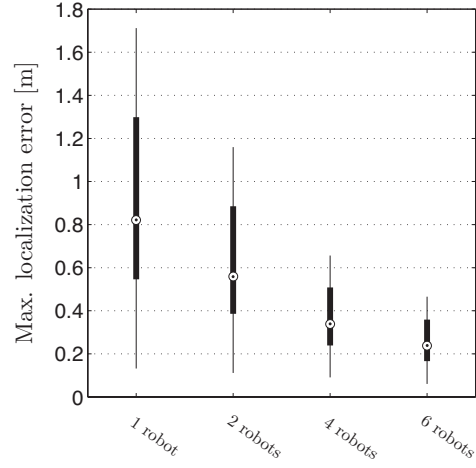


**Figure 10.8:** Average localization error (for all robots) obtained over 50 evaluations of a data set for an experiment of approx. 40 min duration, performed on 10 Khepera III robots with 50 particles each, a reciprocal sampling proportion of  $\alpha = 0.02$ . We vary the number of continuously localized robots (ground truth positioning information is sent periodically, at an average frequency of 2.6 Hz). The error bars show a 95% confidence interval.



**Figure 10.9:** Localization error, shown in log-scale, for 50 evaluations of a 25 s data set for an experiment performed on 10 Khepera III robots. Our localization algorithm employs 50 particles per robot and a reciprocal sampling rate  $\alpha = 0.02$ . We test three variant collaboration rates:  $C_1$ ,  $C_{0.5}$ , and  $C_{0.1}$ . (a) 1 robot is continuously localized. (b) 2 robots are continuously localized. (c) 4 robots are continuously localized. The error bars show 95% confidence intervals.

**Figure 10.10:** Maximum localization error (for all robots, at any given time) obtained over 50 evaluations of a data set for an experiment of approx. 40 min duration performed on 10 Khepera III robots with 50 particles each, and a reciprocal sampling proportion of  $\alpha = 0.02$ . We vary the number of robots that receive continuous absolute positioning updates. The boxplots show the median, 25th, and 75th percentile of the data.



for variable message failure rates, for 1, 2, and 4 continuously localized robots. Increasing failure rates induce a graceful degradation of the localization performance. This result confirms the algorithm's robustness with respect to communication failures, which ultimately reinforces the underlying asynchronous nature of our collaborative paradigm.

### Summary

In this chapter we presented a fully scalable, probabilistic, multi-robot localization algorithm based on the Monte Carlo method. Its maximal overall complexity is  $O(|\mathcal{N}|MK)$ , where  $|\mathcal{N}|$  is the number of neighboring robots (at a given time, for a given robot in the system),  $M$  the number of particles, and  $K$  an adjustable number of clusters produced by the clustering algorithm. This clustering method has shown to produce increasingly accurate probability density function representations for large  $K$ , and when employed in practice, has shown to perform well even for very small  $K$ . Furthermore, given the asynchronous paradigm of our collaboration strategy, the algorithm's update rate is much higher than the inter-robot message communication rate. Thus, the number of detected neighbors  $|\mathcal{N}|$  is in practice no higher than 1, and the complete routine complexity is reduced to  $O(MK)$ . Thus, the algorithm is fully scalable with respect to the number of robots in the system. In addition, the algorithm poses no communication constraints and shows a graceful performance degradation in case of message failures. Our approach was experimentally validated on teams of two, four and ten real robots, over short as well as long experimental runs. In particular, our long runs have shown the benefits of maintaining a subset of robots that receive regular absolute positioning updates.

# 11 Conclusion

**A**LTHOUGH our work on multi-robot localization is only a small piece of the collaborative localization puzzle, it distinguishes itself from other works with its decentralized and fully scalable approach. The following text will bring our approach into a broader context and discuss its merits critically. We conclude this part of the manuscript by providing an outlook on the domain of collaborative localization techniques.

## 11.1 Related Work

A synopsis of currently available work on multi-robot localization promotes a division of the various approaches into two main categories, as already introduced in Section 8.2: a *multi-centralized* approach, and a *decentralized* approach. We remind the reader that the *multi-centralized* approach distinguishes itself formally from the *decentralized* approach by imposing that each robot in the team maintains a state vector containing the poses of all robots—in other words, each robot maintains a full-system state estimate, versus an estimate of only its own pose.

The *multi-centralized* [82] approach is indeed the more popular (and classical) approach, as it enables the robots to directly take account of inter-robot dependencies and to estimate correlations. However, it entails some inconveniences. In an early work, Roumeliotis et al. [107] enable the distribution of a Kalman estimation scheme by constructing communicating filters, which allows team-members to propagate their state and covariance estimates independently. Yet, as covariance matrix updates occur during each update step and require information exchange between all robots and a centralized processor, the method is particularly vulnerable to single-point failures. In particular, the requirement to update the information in all robots after a single observation of an individual robot assumes a communication infrastructure without any packet loss. The method scales in  $O(N^3)$  with respect to the number of robots, and thus limits its scalability due to the high computational cost. In [73], Martinelli et al. propose an extension to [107], which relaxes the assumptions on relative observations, but without further improving the algorithm's scalability and cost. Howard et al. [53] propose an algorithm

based on maximum likelihood estimation, and validate it on a team of four real robots. Their method relies on periodical information broadcasts, and it is unclear how the method scales and how sensitive it is to local minima. In a recent work, Nerurkar et al. [83] address the reduction of computational complexity and single-point failures by implementing a maximum a posteriori estimation method. Nevertheless, the  $O(N^2)$  computational cost is significant. Also, the proposed method requires synchronous communication among the robots, and its feasibility still remains to be validated on real robots. Mourikis et al. [80] consider the problem of resource-constrained collaborative localization with the goal of deriving optimal sensing frequencies. Yet, as exteroceptive data is dealt with in a centralized way, the sensing frequencies inevitably decrease with an increasing number of robots, thus limiting the scalability of the approach. Cristofaro et al. [25] present a localization algorithm that arguably alleviates the problems described above. The approach is based on an extended information filter, whose implementation is distributed over the robot team members. However, its computational cost increases for each new observation made and it assumes bidirectional synchronous communication, the feasibility of which remains to be evaluated on real robots. Finally, Leung et al. [68] develop a framework based on ‘checkpoints’ which facilitates decentralization of a given localization algorithm. Their method, however, still aims to maintain full-system state estimates on all robots, and remains to be evaluated on real robots.

The category of work representing the *decentralized* approach has an alternative take on the collaborative localization problem: each robot maintains an estimate of only its own pose, and fuses relative observations in an opportunistic fashion. Fox et al. [35] first introduced a multi-robot Monte-Carlo localization algorithm for global localization (i.e., without initialization), that also relaxes noise assumptions as well as inter-robot dependencies. They propose a method with which robots mutually synchronize their position beliefs upon detection, and show successful global localization on two real robots. However, the method has limited scalability due to overconfidence occurring upon multiple robot detections, and no analysis is provided of the algorithm’s processing requirements. Bahr et al. [5] develop a decentralized localization algorithm, based on the extended Kalman filter framework, that is especially well suited for autonomous underwater vehicles with very low data rates. This method, however, allows cyclic updates and, thus, may suffer from overconfidence. In an addition to this work [6], the authors remedy the overconfidence problem, but at the cost of a computationally expensive solution (in particular for a large number of robots and a high frequency of relative observations).

Our work situates itself in the *decentralized* category, and marks itself by our additional, probabilistic sampling strategy (which we termed reciprocal sampling). The benefits of this additional sampling method were demonstrated in previous chapters: the main aims are mitigation of overconfidence and acceleration of convergence. The idea of extending standard MCL with additional sampling methods was first shown by Thrun et al. [120]. The resulting algorithm, named *Mixture* MCL, was shown to increase the robustness of single-robot global localization. Our method differs from the work by Thrun et al. in that it builds on collaborative multi-robot localization algorithms by sampling from the detection model of one or several mobile robots (whose positions are initially unknown)—this is in contrast to the work of Thrun et al., which samples from the detection model of a potentially large set of static environmental features (whose positions have to be known or mapped a priori). Indeed, for complex environments, the method in [120] must be preceded by a fingerprinting process.

## 11.2 Discussion

As indicated in Section 8.2, the decentralized estimation strategy is not an exact method, as long as it fails to track elements of the system upon which it depends. Nevertheless, our experiments have shown results in favor of our method. In particular, when our collaborative method is used in combination with regular absolute positioning updates, we have shown that the overall localization error does not diverge. Here, the main benefit of our method is that we are able to equip only a few of the team members with absolute positioning capabilities, whilst the other members rely fully on inter-robot positioning plus dead-reckoning.

In order to fully understand the consequences of our approximations beyond the experimental scenarios considered in this work, it would be interesting to perform additional large-scale and long-duration experiments under harsher conditions. The main causes of failure could include: significant odometry (dead-reckoning) errors due to rough terrain or wheel slip, interrupted robot visibility (leading to isolated robots over longer periods of time), and sensor failure. In the same line of thought, such experiments would help determine the limitations of our clustering strategy. It is to be assumed that in dynamic or cluttered environments, more flexible strategies, including adaptive cluster sizes, should be envisioned. Although we do not formalize the necessity for fully connected robot teams, we acknowledge the fact that most of our experiments do exhibit this feature (with dense experimental conditions and few obstacles), and that this contributed to the success of our methods. Finally, we also acknowledge that many of our experiments are baseline tests and do not necessarily serve as representative benchmarks for real-world operation.

## 11.3 Conclusion

This part of the manuscript introduced our collaborative localization strategy and tested it on multi-robot systems. We emphasized that the various approaches (decentralized versus multi-centralized or centralized) each have their advantages as well as disadvantages, and clear trade-offs have to be made. From thereon, our efforts focus on the development of a lightweight, decentralized collaboration strategy, which ultimately combines well with our UWB localization framework introduced in Part II. Our experiments have shown that our method performs successfully under a variety of conditions. In particular, we believe that our work stands out as one of the few that performed extensive real-world experimentation on large-scale multi-robot systems. In conclusion, we believe that the beauty of our algorithm lies in its simplicity, and adaptability to variable resource constraints. In the final part of this manuscript, we will show how our collaborative algorithm is easily combined with an additional sensing modality, namely UWB, which will take the place of the absolute positioning system emulated in this chapter by the overhead camera system.

### Summary

This chapter concludes our work on multi-robot localization. Our approach was strongly motivated by the aim of developing a solution for large-scale, resource-constrained platforms, and distinguishes itself from other works by following a methodology which allows for maximum flexibility within multi-robot systems. Our methods were experimentally validated on real and noisy hardware, and showed that decentralized collaborative localization can be implemented in a light-weight manner to produce good overall results.



# **Ultra-Wideband and Collaboration Part IV**



## 12 Introduction

*Making complementarity work.*

**S**ENSOR fusion lies at the core of state estimation and environmental modeling. This chapter introduces the main notions of sensor fusion and brings our particular case study into a general context. As an introduction to the final part of this manuscript, we will characterize the design of our full system, which combines the two core sensing modalities (relative robot observation data and UWB data). We emphasize the complementarity of these modalities and lay out our motivation for this final part of our study. The final paragraph in this chapter details our experimental setup.

### 12.1 Multi-sensor Data Fusion

The process of multi-sensor data fusion consists of combining the information (observations) from different sensors to produce a superior state estimate or environmental model [113]. This superiority is most often characterized by *completeness* and *robustness*. Due to their dependence on a specific medium (e.g., light, sound, or radio waves) and their unique underlying mechanisms, different sensors are able to contribute complementary information streams to a system. The first goal of multi-sensor fusion is to exploit this complementarity to produce a richer or more complete state estimate or model. A salient example of such a multi-sensor system is the Microsoft Kinect [60], which combines vision data with infrared depth measurements to produce 3-dimensional models of its proximal environment. This particular innovation enabled the development of powerful gesture recognition algorithms, and is still in the midst of generating a flurry of cutting-edge applications. The second goal of multi-sensor fusion is robustness. This concept is demonstrated in research performed by Girod et al. [39] with a study on acoustic range sensing. Their results show that there are situations when it is difficult, or even impossible, for the sensor to identify that it is failing. Their research suggests using orthogonal sensing modalities to alleviate this problem. Indeed, they formalize the following hypothesis:

Some sensory modalities are *orthogonal* to each other, meaning that their sets of failure conditions are largely disjoint. Orthogonal modalities can identify each others' failure modes, and thus improve the data quality through coordination and communication, with significantly less effort relative to the effort required to incrementally improve the sensors on their own.

We shall later see how our research aims to incorporate both completeness and robustness into its framework by considering the fusion of two orthogonal modalities: radio-based UWB and light-based infrared.

Sensor fusion methods can generously be categorized into two main schools of thought: those that are based on probabilistic methods, and those that are not. Literature indicates that probabilistic methods are the prevailing approach, and are even considered the standard choice in robotics research [113]. The basic building block of all probabilistic methods is the Bayes' rule. Importantly, in a multi-sensor scenario, most probabilistic methods using this rule require that the observations are conditionally independent. That is, if  $z$  is a sensor measurement and  $\mathbf{x}$  the state to be estimated, then

$$P(z_1, \dots, z_n | \mathbf{x}) = P(z_1 | \mathbf{x}) \cdot \dots \cdot P(z_n | \mathbf{x}). \quad (12.1)$$

There are numerous ways in which probabilistic sensor fusion can be implemented. Popular frameworks include the Kalman Filter and sequential Monte Carlo methods (particle filters). Besides the choice of framework, there are also a number of high-level design choices. The final outcome is a fusion architecture that defines the routing of the sensor information streams, and their interaction within the various computational stages.

In a review of fusion architecture taxonomies, Durrant-Whyte and Henderson [113] promote three general and independent design dimensions (here, we extend their enumeration with our own elaborations):

- **centralized vs. decentralized**

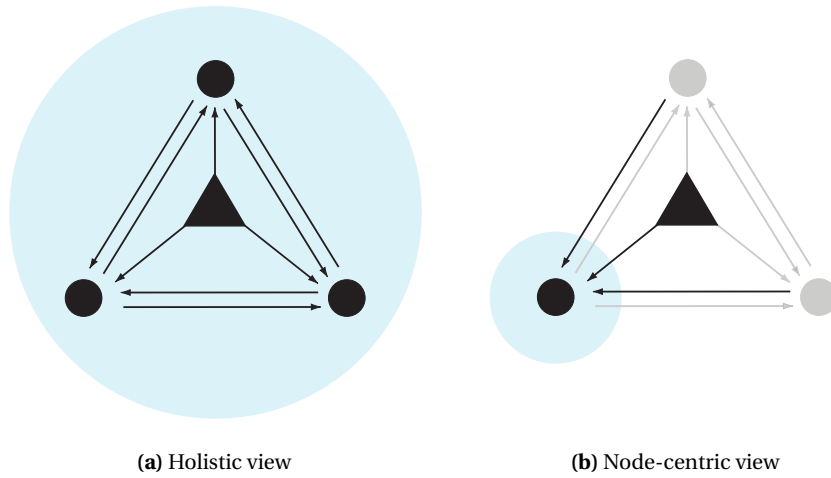
A centralized strategy fuses sensory information within one single machine, or within one process. Centralized fusion is common in single-robot, autonomous navigation problems. Decentralized strategies are common in sensor networks or distributed systems, where incoming data streams are processed by independent sensor nodes, and where communication between the sensor nodes is peer-to-peer.

- **local vs. global interaction**

Systems qualified by local interaction rely on sensor components that use a peer-to-peer (local) manner of sharing information. Local interaction is common in distributed sensor networks. The well known subsumption architecture [17] or Braitenberg vehicle [15] also count as examples of local interaction. Global interaction employs a means of centralizing the point of interaction between the sensor modules. The blackboard system [8] is an example of this type of interaction.

- **heterarchical vs. hierarchical**

In a hierarchical architecture, certain sensor streams may use the output of other streams to influence their own output, or even to inhibit or activate other sensor streams. This type of architecture is often formalized in layers (as seen in the subsumption architecture [17]). Heterarchical architectures treat all sensor inputs on par.



**Figure 12.1:** Two variant ways of viewing multi-agent systems. A holistic view considers the system as one mechanism, and includes all known sensor-actuator instances in the environmental model. A node-centric view considers a single node as the point of departure for all reasonings, and includes an interface for input (and output) sensor streams.

Let us apply this taxonomy to our case study. First, though, let us analyze the design of our current system. As discussed in Chapter 8, our multi-robot system is built upon notions of decentralization and scalability. The system as a whole is modular, with opportunistic peer-to-peer interactions. The system model is node-centric, and approximates the fusion of relative robot observations. This leads us to the following characterization, which draws upon two alternative views of the multi-robot system: **holistic** versus **node-centric** (as illustrated in Figure 12.1). Since our multi-robot system is loosely coupled, and since we model the environment locally (through approximations), our fusion strategy is *centralized* with respect to a node-centric view of the system, but *decentralized* with respect to a holistic view of the system. Let us consider a counter-example. Had we chosen a holistic system design, this would have resulted in a strongly coupled system derived from a holistic model. This, in turn, would lead to a *centralized* (or *multi-centralized*) fusion strategy with respect to a holistic view of the system, and an *outsourced* (or *centralized*) strategy with respect to a node-centric view of the system. This juxtaposition of holistic versus node-centric design is summarized in Table 12.1. We note that the labels of *decentralized* and *centralized* are not conclusive by themselves, and need to be accompanied by a more complete characterization of the system.

The interactions of the sensor components in our system are *local* with respect to the relative positioning observations. Since our UWB component is absolute and communicates centrally with all the nodes in the system, these interactions can be termed *global*. Finally, as we do not impose any layering in our sensor fusion architecture, and both UWB as well as the relative positioning updates are integrated opportunistically, our system is *heterarchical*.

	Holistic design		Node-centric design
Characteristics	strongly coupled interdependent exact		loosely coupled opportunistic approximative
Holistic view	centralized	multi-centralized	decentralized
Node-centric view	outsourced	centralized	centralized

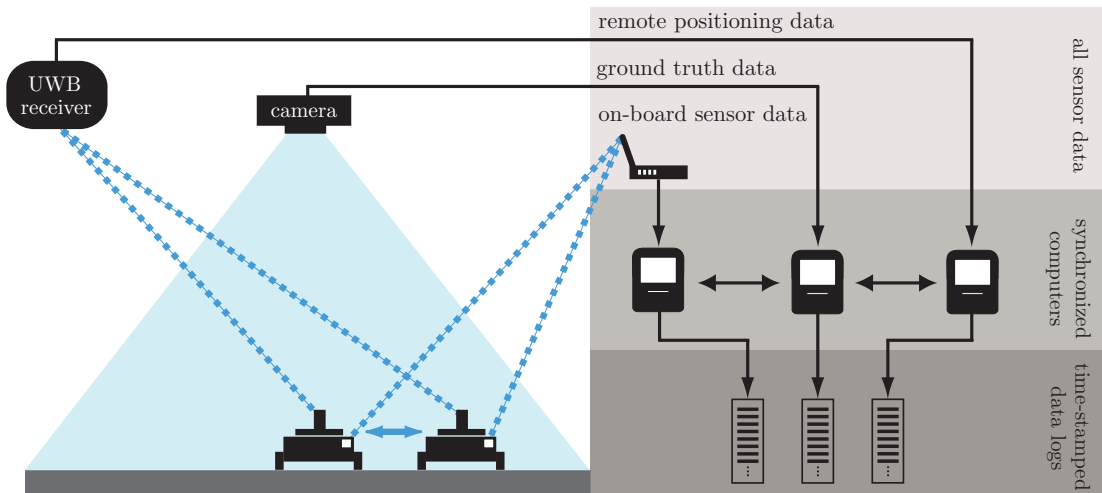
**Table 12.1:** A characterization of two alternative design approaches for multi-robot systems. The table suggests two different view points that lead to different system attributes.

## 12.2 Motivation

The goals of our combined UWB-multi-robot system are threefold. Our first goal consists of increasing the robustness of our position estimates by benefiting from the strengths of our two sensing modalities, and harnessing their complementarities. In Chapter 4, we illustrated the issues that can arise due to obstructions and reflections of UWB signal propagation in NLOS scenarios. We have seen that in harsh scenarios, these issues lead to multimodal error statistics. Our IR-based relative positioning sensors, however, are completely blocked by NLOS and will, thus, never underlie the same failure mode. Indeed, we have shown (in Section 9.2) that the error statistics of the range and bearing module are Gaussian, albeit with a relatively large standard deviation. Additionally, we harness IR-based relative positioning to share and improve robot location estimates. This attribute addresses the spatiality of UWB error behavior: robots in favorable localities (that are associated with mild UWB errors) can positively influence the estimates of other robots. Reciprocally, UWB signals can alleviate errors associated with the IR sensors. We have shown (in Section 8.4.2) that the calibration of IR-based relative positioning modules is a difficult task, and that inaccuracies are hard to avoid. Also, we pointed out that the design of portable IR-based relative positioning modules implies a clear trade-off between cost-efficiency and accuracy/precision. For these reasons, IR-based relative positioning sensors are often noisy. Finally, we recall that the LOS mode of UWB positioning signals has a very narrow measurement distribution (see Section 5.2.3). From these reasonings, we conclude that the signal modalities pair in a complementary manner.

Our second goal is to study and evaluate the benefits that are derived from our combined system. In order to do this, we develop a baseline sensor fusion algorithm, and subsequently perform targeted experiments. These experiments are designed in such a way that the results indicate which strategies are likely to improve overall system performance. In particular, we are interested in understanding the contributions that collaboration (by means of relative positioning) brings to a system equipped with UWB capabilities, and under which circumstances the contributions are maximized.

Finally, our third goal is to validate our two core elements by combining them into a stand-alone product that is able to provide position estimates without additional infrastructure, over longer periods of time. We exploit our controlled setup to monitor the resulting performance, and to compare with previously obtained results.



**Figure 12.2:** The experimental infrastructure is composed of three elements: a vision-based ground truth positioning system, a UWB remote positioning system, and multiple mobile robots carrying UWB emitter tags as well as relative positioning sensors. The overhead camera monitors the uniquely identified mobile robots which drive around in a delimited area. The information from the three subsystems is collected and time-stamped by synchronized computers to produce data logs.

## 12.3 Experimental Setup

The studies in this part focus on the fusion of UWB localization with relative positioning in multi-robot systems. In order to test and validate our methods, we use an experimental setup that includes multiple, networked robots that are endowed with relative observation sensors, as well as UWB emitter tags (the actual platform is depicted in Figure 3.2). Thus, our new experimental system is a combination of our two previous systems (see Figures 4.4 and 8.3), and uses all the components introduced in the respective setups. The UWB tags emit at a frequency of 10 Hz, but practically, UWB measurement updates are received at a frequency of 2.6 Hz. To test the effect of collaboration on the localization performance, we will vary the frequency with which the robots exchange relative detection data. Per default, the robots collaborate at an average frequency of 2 Hz. We also test the efficiency of the collaboration by varying the number of robots that receive UWB positioning updates. The final, combined setup is illustrated in Figure 12.2.

### Summary

This chapter lays the ground for our final series of studies: collaborative multi-robot systems endowed with UWB technology. We introduce our experimental setup, which is essentially the union of our two previously introduced setups, considering both relative positioning as well as UWB localization methods. The concept of sensor fusion is our point of departure, upon which we will build our final algorithm that merges the observation models for our two sensing modalities (UWB and relative range and bearing).





# 13 Algorithms

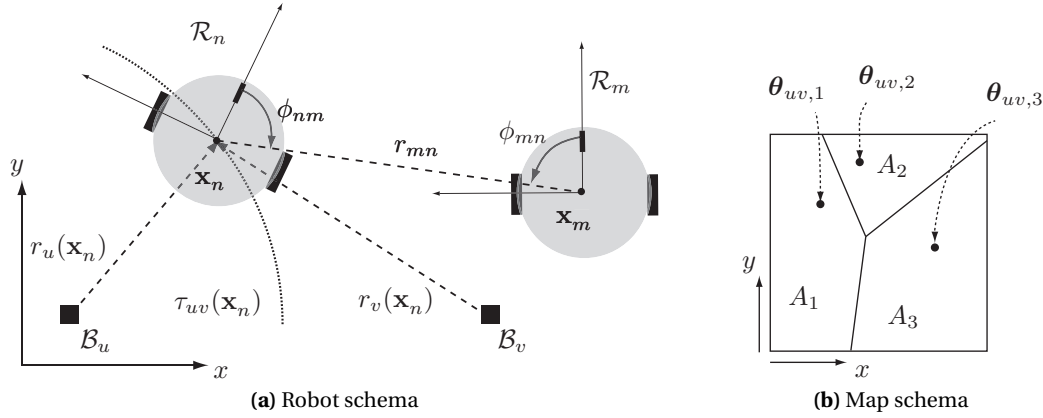
**I**N this chapter, we will evaluate a system that combines relative positioning and UWB localization. We will show how we unify our UWB error model and our robot detection model in our particle filter localization framework. The aim of our experiments is twofold. First, we validate the feasibility of our approach, and show that a localization error in the order of 10 cm is achievable, and maintainable over longer periods of time. Second, we perform a suite of studies to understand the potential contributions of various collaborative strategies to improve UWB localization.

## 13.1 Problem Formulation

Our new problem, illustrated in Figure 13.1, now consists of our two previously introduced problems combined: we consider a system composed of multiple mobile robots that use absolute UWB positioning data as well as relative range and bearing data to localize in an absolute coordinate system. In particular, this part of the manuscript details an algorithm that fuses UWB TDOA measurements and relative positioning measurements with dead-reckoning information. We remind the reader of our two core elements, UWB positioning and relative positioning, and the previously introduced probability densities  $p$  and  $q$  that implement the measurement models for the respective sensor modalities. The final goal of our work is to show how these models are embedded in an efficient localization filter, and to test the resulting performance through extensive experimental work. For convenience, we will revise the formulations of  $p$  and  $q$ .

### UWB measurement model

In order to take account of spatiality, we use the map  $\mathcal{M}_{uv}$  as defined in Section 5.1. Furthermore, we use a function  $m_{uv} : \mathbb{R}^2 \mapsto \Theta$  that maps to any position in two-dimensional space a parameter vector  $\theta_{uv,a}$  in the finite set  $\Theta$ . We model the TDOA error  $\Delta\tau_{uv,n,t}$  for a given base station pair  $\langle \mathcal{B}_u, \mathcal{B}_v \rangle$  with a probability density function  $p$  that covers an area  $A_a$  (such that  $\mathbf{x}_{n,t} \in A_a$ ) and that depends on the parameter vector  $\theta_{uv,a}$ . The reader will recall from



**Figure 13.1:** (a) System of  $N_{\mathcal{R}} = 2$  robots  $\mathcal{R}_n$  and  $\mathcal{R}_m$ , at positions  $\mathbf{x}_n$  and  $\mathbf{x}_m$ , respectively. The figure depicts the true ranges  $r_u(\mathbf{x}_n)$  and  $r_v(\mathbf{x}_n)$  of robot  $\mathcal{R}_n$  to the respective base stations, as well as a segment of the hyperbola resulting from the range-difference  $\tau_{uv,n}$ . The figure also depicts the relative robot range  $r_{mn} = r_{nm}$ , and the relative bearing values  $\phi_{mn}$  and  $\phi_{nm}$ . (b) Distinct UWB error models  $p(\Delta\tau_{uv,n,t}; \theta_{uv,a})$  are mapped to individual areas  $A_a$ .

Equation (5.3) that our UWB error model is formulated as

$$p(\Delta\tau_{uv,n,t}; \theta_{uv,a}).$$

### Relative range and bearing model

The robot detection model consists of integrating the belief (which is represented by particles) of the detecting robot into that of the detected robot. The robot detection model  $q$  describes the probability that robot  $\mathcal{R}_m$  detects robot  $\mathcal{R}_n$  at position  $\mathbf{x}_{n,t}$ , by considering all the range and bearing errors, for all particles belonging to robot  $\mathcal{R}_m$ . This model depends upon a parameter  $\xi$  that describes the range and bearing covariance matrix, and in our case, is calibrated for all robots simultaneously. The reader will recall from Equation (9.6) that our robot detection model is formulated as

$$q(\Delta r_{mn,t}, \Delta \phi_{mn,t}; \xi).$$

## 13.2 Fusion Algorithm

Algorithm 9 shows the complete localization algorithm, embedding all previously developed models and algorithms. Line 3 shows the application of the motion model, where  $u_{n,t}$  represents dead-reckoning information. Line 4 shows the application of the measurement model where  $T_{n,t}$  represents the TDOA data and  $\mathcal{M}$  is the map. Line 5 shows the application of the robot detection model, where  $D_{n,t} = \{d_{mn,t} | \mathcal{R}_m \in \mathcal{N}_{n,t}\}$  is the set of all communication messages received by robot  $\mathcal{R}_n$ . The frequency of the run-loop is dictated by the odometry updates, and UWB measurements and relative observations are fused when available. Finally, in addition to using the robot detection model for updating the belief representation  $\text{Bel}(\mathbf{x}_{n,t})$ , our approach utilizes the reciprocal sampling method, shown in line 13, with a reciprocal

sampling proportion  $\alpha \leq 1$ .

---

**Algorithm 9** MultiRobot\_UWB\_MCL( $X_{n,t-1}, u_{n,t}, T_{n,t}, D_{n,t}$ )

---

```

1:  $\bar{X}_{n,t} = X_{n,t} = \emptyset$ 
2: for  $i = 1$  to  $M$  do
3:    $\mathbf{x}_{n,t}^{[i]} \leftarrow \text{Motion\_Model}(u_{n,t}, \mathbf{x}_{n,t-1}^{[i]})$ 
4:    $\bar{w}_{n,t}^{[i]} \leftarrow \text{Measurement\_Model\_Map}(T_{n,t}, \mathcal{M}, \mathbf{x}_{n,t}^{[i]}, w_{n,t-1}^{[i]})$ 
5:    $w_{n,t}^{[i]} \leftarrow \text{Detection\_Model}(D_{n,t}, \mathbf{x}_{n,t}^{[i]}, \bar{w}_{n,t}^{[i]})$ 
6:    $\bar{X}_{n,t} \leftarrow \bar{X}_{n,t} + \langle \mathbf{x}_{n,t}^{[i]}, w_{n,t}^{[i]} \rangle$ 
7: end for
8: for  $i = 1$  to  $M$  do
9:    $r \sim \text{Uniform}(0, 1)$ 
10:  if  $r \leq (1 - \alpha)$  then
11:     $\mathbf{x}_{n,t}^{[i]} \leftarrow \text{Sampling}(\bar{X}_{n,t})$ 
12:  else
13:     $\mathbf{x}_{n,t}^{[i]} \leftarrow \text{Reciprocal\_Sampling}(D_{n,t}, \bar{X}_{n,t})$ 
14:  end if
15:   $X_{n,t} \leftarrow X_{n,t} + \langle \mathbf{x}_{n,t}^{[i]}, \frac{1}{M} \rangle$ 
16: end for
17: return  $X_{n,t}$ 

```

---

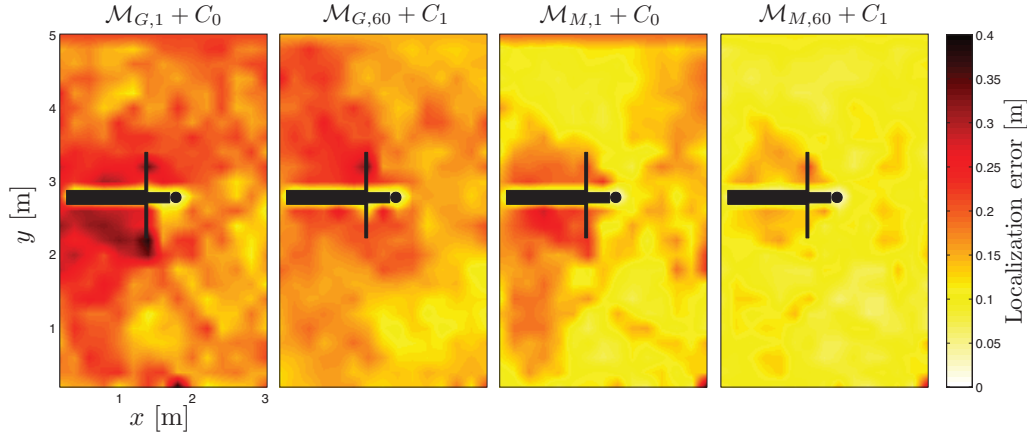
### 13.3 Performance

In our following evaluations, we will consider the experimental configurations **A** and **B** introduced in Part II. For our UWB localization, we will again test three variant models (multimodal, Gaussian, and histogram). We will reuse the notation introduced in Section 6.4 to label our three map types, i.e.,  $\mathcal{M}_M$  (multimodal),  $\mathcal{M}_G$  (Gaussian), and  $\mathcal{M}_H$  (histogram). Also, we will reuse the same four map granularities: (a)  $N_A = 1$  with one  $3 \times 5$  m<sup>2</sup> cell, (b)  $N_A = 15$  with  $1 \times 1$  m<sup>2</sup> cells, (c)  $N_A = 60$  with  $0.5 \times 0.5$  m<sup>2</sup> cells, and (d)  $N_A = 375$  with  $0.2 \times 0.2$  m<sup>2</sup> cells. As before, these two map parameters are combined, with resulting maps referred to as  $\mathcal{M}_{model, N_a}$  (e.g.,  $\mathcal{M}_{M, 375}$  corresponds to a multimodal map with 375 cells).

In addition, we will test the effect of collaboration on the overall localization performance. To this means, we will vary the frequency with which the robots exchange detection data. Per default, the robots collaborate at an average frequency of 2 Hz. We remove detection messages from our data log a posteriori, producing four additional data sets with collaboration rates at 1 Hz, 0.5 Hz, 0.1 Hz, and 0 Hz (i.e., no collaboration). In the following, these schemes are referred to as  $C_2$ ,  $C_1$ ,  $C_{0.5}$ ,  $C_{0.1}$ , and  $C_0$ , respectively (as done in Section 10.4).

#### 13.3.1 Overall Localization Error

Our localization algorithm is evaluated repeatedly (over 10 iterations) on a data set for each of the two configurations **A** and **B**, comprising a 40 minute experiment involving ten robots, initially randomly distributed over the arena. Each robot runs Algorithm 9 with 50 particles, and a reciprocal proportion  $\alpha = 0.02$ . We discuss the localization performance in terms of the



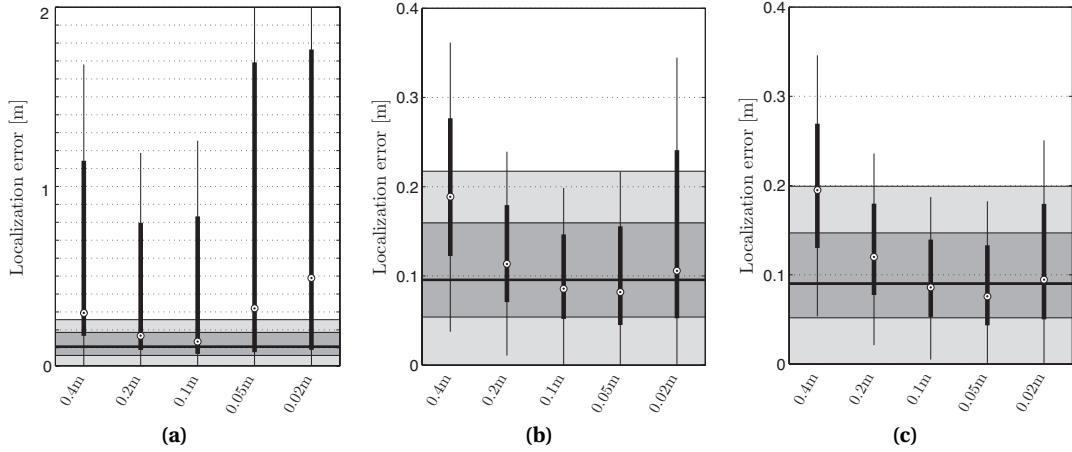
**Figure 13.2:** The graphs show the average localization error as a function of space, over a 40 minute experiment with ten robots. Four different maps ( $\mathcal{M}_{G,1}$ ,  $\mathcal{M}_{G,60}$ ,  $\mathcal{M}_{M,1}$ ,  $\mathcal{M}_{M,60}$ ) are tested with two different collaboration frequencies ( $C_0$ ,  $C_1$ ).

positioning error (distance to ground truth position) of the center of mass of the particles in a robot's belief, for all ten robots used in the experiment (as formulated in Section 6.4, Equation (6.19)).

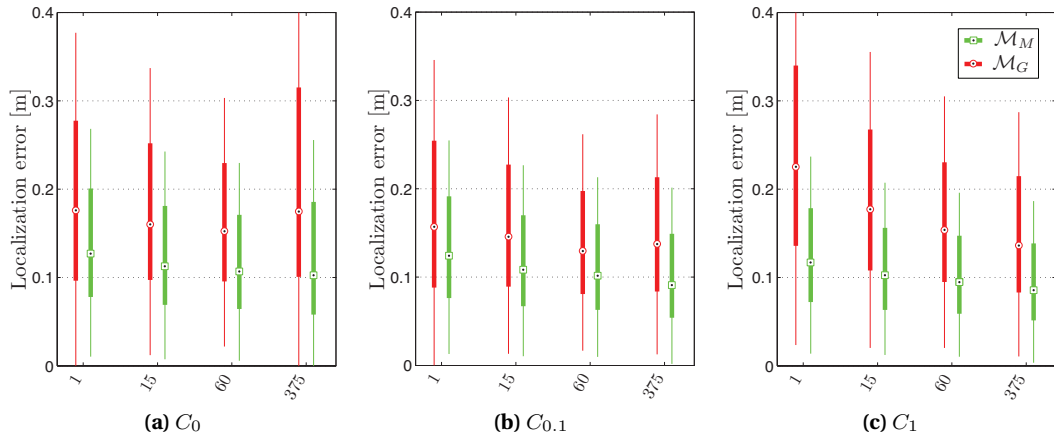
Figure 13.2 shows the localization performance as a function of space. We note the irregularity of the error distribution: higher errors tend to be in the vicinity of the obstacle. The plots indicate that, for the same resolution and same collaboration rate, our multimodal model ( $\mathcal{M}_{M,.}$ ) is better than the Gaussian ( $\mathcal{M}_{G,.}$ ) model. They also indicate that a high resolution map ( $\mathcal{M}_{.,60}$ ) combined with collaboration is better than a low resolution map ( $\mathcal{M}_{.,1}$ ) without collaboration. Moreover, the results indicate that the improvements help mitigate errors in areas prone to NLOS (confer Figure 6.8): the right-most panel ( $\mathcal{M}_{M,60} + C_1$ ) shows a quasi constant distribution of localization errors in the range of 5 cm to 10 cm. Finally, for comparison, we remind the reader that the error of the maximum likelihood trilateration estimates (computed with TDOA measurements and evaluated over the whole experimental run) amounts to 0.18 m (median) and 0.48 m (mean) for configuration **A**, and 0.2 m (median) and 0.45 m (mean) for configuration **B**. The following experiments will isolate the variables (collaboration and map resolution), to better understand their individual contributions.

### 13.3.2 Evaluation of Collaboration

Section 10.4 showed the relevance of our multimodal model  $\mathcal{M}_{M,.}$  by comparing it to alternative models, a histogram model  $\mathcal{M}_{H,.}$  and a Gaussian model  $\mathcal{M}_{G,.}$  We will extend this analysis by including collaboration into the design of our experiments. Figure 13.3 shows a comparison of the localization performance obtained for  $\mathcal{M}_{H,375}$ , for five different bin sizes (0.4 m, 0.2 m, 0.1 m, 0.05 m, 0.02 m), and  $\mathcal{M}_{M,375}$ , and is analogous to the Figure 6.10 of Part II (which was performed without collaboration). The plots show results for configuration **A** and **B** combined, with an increasing collaboration frequency from left to right. A significant difference in performance is seen in Figure 13.3 for the histogram model, where the median of the non-collaborative scheme ( $C_0$ ) in plot (a) ranges in between 0.13 m to 0.48 m, compared to the collaborative schemes in figures (b) and (c) ( $C_{0.1}$  and  $C_1$ ), with a range in between 0.075 m



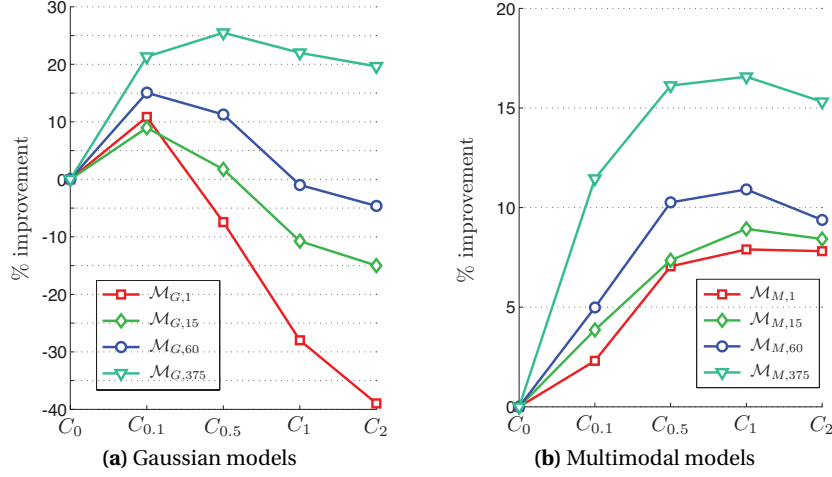
**Figure 13.3:** Localization performance using  $\mathcal{M}_{H,375}$ , with varying bin-sizes (5 boxplots along the x-axis), as well as  $\mathcal{M}_{M,375}$  (1 boxplot spanning the x-axis, shaded in grey). The boxplots show the median, 25th and 75th percentile of all localization errors for all robots throughout the experiment. The plots show results for configuration A and B combined. The collaboration frequency increases from left to right. (a)  $C_0$ , (b)  $C_{0.1}$ , (c)  $C_1$ .



**Figure 13.4:** Localization performance using  $\mathcal{M}_M$  and  $\mathcal{M}_G$  with four grid resolutions  $N_a$  (along the x-axis). The boxplots show the median, 25th and 75th percentile of all localization errors for all robots throughout the experiment. The plots show results for configuration A and B combined. The collaboration frequency increases from left to right. (a)  $C_0$ , (b)  $C_{0.1}$ , (c)  $C_1$ .

and 0.19 m. Indeed, collaboration plays an important role for the histogram models—even for the optimal bin size of 0.1 m, the non-collaborative variant produces a large number of outliers. With collaboration, the histogram models produce a performance comparable to that of the multimodal models.

Figure 13.4 shows a comparison of the localization performance obtained for  $\mathcal{M}_M$  and  $\mathcal{M}_G$ , for all four grid resolutions,  $N_a = \{1, 15, 60, 375\}$ , and is analogous to Figure 6.11 of Part II. Again, the plots show results for configuration A and B combined, with an increasing collaboration frequency from left to right. The results qualitatively indicate that, when employing the multimodal error maps  $\mathcal{M}_M$ , a performance increase is obtained for increasing col-

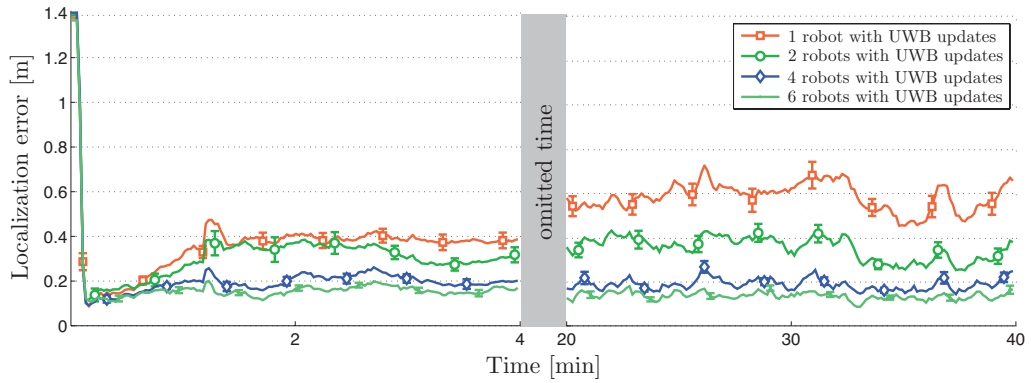


**Figure 13.5:** Improvement in percent of the localization accuracy for *collaborative* versions of the algorithm over the *non-collaborative* version ( $C_0$ ). The data points were calculated by considering the respective median error values. The collaboration frequencies vary from 0.1 Hz to 2 Hz:  $C_{0.1}$ ,  $C_{0.5}$ ,  $C_1$ ,  $C_2$ . We test (a) the Gaussian model with maps  $\mathcal{M}_{G,\cdot}$  and (b) multimodal models with maps  $\mathcal{M}_{M,\cdot}$ .

laboration frequencies. This result is represented quantitatively in Figure 13.5: we use the non-collaborative results as a reference and calculate the relative (to the collaborative variants) performance improvement in percent, and we analyze the performances for an increasing collaboration rate, for all map granularities. Let us first consider Figure 13.5a. It becomes clear that for the Gaussian maps  $\mathcal{M}_{G,\cdot}$ , a consistent performance increase is only obtained for the lowest collaboration frequency (0.1 Hz), over all map granularities, and only the highest map granularity  $\mathcal{M}_{G,375}$  produces a consistent improvement for all collaboration frequencies. In the worst case, the map  $\mathcal{M}_{G,1}$  produces a 40% performance decline for a collaboration frequency of 2 Hz. Let us now consider Figure 13.5b. A performance improvement is achieved for all map granularities and collaboration frequencies, with a maximum performance improvement of 16.5% achieved for  $\mathcal{M}_{M,375}$  at 1 Hz. The panel also shows that an increased map granularity consistently improves the performance. However, we also note that the maximum performance improvement is not achieved for the highest collaboration frequency. This is likely due to the reciprocal sampling method, which, for an increased frequency, also samples an increased number of reciprocal particles over time. The uncertainty in the particle distribution is thereby increased with respect to the current, optimally tuned uncertainty, and thus the potential decline in performance. We note that this effect can be mitigated by calibrating the reciprocal sampling proportion  $\alpha$  as a function of the collaboration rate.

### 13.3.3 Evaluation of Heterogeneity

Analogous to Figure 10.8 in Chapter 10, Figure 13.6 shows the results over a 40 minute data set performed for experimental configuration **A**. A variable number of robots receive UWB updates. The experimental data is evaluated over 50 iterations, where at each iteration the set of robots that receive the UWB updates is randomly allocated, with the remaining robots relying on our relative positioning method for localization. We evaluate the localization error for all robots



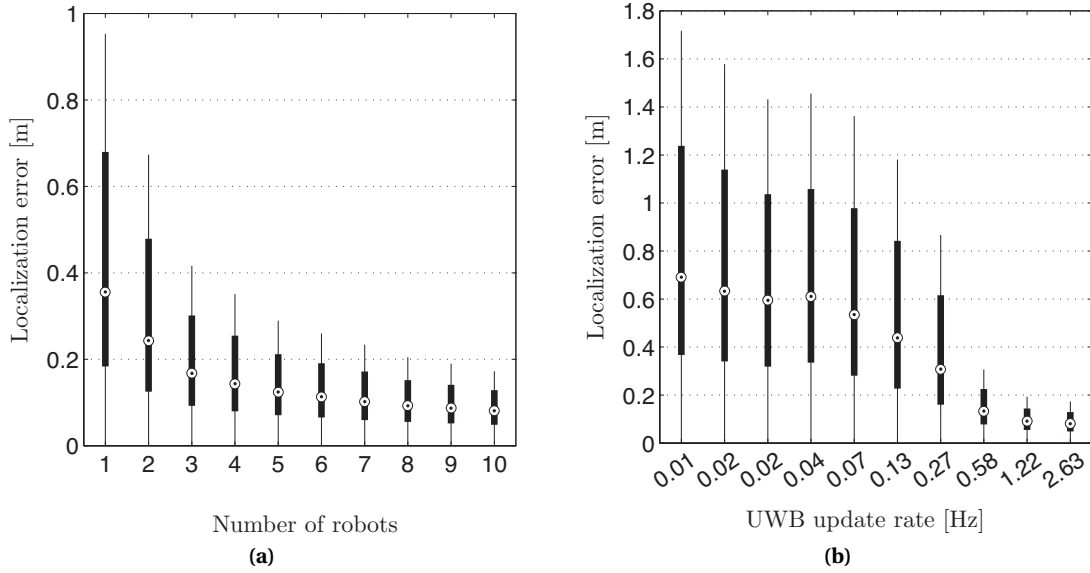
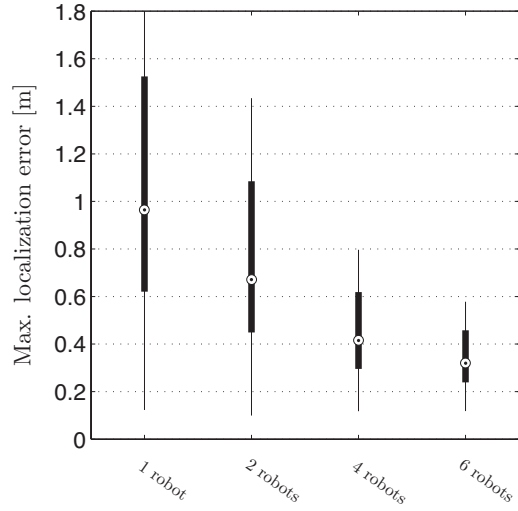
**Figure 13.6:** Average localization error (for all robots) obtained over 50 evaluations of a data set for an experiment of 40 min duration, performed on 10 Khepera III robots with 50 particles each, and a reciprocal sampling proportion of  $\alpha = 0.02$ . We vary the number of robots that receive UWB positioning updates. The error bars show a 95% confidence interval.

in the system, and for better clarity, omit 16 minutes worth of data points in the plots. The results show that by providing at least one robot in the team with absolute position updates, the localization error is bounded and constant over time. Also, we note that by increasing the number of continuously localized robots, the error decreases. This result also indicates the utility of a potentially heterogeneous robot team, where some robots are equipped with UWB, and others rely purely on collaborative methods (on top of dead-reckoning). The resulting localization error will then depend on various factors that affect the system's connectivity, such as the spread of the robots in space and the frequency of the interactions between the heterogeneous robot types.

In order to assess the worst performance, we compute the maximum error value (over all robots) for every time step, throughout the whole run. These data points are summarized in the form boxplots in Figure 13.7. The results indicate that by increasing the number of robots that receive UWB updates, the maximum localization error decreases consistently, as we are able to positively affect the performance of all other robots in the team. This result is analogous to the results shown in Figure 10.10, in Part III. We note that by using ground truth position updates instead of UWB position updates, the maximum error is reduced by 18% in average.

In order to assess the form of collaboration, we perform a set of two experiments. In the first experiment, we vary the number of robots that receive UWB updates from 1 to 10 robots. In the second experiment, we vary the UWB update frequency (all robots receive updates), in a log-scale interval {0.014 Hz, 0.018 Hz, 0.024 Hz, 0.038 Hz, 0.069 Hz, 0.27 Hz, 0.58 Hz, 1.22 Hz, 2.63 Hz}. Figure 13.8 shows the results for both sets. In Figure 13.8a, we see that the localization error decreases exponentially for an increasing number of robots with UWB. This indicates that little performance improvement is to be gained above a certain threshold number of robots with UWB updates. Figure 13.8b shows that the localization error decreases for an increasing rate of updates, and that the strongest performance gain is achieved in between 0.27 Hz and 0.58 Hz. We compare the results of Figures 13.8a and 13.8b in Figure 13.9 by computing the total number of UWB updates received over all evaluations, and by using this value as the reference for comparison. Interestingly, we note that, for identical effective update rates, the 10-robot variant (with variable UWB rates) performs better (in the interval [0.5 Hz 2.6 Hz]). This result highlights the value of distributing the capability of measuring the absolute position

**Figure 13.7:** Maximum localization error (for all robots, at any given time) obtained over 50 evaluations of a data set for an experiment of approximately 40 min duration performed on 10 Khepera III robots with 50 particles each, a reciprocal sampling proportion of  $\alpha = 0.02$ . We vary the number of robots that receive UWB positioning updates. The boxplots show the median, 25th, and 75th percentile of all data.



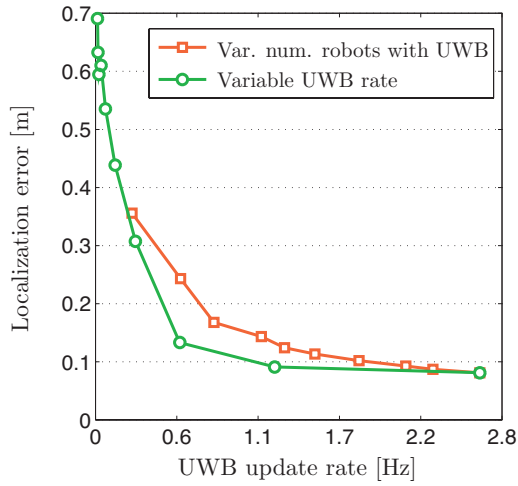
**Figure 13.8:** Localization error (for all robots, throughout the whole run) obtained over 20 evaluations of a data set for an experiment of approximately 40 min duration performed on 10 Khepera III robots with 50 particles each, a reciprocal sampling proportion of  $\alpha = 0.02$ . We vary (a) the number of robots that receive UWB positioning updates, and (b) the UWB update frequency. The boxplots show the median, 25th, and 75th percentile of all data.

within the team. For a cost defined by the number of UWB measurements received over all robots, a homogenous team fares better than a heterogeneous team.

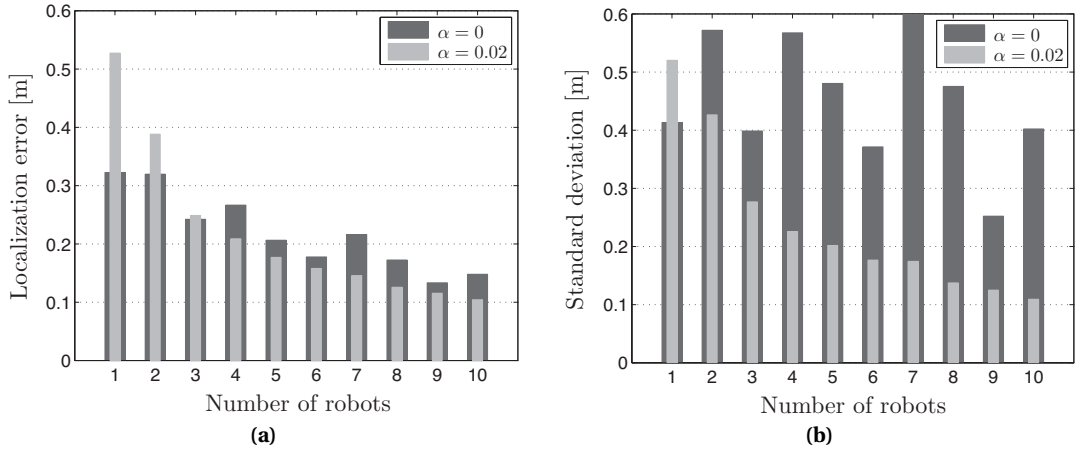
### 13.3.4 Evaluation of Reciprocal Sampling

Finally, we perform an assessment of our reciprocal sampling strategy. To this means, we evaluate our localization algorithm, using 50 particles, running on 10 robots, on our data set of 40 min duration—once with reciprocal sampling ( $\alpha = 0.02$ ), and once without reciprocal





**Figure 13.9:** Summary of the median localization errors reported in Figure 13.8, with results for (i) a variable number of robots that receive UWB updates (1-10 robots), and (ii) a variable UWB update rate (0.014 Hz-0.263 Hz), and all 10 robots receive the updates. To form a reference, we computed the total number of UWB updates received for all runs (and all robots) and deduced the average UWB update frequencies.



**Figure 13.10:** Localization error (for all robots, throughout the whole run) obtained over 50 evaluations of a data set for an experiment of 40 min duration performed on 10 Khepera III robots with 50 particles each. We test our algorithm with and without reciprocal sampling, i.e., once with a proportion of  $\alpha = 0.02$ , and once with  $\alpha = 0$ . We vary the number of robots that receive UWB positioning updates. (a) Mean localization error. (b) Standard deviation of localization errors.

sampling ( $\alpha = 0$ ). We vary the number of robots that receive UWB updates. Figure 13.10 reports the results. In Figure 13.10a we see that for very few robots (1 or 2) receiving UWB updates, the standard sampling routine performs better, since the probability that reciprocal particles are sampled from badly localized robots is greater. For 4 or more robots receiving UWB updates, the reciprocal sampling strategy improves over the standard sampling strategy. A clear performance difference is noticeable in Figure 13.10b, which reports the standard deviation of all localization errors. For nine out of ten results, the reciprocal sampling strategy produces a smaller standard deviation. Moreover, the standard deviation decreases asymptotically for an increasing number of robots that receive UWB updates. This indicates that the reciprocal sampling strategy provides higher robustness.

### Summary

This chapter shows our final evaluations on a stand-alone positioning system consisting of absolute UWB localization and relative robot positioning. We presented a baseline fusion algorithm that merges our measurement model for UWB and relative robot detections. Evaluations on 40 minute long data sets were able to validate our method and showed that even by employing only one of ten robots with UWB, the system's overall localization error remains bounded over time. In the best cases, we report a median localization error below 10 cm. We also showed the importance of capturing the UWB errors with a faithful model when used in combination with inter-robot collaboration. Importantly, we showed that an optimal collaboration rate exists, and that it can be practically implemented by means of our tunable collaboration scheme. Furthermore, our experiments suggested that a robot team, whose nodes are endowed with the same localization strategy, is more successful than a heterogenous robot team, for the same number of UWB positioning updates, and, thus, highlighted the value of distributing the capability of measuring the absolute position throughout the whole robot team. Finally, the application of our reciprocal sampling routine showed a clear gain in robustness with respect to a standard sampling routine. In conclusion, the mutual contributions of UWB and collaboration were emphasized throughout all experimental results, confirming the validity of our collaborative strategies, and confirming our hypothesis of complementarity.

## 14 Conclusion

**W**E conclude our work on multi-robot UWB localization by discussing our approach and relating to similar works. We highlight that the fusion of UWB with exteroceptive sensors has hardly been considered so far. We nevertheless relate to other applicable approaches, such as multi-robot SLAM and collaborative estimation. Our discussion addresses the impact of collaboration on UWB localization, and includes insights on team heterogeneity and sensor configurations.

### 14.1 Related Work

UWB is a relatively novel addition to the suite of sensors considered within robotics research. Our summary in Section 7.1 enumerates a number of studies that consider stand-alone UWB systems, or UWB fused with odometry. However, to the best of our knowledge, no significant studies have been performed on the fusion of UWB with on-board exteroceptive sensors, in the case of single-robot systems, nor any on-board relative positioning sensors, in the case of multi-robot systems.

Our problem of augmenting global (absolute) positioning information with collaborative methods relates loosely to the more general, multi-robot SLAM problem. This subject of research considers the problem of aligning multiple maps, provided by multiple robots that measure their relative poses, to ultimately produce one joint map. Numerous solutions have been proposed, using frameworks such as the Extended Information Filters [121] and Kalman Filters [131]. Howard et al. [52] show a solution to this problem that utilizes a particle filter framework. Such SLAM methods naturally extend our work in that collaboration and joint construction of our UWB error maps could lead to further insights on our topic.

The work by Charrow et al. [22] also bears similarities to our work. In this study, the authors consider a multi-robot system that collaboratively estimates the location of a radio-source, and actively (through motion control) contributes to improving the quality of this estimate. In particular, the authors develop a range model that addresses NLOS scenarios for RF-based sensors. This model based on biased Gaussians, with increasing variances for larger distances. Indeed, by introducing control strategies into our work, the robots could potentially target destinations in space that are associated with low UWB measurement uncertainty (i.e., LOS).

This, in combination with targeted collaboration strategies, could potentially further reduce the overall localization error, at the price of, however, more constraints on the actual task to be carried out.

### 14.2 Discussion

This part of the manuscript discussed the effect of collaboration in the form of relative positioning on the overall localization accuracy. We have shown that the collaboration rate has an important impact on the localization accuracy, and that determining the optimal rate is relevant. How an optimal collaboration rate is to be defined remains to be explored. It is to be assumed, however, that our tunable scheme provides a convenient framework, and that standard optimization methods are easily applied. We have seen that good performance can still be achieved by considering heterogeneous robot teams that are only partially equipped with UWB sensors. We might, at the one extreme, consider a single, highly mobile and agile robot that is equipped with UWB, and that by means of relative positioning can update the position estimates of all its other teammates. On the other extreme, we might consider a fully equipped robot team, but that uses very low UWB update rates. Our results have shown that, under our given experimental conditions, the homogenous system produces better results. However, by varying conditions such as the connectivity in between the two robot types and also by increasing the degree of heterogeneity (to include other factors such as mobility), other conclusions may result.

### 14.3 Conclusion

In our last module of research, we investigated the intertwining of local relative positioning measurements with global UWB measurements. It has served two purposes. The first purpose was the validation of Parts II and III. Our framework has shown to support the coexistence of two complementary sensory modalities, while addressing non-Gaussian and non-unimodal distribution characteristics. We have shown that with simple embedded platforms, we were capable of providing sub-decimeter accuracy (with the help of a priori UWB error maps). The second purpose of this part was to provide a set of insights into the mutual contributions of UWB positioning and collaboration by means of relative positioning. To the best of our knowledge, this manuscript constitutes pioneering work in this domain. Our conclusions highlight the complementary characteristics of the sensing modalities. Meanwhile, they also highlight the feasibility of heterogeneous systems (built of robots with or without UWB) that are able to maintain the full localization functionality, and, if carefully designed, at an insignificant cost of performance.

#### Summary

We highlight that the fusion of UWB with exteroceptive sensors has hardly been considered so far, and that our work represents pioneering work in this domain. Several insights indicate that collaboration—even if through noisy sensors—is a useful tool to further improve global (absolute) positioning estimates. We have shown that our particular collaborative framework can provide the means to reach a minimum localization error, given that the collaborative design parameters are optimally tuned (i.e., collaboration rate and reciprocal sampling rate). The simplicity of our approach promises easy combination with additional strategies, as reviewed in related work.



## **Conclusion Part V**





## 15 Conclusion and Outlook

**I**N its fullest form, this dissertation presents a framework that unifies absolute UWB positioning measurements with local relative range and bearing measurements. This combination was based upon the hypothesis that the complementarity of the measurement types contributes to the robustness of the system, and, thus, improves its overall performance. Testing this hypothesis constituted one of the goals of this work—to this means, preceding our final experiments and analyses, we were required to develop and refine the individual components of our envisioned system. This roadmap led us to design a modular, systematic experimental setup, that was capable of quantifying the performance of our system at all stages of our work. Partially due to the miniature size of our experimental embedded platform, and partially due to our own conviction, our aims imposed a strong emphasis on the development of efficient and low-cost methods. This setting dictated a set of guidelines that helped us produce a consistent set of results. Leading up to our final stage of work, we invested our research into two self-contained modules: UWB localization with a single robot, and collaborative localization in multi-robot systems. This manuscript dedicated a part to each of these modules, and demonstrated their respective contributions.

Our work on single-robot localization with UWB focused on the development of models and algorithms for real-time localization routines. Our contributions are summarized as follows:

- We addressed the peculiarities of the propagation of UWB ranging signals with a probabilistic measurement model that can capture the spatiality as well as the statistical multimodality of the error behavior. We developed a general TOA measurement model that is a mixture of Gaussian and log-normal distributions, and extended this model to include TDOA measurements. We developed an approximation of the TDOA model to produce a compact closed-form equation. We quantified the similarity of the approximated model to the non-approximated model, and showed how the approximated model successfully captures the four essential error modes for an UWB base station pair. To the best of our knowledge, our model is the first UWB TDOA measurement model for mobile robot localization.
- The closed-form model equation enabled the development of an efficient estimation

algorithm based on EM to determine the values of the model parameters. We showed how the algorithm is applied in a batch mode as well as in an online mode, and how it tends towards the CRLB for an increasing number of data samples. We evaluated the performance of the estimation algorithm quantitatively to show that it performs very well in comparison with an alternative, standard optimization method.

- Our approach was experimentally validated in a room-sized setup including obstacles that induce strong NLOS signal propagation. Final results showed a localization performance with median errors in the order of 10 cm, significantly improving over the standard, maximum likelihood position estimates.

Our work on multi-robot localization focused on the development of models and algorithms for collaborative localization. Our contributions are summarized as follows:

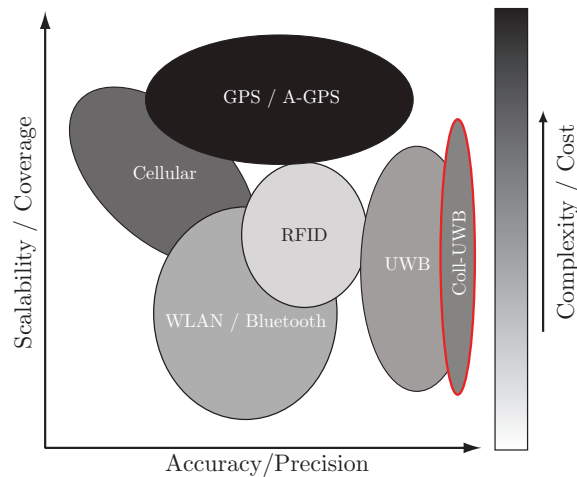
- We developed a collaborative localization strategy based on a Monte Carlo method that distinguishes itself from prior work by emphasizing cost-efficiency, full decentralization, and scalability. Our robot detection model builds upon a particle-based belief representation, and is able to integrate noisy relative range and bearing measurements.
- We developed a reciprocal sampling routine, and showed how it is able to successfully exploit collaboration in the robot team by drawing new particles in regions of the state space that are likely at correct positions, according to neighboring robots' beliefs. Our results showed that the routine facilitates the convergence of position estimates and increases the robustness of the localization algorithm.
- We developed a particle clustering method that enabled a reduction of the overall computational cost of the collaborative localization algorithm from a complexity of  $O(M^2)$  to a complexity of  $O(MK)$ , for  $M$  particles and  $K$  clusters. To this means, we introduced a novel, summarized formulation of relative observations and integrated it directly into our original detection model.
- We validated our algorithm with a team of four mobile robots, in real-time and running on-board. We also showed its successful implementation on a team of 10 robots by evaluating a data set of over 40 minutes experimental run-time. In particular, our results showed that if only one out of the ten robots is localized over that time-span, with all other robots relying on relative positioning, the overall localization error remains bounded.

We worked on our two research modules while bearing in mind that our final product would consist of their combination. The result of this effort was a unified formulation of the various components, and a framework which successfully bound the models and algorithms into one final localization filter. By combining UWB localization with collaborative localization, we presented a novel setup and performed pioneering experimental work in order to study the combined usage of the two sensor modalities and their associated methods. To the best of our knowledge, our setup is the first to consider the fusion of UWB positioning data with exteroceptive sensing data. Moreover, our experiments with teams of up to 10 mobile robots, equipped with both UWB emitters as well as relative positioning modules, are unprecedented in this context. This experimental work allowed us to draw a number of conclusions:

- Collaboration, even if through noisy relative positioning sensors, is a useful tool to reduce localization errors in an UWB positioning system.
- An optimal collaboration rate exists. By exploiting the flexibility of our framework, this optimum can be easily accounted for.
- The underlying UWB model plays an important role: a faithful model positively affects the localization performance, at any collaboration rate; a poorly fitted model penalizes localization performance when used with increasing collaboration rates.
- Robot teams, whose nodes are endowed with the same localization strategy, are more efficient than heterogeneous robot teams, for the same effective UWB positioning update rates. This indicates the value of distributing the capability of receiving absolute positioning updates throughout the team.

In the introduction to this work, we presented a comprehensive overview of current positioning technologies. Figure 1.3 ranked the various technologies according to our custom designed evaluation criteria: accuracy/precision, scalability/coverage, and complexity/cost. In conclusion to this work, we now add our hybrid system to this overview (to be seen in Figure 15.1, outlined in red). We consider our system favorably placed at the top end of the accuracy/precision axis, since significant performance improvements were achieved by adding collaborative localization methods. Simultaneously, we believe that the addition of scalable collaborative methods extends the spatial reach of the system (as suggested by the possibility of heterogeneous systems). Finally, we believe that, although the combined approach may enable an overall cost reduction, as (potentially cheap and noisy) relative positioning sensors compensate for more expensive, incremental improvements on a system based purely on UWB, the added cost of maintaining the relative positioning sensors increases the overall complexity.

This work constitutes the first steps towards studying the effects of collaboration on UWB localization. The consistency and continuity of the presented experimental work is an instrumental



**Figure 15.1:** Overview of positioning technologies, as a function of our evaluation criteria: accuracy/precision, scalability/coverage, and complexity/cost. An ideal system would be located at the top right corner of the chart and colored in white. Our proposed UWB-multi-robot localization system is outlined in red.

part of our methodology. This being stated, the establishment of a controlled and systematic setup comes with the sacrifice of a more large-scale, dynamic experimental setup. Importantly, although we believe our online formalisms can cope with dynamic environments, such settings still need to be rigorously tested. Also, in order to reduce the cost of building a priori error maps, one should explore the simultaneous localization and mapping of UWB measurement models, beyond our presented work. For this, we believe that a number of theories from classical SLAM literature are applicable to our given problem statement. Finally, the implications of adding a third dimension to the state space should be considered, and tested on similarly designed, rigorous robotic setups using flying vehicles. Indeed, an augmented setup would ultimately enable tests that approximate actual, real-life conditions—the success of such tests would close the final gap to commercial, real-world integration of our proposed methods.

We hope that our discoveries in UWB localization will affect research communities on both sides of the divide: positioning systems researchers as well as roboticists. In particular, we believe that our TDOA measurement model will complement recent advancements in path detection algorithms for UWB ranging. A potential coupling of the two approaches may lead to even further performance improvements. Also, we hope that our insights on collaboration will incite positioning systems researchers to consider collaborative paradigms as an integral element of their systems' designs. We believe that the integration of cheap relative positioning hardware is relevant and feasible, and represents a cost-efficient way of improving absolute positioning systems. Finally, we believe that our computationally lightweight solutions will gain in importance, particularly with the advancement of embedded, miniaturized technologies—we hope that our highly scalable approach will prove effective with the ultimate realization of pervasive computing in our every-day lives.

# A Analysis of Reciprocal Sampling

The performance of the reciprocal sampling algorithm depends on the accuracy of the detection sensors. Simultaneously, the reciprocal sampling proportion  $\alpha$  needs to be calibrated to enable fast convergence to a low localization error. For the purpose of this analysis, we resort to a minimal scenario consisting of two collaborative robots moving randomly in bounded space. The goal of the exercise is to discuss the localization performance of the first robot  $\mathcal{R}_n$ , which is initially unlocalized, given two alternative conditions for the second robot  $\mathcal{R}_m$ : (1)  $\mathcal{R}_m$  is continuously localized (i.e., receives absolute position fixes at a frequency that is at least as high as the robot detection frequency) and (2)  $\mathcal{R}_m$  is initially localized (and then receives no more absolute position fixes).

## A.1 Analysis

In this analysis, we will show that the reciprocal sampling proportion  $\alpha$  can be tuned to affect the steady-state performance, and we will show how the convergence speed is affected when employing a finite number of particles. We derive the formulations for both the standard sampling algorithm (Algorithm 6 with  $\alpha = 0$ ), as well as our reciprocal sampling algorithm (Algorithm 6 with  $\alpha > 0$ ). To simplify the following formalisms, without loss of generality, we consider that the origin of the coordinate system coincides with the true position of robot  $\mathcal{R}_n$ . Thus, we assume that the state of any robot can be expressed in a 1-dimensional space as a position  $\rho$ , bounded by  $-\rho_{\max} \leq \rho \leq \rho_{\max}$ . The belief of robot  $\mathcal{R}_n$  at time  $t$  is, thus, simply given by  $\text{Bel}(\rho_{n,t})$ . In consequence, we formulate our error metric as the expectancy over all possible estimated distances  $\rho_{n,t}$  of robot  $\mathcal{R}_n$  to the origin

$$\mathbb{E}(\rho_{n,t}^2) = \int \text{Bel}(\rho_{n,t}) \cdot \rho_{n,t}^2 d\rho_n. \quad (\text{A.1})$$

The following two paragraphs detail  $\text{Bel}(\rho_{n,t})$  as well as  $\mathbb{E}(\rho_{n,t}^2)$  for  $t \rightarrow \infty$ , for both the standard sampling algorithm (SS) and the reciprocal sampling algorithm (RS), in cases (1) and (2).

### A.1.1 Robot $\mathcal{R}_m$ is continuously localized

We initialize  $\text{Bel}_{n,t=0} = 1/(2\rho_{max})$  as a uniform distribution over  $\rho$ , and  $\text{Bel}_{m,t} = \delta(\rho_m)$ , for all  $t$ , where  $\rho_m$  is the true position of robot  $\mathcal{R}_m$  and  $\delta(\cdot)$  the Dirac function.

Let us assume that the observation and motion models can be modeled with  $\Phi(\cdot, \sigma)$ , a zero-mean, normal probability density function with a standard deviation  $\sigma_{mn}$  (detection model) or  $\sigma_n$  (motion model), respectively. The belief  $\text{Bel}(\rho_{n,t})$  of robot  $\mathcal{R}_n$  at time  $t$  for the standard sampling algorithm then reads

$$\text{Bel}_{SS}(\rho_{n,t}) = \Phi(\rho_{n,t}, \sigma_{mn}) \int \Phi(\rho_{n,t} - \rho_{n,t-1}, \sigma_n) \cdot \text{Bel}_{SS}(\rho_{n,t-1}) d\rho_{t-1}. \quad (\text{A.2})$$

Given that the probability density of Equation A.2 is normal, we can easily calculate the steady-state error. For  $t \rightarrow \infty$ , Equation A.1 is

$$\mathbb{E}_{SS}(\rho_{n,t}^2) \xrightarrow{t \rightarrow \infty} \frac{\sigma_n}{2} \left( \sqrt{\sigma_n^2 + 4\sigma_{mn}^2} - \sigma_n^2 \right). \quad (\text{A.3})$$

We now extend this formalism to the case of reciprocal sampling with a reciprocal sampling proportion  $\alpha$ . We have

$$\begin{aligned} \text{Bel}_{RS}(\rho_{n,t}) &= \alpha \cdot \Phi(\rho_{n,t}, \sigma_{mn}) + (1 - \alpha) \cdot \Phi(\rho_{n,t}, \sigma_{mn}) \cdot \\ &\quad \int \Phi(\rho_{n,t} - \rho_{n,t-1}, \sigma_n) \cdot \text{Bel}_{RS}(\rho_{n,t-1}) d\rho_{n,t-1}. \end{aligned} \quad (\text{A.4})$$

The steady-state error of Equation A.4 will vary, depending on the reciprocal proportion  $\alpha$ . For this work, it suffices to consider the maximum possible steady state error, which is simply given by  $\sigma_{mn}^2$  (for  $\alpha = 1$ ).

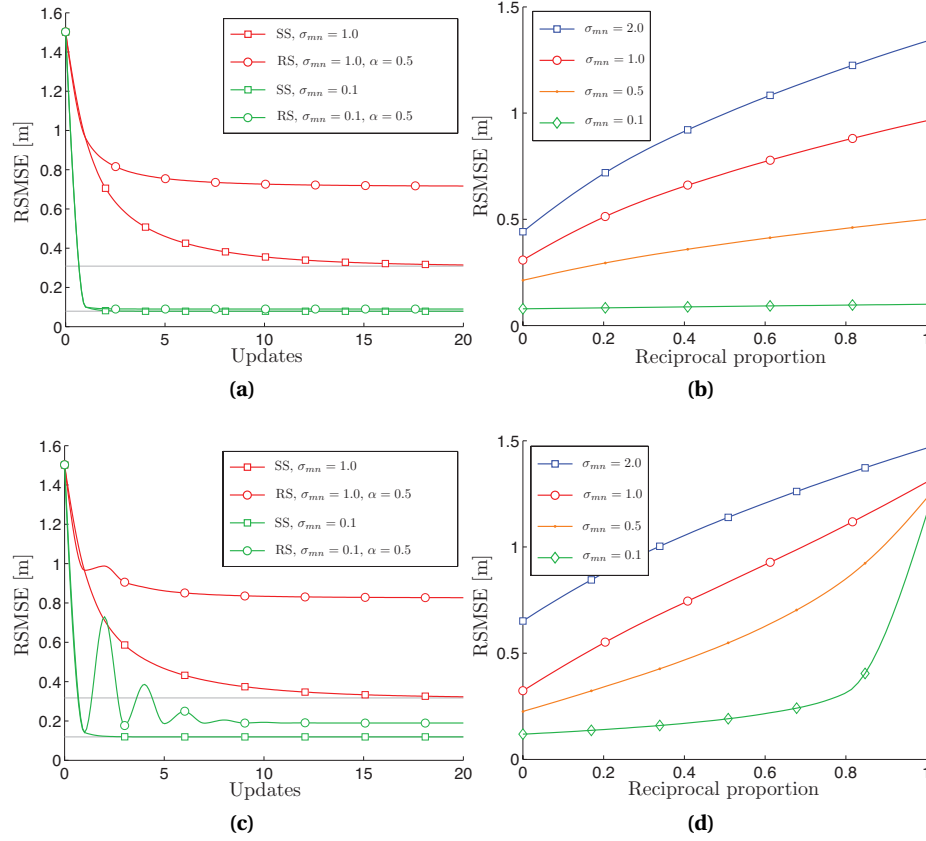
### A.1.2 Robot $\mathcal{R}_m$ only initially localized

As before, we initialize  $\text{Bel}_{n,t=0} = 1/(2\rho_{max})$ , and  $\text{Bel}_{m,t=0} = \delta(\rho_m)$ . As robot  $\mathcal{R}_m$  does not receive regular position fixes anymore, we must now reformulate the equations above to include the reciprocal sampling mechanism for both robots simultaneously. Without loss of generality, we assume that the moment of information exchange takes place at each update step after application of the motion model when the prior  $\overline{\text{Bel}}(\rho_{\cdot,t})$  has been calculated. Thus, when a robot  $\mathcal{R}_m$  detects a robot  $\mathcal{R}_n$ , it transmits its prior  $\overline{\text{Bel}}(\rho_{m,t})$ , defined as

$$\overline{\text{Bel}}(\rho_{m,t}) = \int \Phi(\rho_{m,t} - \rho_{m,t-1}, \sigma_m) \cdot \text{Bel}(\rho_{m,t-1}) d\rho_{m,t-1}. \quad (\text{A.5})$$

The belief of robot  $\mathcal{R}_n$  for the standard sampling algorithm is then

$$\text{Bel}_{SS}(\rho_{n,t}) = \overline{\text{Bel}}_{SS}(\rho_{n,t}) \cdot \int \Phi(\rho_{n,t}, \sigma_{mn}) \cdot \overline{\text{Bel}}_{SS}(\rho_{m,t}) d\rho_{m,t}. \quad (\text{A.6})$$



**Figure A.1:** The plots show in (a) and (c) the development of the the root-mean-square-error for the standard sampling (SS) and reciprocal sampling (RS) algorithms, and in (b) and (d) the steady-state error in function of the reciprocal sampling proportion. The first row shows results for robot  $\mathcal{R}_m$  continuously localized, and the second row for robot  $\mathcal{R}_m$  only initially localized. The space was defined by  $\rho_{max} = 2.6$  m and a constant motion noise of  $\sigma_n = 0.1$  m was employed.

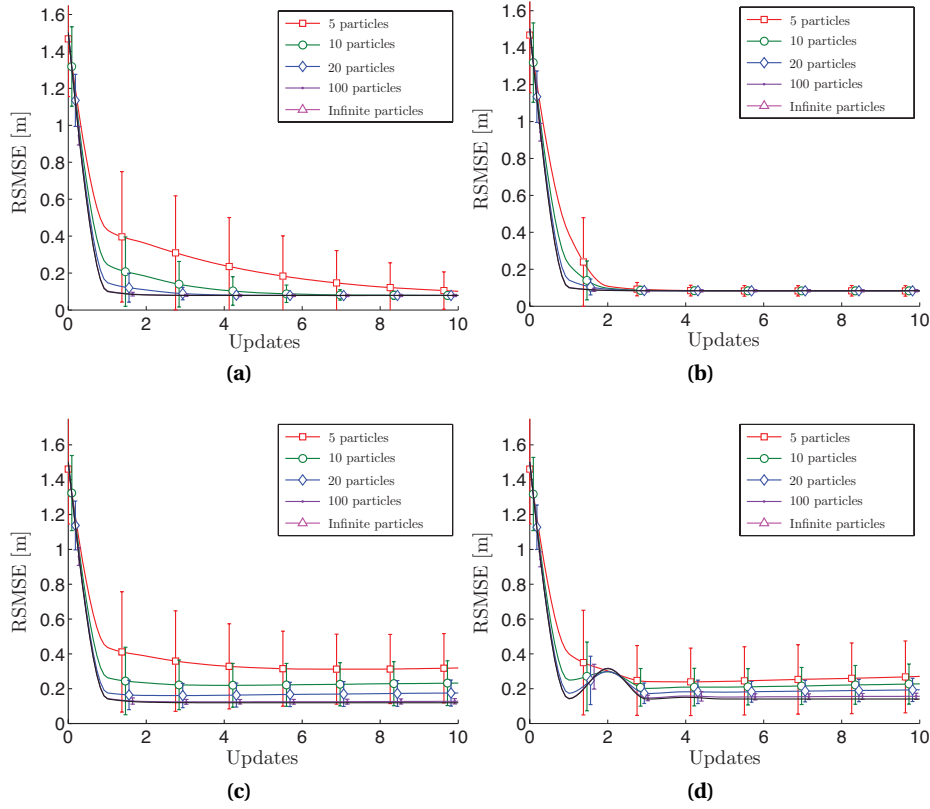
As in Equation A.3, we formulate the steady state error of Equation A.6 as

$$\mathbb{E}_{SS}(\rho_{n,t}^2) \xrightarrow{t \rightarrow \infty} \sigma_n \sqrt{\sigma_{mn}^2 + \sigma_n^2}. \quad (\text{A.7})$$

Analogously, we extend the formalism for the reciprocal sampling algorithm

$$\begin{aligned} \mathbf{Bel}_{RS}(\rho_{n,t}) &= \alpha \cdot \int \Phi(\rho_{n,t}, \sigma_{mn}) \cdot \overline{\mathbf{Bel}}_{RS}(\rho_{m,t}) d\rho_{m,t} + \\ &\quad (1 - \alpha) \cdot \int \Phi(\rho_{n,t}, \sigma_{mn}) \cdot \overline{\mathbf{Bel}}_{RS}(\rho_{m,t}) d\rho_{m,t} \cdot \overline{\mathbf{Bel}}_{RS}(\rho_{n,t}). \end{aligned} \quad (\text{A.8})$$

In this particular case, the steady-state cannot be found analytically. Yet, we note that the maximal steady-state error of Equation A.8 for  $\alpha = 1$  is unbounded for an unbounded space. Equations A.5–A.8 are equally formulated for robot  $\mathcal{R}_m$ .



**Figure A.2:** The figures show the root-mean-square-error for (a), (c) standard sampling and (b), (d) reciprocal sampling, in a space defined by  $\rho_{max} = 2.6\text{m}$ . A variable number of particles was employed, ranging from a total of 5 to 100 particles. Additionally, the solid black line shows the results for an ‘infinity’ of particles (by numerically evaluating the closed-form equation). We employed a motion noise  $\sigma_n = 0.1$  and detection noise  $\sigma_{mn} = 0.1$  for all plots, and a reciprocal sampling proportion of  $\alpha = 0.2$  for plots (b) and (d). The first row shows results for robot  $\mathcal{R}_m$  continuously localized, the second row for robot  $\mathcal{R}_m$  only initially localized. For each experiment employing a finite set of particles, 5000 runs were performed. The errorbars show the standard deviation.

## A.2 Discussion

In order to analyze the performance of the filters in Equations A.2, A.4, A.6 and A.8, we resort to a numerical solution for each update  $t$ , and discuss the localization performance of robot  $\mathcal{R}_n$ .

Figure A.1 shows the performance for standard and reciprocal sampling algorithms in an ideal filter (with an infinity of particles). A gray line marks the lower-bound error derived from Equations A.3 and A.7. The steady-state bounds in Figure A.1b vary between the steady-state errors defined by Equation A.3 and  $\sigma_{mn}$ , and provide a lower bound on the steady-state performance in Fig. A.1d. Since this analysis discusses an ideal filter with an ‘infinity’ of particles (we numerically evaluate the analytical formula), we do not expect any benefit from the reciprocal sampling algorithm. However, the results show that in particular for moderate observation noise values  $\sigma_{mn}$ , the loss of accuracy is very small, regardless of the reciprocal sampling proportion  $\alpha$ .



Figure A.2 shows the localization error for both standard sampling (first column) and reciprocal sampling (second column) for a variable set of finite particle numbers. We see that for an increasing number of particles, the localization error converges to that of the ideal filter (with an infinity of particles). In contrast to Figure A.1, we observe how, in the case of a finite number of particles, the reciprocal sampling algorithm converges faster to the steady-state error. Even when employing as little as 5 particles in total, 1 reciprocal particle is enough to accelerate convergence nearly 10-fold.

Indeed, for an infinity of particles, the standard sampling algorithm will always outperform reciprocal sampling. Yet, for any finite number of particles, the reciprocal sampling algorithm is highly likely to accelerate convergence. Moreover, we note that for modest detection noise values  $\sigma_{mn}$ , we are able to benefit from the reciprocal sampling algorithm, regardless of the sampling proportion  $\alpha$ , without significantly affecting the steady-state performance of the system. In conclusion, we note that while  $\sigma_{mn}$  and  $\sigma_n$  are often given by the real system at hand, parameters such as  $\alpha$  and the number of particles can be tuned and thus adapted to available computational and communication resources.



## B Simultaneous Localization and Mapping of UWB Error Models

In our work, we assume the possibility of creating the UWB error maps a priori, by some alternative means of localization. In other words, our framework enables the creation of UWB error maps given that  $\mathbf{x}_t$  is known (confer with Algorithm 4 on page 64). Now, we will consider the problem of simultaneously estimating  $\mathbf{x}_t$  and building the set of maps  $\mathcal{M}$ . The goal of this section is to show that this extended problem can be solved with the very same framework introduced in our work—in particular, this new problem formulation will require the application of our online model estimation algorithm within the run-time localization algorithm. For brevity, we will omit the robot index  $n$  in our notations (and as a consequence, consider a single-robot setting).

Analogous to Equation (6.3), we formulate the belief of the robot's pose and the set of true maps as

$$\text{Bel}(\mathbf{x}_t, \mathcal{M}_t) \sim \{\langle \mathbf{x}_t^{[i]}, w_t^{[i]}, \mathcal{M}_t^{[i]} \rangle | i = 1, \dots, M\} = Y_t \quad (\text{B.1})$$

where, in addition to a pose estimate, each particle now also maintains its own set of maps  $\mathcal{M}^{[i]}$ . Equation (5.2) deals with the association of a specific UWB error model and an area  $A_a$ , given a position  $\mathbf{x}_t$ . Without the knowledge of  $\mathbf{x}_t$ , however, the new data association problem is formulated as follows. For a given base station pair  $\langle \mathcal{B}_u, \mathcal{B}_v \rangle$ , we have for each particle  $i$

$$m_{uv}(\mathbf{x}_t^{[i]}) = \theta_{uv,a}, \text{ such that } \exists \langle A_a, \theta_{uv,a} \rangle \in \mathcal{M}_{uv}^{[i]} \text{ with } \mathbf{x}_t^{[i]} \in A_a, \text{ and } \mathcal{M}_{uv}^{[i]} \in \mathcal{M}^{[i]}. \quad (\text{B.2})$$

Thus, the correspondence of feature  $\langle A_a, \theta_{uv,a} \rangle$  for any given measurement is directly determined by the particle position  $\mathbf{x}_t^{[i]}$ . This approach is valid because the areas  $A_a$  and the number of features  $N_A$  are static over time. Following, the TDOA measurement error for a particle  $i$  and base station pair  $\langle \mathcal{B}_u, \mathcal{B}_v \rangle$  is defined as the difference between the ‘nominal’ TDOA value perceived at the particle position, and the measured TDOA value  $\hat{\tau}_{uv,t}$  (refer to Equation (5.1)):

$$\Delta\tau_{uv,t}^{[i]} \triangleq \Delta\tau_{uv}(\hat{\tau}_{uv,t}, \mathbf{x}_t^{[i]}). \quad (\text{B.3})$$

This value is used to update the UWB error map via Algorithm 4, as introduced in Section 6.3. The resulting *simultaneous localization and mapping* routine is summarized in Algorithm 10.

## Appendix B. Simultaneous Localization and Mapping of UWB Error Models

---

### Algorithm 10 UWB\_SLAM( $Y_{t-1}, u_t, T_t$ )

---

```

1:  $\bar{Y}_t = Y_t = \emptyset$ 
2: for  $i = 1$  to  $M$  do
3:    $\mathbf{x}_t^{[i]} \leftarrow \text{Motion\_Model}(u_t, \mathbf{x}_{t-1}^{[i]})$ 
4:    $w_t^{[i]} \leftarrow \text{Measurement\_Model\_Map}(T_t, \mathcal{M}_{t-1}^{[i]}, \mathbf{x}_t^{[i]}, w_t^{[i]})$ 
5:    $\mathcal{M}_t^{[i]} \leftarrow \text{Update\_Map}(\mathbf{x}_t^{[i]}, T_t, \mathcal{M}_{t-1}^{[i]})$ 
6:    $\bar{Y}_t \leftarrow \bar{Y}_t \cup \langle \mathbf{x}_t^{[i]}, w_t^{[i]}, \mathcal{M}_t^{[i]} \rangle$ 
7: end for
8: for  $i = 1$  to  $M$  do
9:    $\langle \mathbf{x}_t^{[i]}, \mathcal{M}_t^{[i]} \rangle \leftarrow \text{Sampling}(\bar{Y}_t)$ 
10:   $Y_t \leftarrow Y_t \cup \langle \mathbf{x}_t^{[i]}, w_t^{[i]}, \mathcal{M}_t^{[i]} \rangle$ 
11: end for
12: return  $Y_t$ 

```

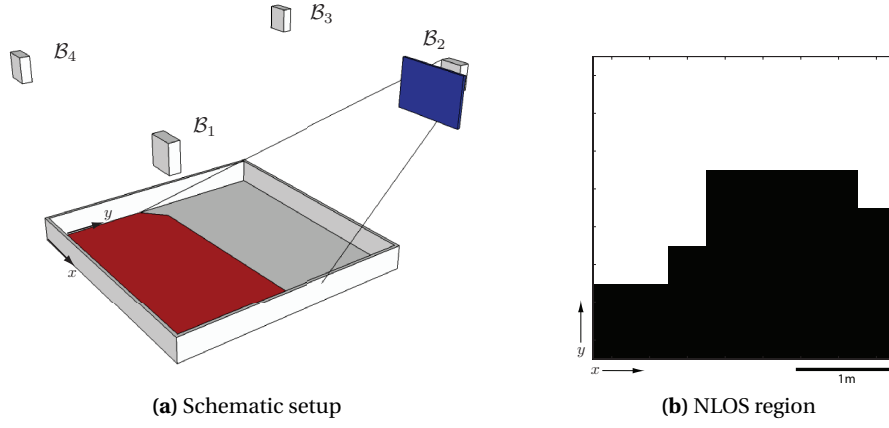
---

We note that, since each map has a set of  $N_A$  separate areas, which can be treated as independent, and these areas (or features) are independent from the robot's path, our SLAM algorithm bears strong resemblance to FastSLAM [119].

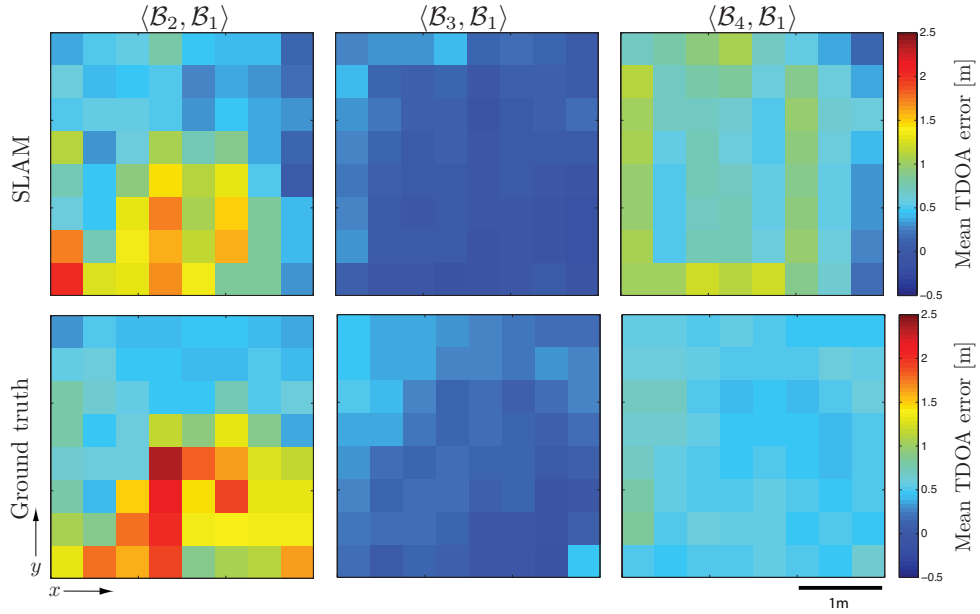
We test this algorithm in a simplified setup with a single robot. The robot arena is reduced in size ( $9\text{m}^2$ ) and contains no obstacles other than a metal sheet, which is placed in front of base station  $B_2$  to provoke a NLOS scenario. We illustrate this setup in Figure B.1a. Using this setup, we perform experiments of 50 min duration (throughout this time-span, the robot is able to completely cover the arena). The robot's particle filter runs with a set of 100 particles (initially localized) and employs a standard motion model based on the odometry readings.

We qualitatively evaluate the performance of our algorithm by considering the maps built by our SLAM algorithm. Figure B.2 reports the mean value of the estimated error model for each area  $A_n$ , for each map and for each base station pair. The estimated maps (top row) show a qualitatively good correspondence with the ground truth maps (bottom row). The average error (obtained by comparing the mean TDOA error values of the true and estimated maps) is 0.2 m, which is relatively small with respect to the range of observed mean error values, roughly 3 m. A schematized version of our map is obtained by imposing a threshold on the true map  $\mathcal{M}_{21}$  (see Figure B.1b), and shows good correspondence with the NLOS scenario we intended to create in our setup.

This work shows one possible implementation of a simultaneous localization and mapping solution. More sophisticated algorithms should consider ways of reducing the computational complexity, in particular in scenarios where a high number of particles is needed. Extensive quantitative evaluations should also focus on the achieved localization accuracy using the estimated maps. Finally, when using the method in multi-robot systems, further improvements could consider the integration of cooperative methods (such as seen in the map-alignment problem).



**Figure B.1:** (a) A schematic illustration of the arena including the occluding metal sheet in front base-station  $B_2$ . A NLOS region is projected onto the ground plane (which is at the approximate height of the robot). (b) Based on ground-truth positions  $\mathbf{x}_n$ , we produce a NLOS-LOS map by imposing a threshold of  $\pm 0.8\text{m}$  on the TDOA error  $\Delta\tau_{uv}(\hat{\tau}_{uv}, \mathbf{x}_n)$ . The pattern corresponds to the projected NLOS area obtained in (a).



**Figure B.2:** Grid map with 64 cells showing the mean TDOA error (in meters) for two experiments: The top row shows the maps built by the UWB SLAM algorithm for each of the three base station pairs. In order to compute the values shown in the plot, we calculate the mean of the probability density function of our error model  $E[p(\cdot; \theta_{uv})]$  analytically, for each cell in the grid map. The bottom row shows the mean ground truth error value  $\Delta\tau_{uv}(\hat{\tau}_{uv,t}, \mathbf{x}_t)$ , for  $\mathbf{x}_t$  the true position. We use the ground truth map  $\mathcal{M}_{21}$  to produce the map which illustrates the LOS-NLOS layout in Figure B.1b.



# Glossary

AOA	Angle of Arrival	5–7, 10, 37, 38, 40
CDF	Cumulative Density Function	7
CEP	Circular Error Probability	7, 9
CRLB	Cramér Rao Lower Bound	35, 51, 52, 57, 134
EM	Expectation Maximization	29, 57, 58, 60, 61, 66, 134
FCC	Federal Communications Commission	33, 35
GPS	Global Positioning System	3, 5, 9, 11
INS	Inertial Navigation System	72
IR	Infrared	6, 11, 13–16, 114
LOS	Line of Sight	6, 10, 36, 45, 52, 57, 61, 114, 127
MEMS	Micro Electro Mechanical Sensors	72
NLOS	Non Line of Sight	5, 10, 29, 36, 38, 45, 46, 52, 57, 61, 72, 114, 127, 134, 144
POA	Phase of Arrival	5, 6
RF	Radio Frequency	4, 9, 11, 15, 16, 35
RFID	Radio Frequency Identification	6, 9, 11, 15
RSS	Received Signal Strength	5, 6, 8, 10, 15
RTOF	Roundtrip Time of Flight	5, 6
SLAM	Simultaneous Localization and Mapping	13, 18, 127, 136, 144
SNR	Signal-to-Noise Ratio	35, 36
SQP	Sequential Quadratic Programming	61
TDOA	Time Difference of Arrival	5, 6, 8, 10, 15, 29, 35, 37, 38, 40, 43–46, 57, 70, 72, 117, 133, 136, 143, 144
TOA	Time of Arrival	5, 6, 29, 34–37, 44, 46, 57, 133
UDP	User Datagram Protocol	80
UWB	Ultra-Wideband	10, 11, 15, 18, 27–30, 33–38, 40, 41, 43, 44, 50, 53, 54, 56, 57, 60, 61, 67, 68, 70–74, 77, 107, 111–115, 117–119, 122–129, 133–136, 143
WLAN	Wireless Local Area Network	10, 11





# Bibliography

- [1] N. Alsindi, B. Alavi, and K. Pahlavan, "Measurement and Modeling of Ultrawideband TOA-Based Ranging in Indoor Multipath Environments", *IEEE Transactions on Vehicular Technology*, vol. 58, pp. 1046–1058, 2009 (cit. on pp. 46, 48, 50).
- [2] G. Antonelli, S. Chiaverini, and G. Fusco, "A calibration method for odometry of mobile robots based on the least-squares technique: theory and experimental validation", *IEEE Transactions on Robotics*, vol. 21, no. 5, pp. 994–1004, 2005 (cit. on pp. 19, 22, 23).
- [3] R. C. Arkin, "Motor Schema-based Mobile Robot Navigation", *The International Journal of Robotics Research*, vol. 8, no. 4, pp. 92–112, 1989 (cit. on p. 18).
- [4] P. Bahl and V. N. Padmanabhan, "RADAR: an in-building RF-based user location and tracking system", *INFOCOM, Annual Joint Conference of the IEEE Computer and Communications Societies*, pp. 775–784, 2000 (cit. on p. 10).
- [5] A. Bahr, J. J. Leonard, and M. F. Fallon, "Cooperative Localization for Autonomous Underwater Vehicles", *Int. Journal of Robotics Research*, vol. 28, no. 6, pp. 714–728, 2009 (cit. on p. 106).
- [6] A. Bahr, M. R. Walter, and J. J. Leonard, "Consistent cooperative localization", *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3415–3422, 2009 (cit. on p. 106).
- [7] G. H. Ball and D. J. Hall, "ISODATA, a novel method of data analysis and classification", Tech. Rep., 1965 (cit. on p. 97).
- [8] V. Barge-Cherfaoui and B. Vachon, "A multi-agent approach of the multi-sensor fusion", *International Conference on Advanced Robotics (ICAR)*, pp. 1264–1269, 1991 (cit. on p. 112).
- [9] R. Battiti, T. Le Nhat, and A. Villani, "Location-aware computing: a neural network model for determining location in wireless LANs", Tech. Rep. DIT-02-0083, 2002 (cit. on p. 10).
- [10] J. L. Bentley, "Multidimensional Divide-and-Conquer", *Communications of the ACM*, vol. 23, pp. 214–229, 1980 (cit. on p. 97).
- [11] J. Biswas and M. Veloso, "Wifi localization and navigation for autonomous indoor mobile robots", *IEEE International Conference on Robotics and Automation*, pp. 4379–4384, 2010 (cit. on p. 15).
- [12] M. Bloesch, S. Weiss, D. Scaramuzza, and R. Siegwart, "Vision based MAV navigation in unknown and unstructured environments", *IEEE International Conference Robotics and Automation (ICRA)*, pp. 21–28, 2010 (cit. on p. 13).
- [13] J. Borenstein and L. Feng, "Measurement and correction of systematic odometry errors in mobile robots", *IEEE Transactions on Robotics and Automation*, vol. 12, no. 6, pp. 869–880, 1996 (cit. on p. 23).
- [14] M. Bouet and A. L. dos Santos, "RFID tags: Positioning principles and localization techniques", *Wireless Days, 1st IFIP*, 2008. DOI: 10.1109/WD.2008.4812905 (cit. on p. 9).
- [15] V. Braitenberg, *Vehicles: Experiments in synthetic psychology*. MIT Press, 1984 (cit. on pp. 20, 112).

## Bibliography

---

- [16] R. Brega, N. Tomatis, and K. Arras, "The need for autonomy and real-time in mobile robotics: a case study of XO/2 and Pygmalion", *Proceedings of the 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1422–1427, 2000 (cit. on p. 18).
- [17] R. Brooks, "A robust layered control system for a mobile robot", *IEEE Journal of Robotics and Automation*, vol. 2, no. 1, pp. 14–23, 1986 (cit. on p. 112).
- [18] N. Bulusu, J. Heidemann, and D. Estrin, "GPS-less Low-Cost Outdoor Localization for Very Small Devices", *IEEE Personal Communications*, vol. 7, no. 5, pp. 28–34, 2000 (cit. on p. 77).
- [19] O. Cappe and E. Moulines, "On-line expectation-maximization algorithm for latent data models", *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 71, pp. 593–613, 2009 (cit. on pp. 58, 60).
- [20] H. Carvalho, P. Del Moral, A. Monin, and G. Salut, "Optimal nonlinear filtering in GPS/INS integration", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 33, no. 3, pp. 835–850, 1997 (cit. on p. 72).
- [21] A. Censi, L. Marchionni, and G. Oriolo, "Simultaneous maximum-likelihood calibration of odometry and sensor parameters", *Proceedings of the 2008 IEEE International Conference on Robotics and Automation*, pp. 2098–2103, 2008 (cit. on p. 19).
- [22] B. Charrow, N. Michael, and V. Kumar, "Cooperative Multi-Robot Estimation and Control for Radio Source Localization", *International Symposium on Experimental Robotics (ISER)*, 2012 (cit. on p. 127).
- [23] C. M. Cianci, J. Nembrini, **A. Prorok**, and A. Martinoli, "Assembly of Configurations in a Networked Robotic System: A Case Study on a Reconfigurable Interactive Table Lamp", *IEEE Swarm Intelligence Symposium (SIS)*, 2008. DOI: 10.1109/SIS.2008.4668318 (cit. on p. 158).
- [24] I. Cox, "Blanche - An experiment in guidance and navigation of an autonomous robot vehicle", *IEEE Transactions on Robotics and Automation*, vol. 7, no. 2, pp. 193–204, 1991 (cit. on p. 18).
- [25] A. Cristofaro, A. Renzaglia, and A. Martinelli, "Distributed Information Filters for MAV Cooperative Localization", *International Symposium on Distributed Autonomous Robotic Systems (DARS), Springer Tracts in Advanced Robotics*, pp. 133–146, 2013 (cit. on p. 106).
- [26] B. Denis, J. Keignart, and N. Daniele, "Impact of NLOS propagation upon ranging precision in UWB systems", *IEEE Conference on Ultra Wideband Systems and Technologies*, pp. 379–383, 2003 (cit. on p. 36).
- [27] G. N. Desouza and A. C. Kak, "Vision for mobile robot navigation: a survey", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 2, pp. 237–267, 2002 (cit. on p. 13).
- [28] C. Drane, M. Macnaughtan, and C. Scott, "Positioning GSM telephones", *IEEE Communications Magazine*, vol. 36, pp. 46–59, 1998 (cit. on p. 4).
- [29] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: part I", *Robotics & Automation Magazine, IEEE*, vol. 13, no. 2, pp. 99–110, 2006 (cit. on p. 13).
- [30] F. Endres, J. Hess, N. Engelhard, J. Sturm, D. Cremers, and W. Burgard, "An evaluation of the RGB-D SLAM system", *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1691–1696, 2012 (cit. on p. 13).
- [31] H. R. Everett, *Everett, H. R., 1995, Sensors for Mobile Robots: Theory and Application*. A. K. Peters, Ltd., Wellesley, MA, 1995 (cit. on p. 13).
- [32] M. F. Fallon, H. Johannsson, and J. J. Leonard, "Efficient Scene Simulation for Robust Monte Carlo Localization using an RGB-D Camera", *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1663–1670, 2012 (cit. on p. 13).
- [33] J. Fink and V. Kumar, "Online methods for radio signal mapping with mobile robots", *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1940–1945, 2010 (cit. on p. 83).

- [34] A. M. Flynn, "Combining Sonar and Infrared Sensors for Mobile Robot Navigation", *The International Journal of Robotics Research*, vol. 7, no. 6, pp. 5–14, 1988 (cit. on p. 14).
- [35] D. Fox, W. Burgard, H. Kruppa, and S. Thrun, "A Probabilistic Approach to Collaborative Multi-Robot Localization", *Autonomous Robots*, vol. 8, pp. 325–344, 2000 (cit. on pp. 81, 91, 92, 106).
- [36] B. Freedman, A. Shpunt, M. Machline, and Y. Arieli, "United States Patent Application Publication. Depth Mapping Using Projected Patterns", pp. 1–12, May 2010 (cit. on p. 13).
- [37] J. Gaspar, N. Winters, and J. Santos-Victor, "Vision-based navigation and environmental representations with an omnidirectional camera", *IEEE Transactions on Robotics and Automation*, vol. 16, no. 6, pp. 890–898, 2000 (cit. on p. 13).
- [38] P. Giguere, I. Rekleitis, and M. Latulippe, "I See You, You See Me: Cooperative Localization through Bearing-Only Mutually Observing Robots", *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 863–869, 2012 (cit. on p. 81).
- [39] L. Girod and D. Estrin, "Robust range estimation using acoustic and multimodal sensing", *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2001. DOI: 10.1109/IROS.2001.977164 (cit. on p. 111).
- [40] J. Gonzalez, J. L. Blanco, C. Galindo, A. Ortiz-de Galisteo, J. A. Fernandez-Madrigal, F. A. Moreno, and J. L. Martinez, "Mobile robot localization based on Ultra-Wide-Band ranging: A particle filter approach", *Robotics and Autonomous Systems*, vol. 57, no. 5, pp. 496–507, 2009 (cit. on pp. 15, 18, 71, 72).
- [41] S. Goyal, **A. Prorok**, and A. Martinoli, "Two-Phase Online Calibration for Infrared-Based Inter-Robot Positioning Modules", *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3313–3319, 2011 (cit. on pp. 28, 82, 83, 158).
- [42] R. L. Greenspan, "Inertial Navigation technology from 1970-1995", *Navigation*, vol. 42, no. 1, pp. 165–185, 1996 (cit. on p. 72).
- [43] A. Gutiérrezrez, A. Campo, M. Dorigo, J. Donate, F. Monaserio-Huelin, and L. Magdalena, "Open e-puck range & bearing miniaturized board for local communication in swarm robotics", *IEEE International Conference on Robotics and Automation*, pp. 3111–3116, 2009 (cit. on p. 81).
- [44] J.-S. Gutmann, W. Burgard, D. Fox, and K. Konolige, "An experimental comparison of localization methods", *Proceedings of the 1998 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 736–743, 1998 (cit. on p. 25).
- [45] D. Hähnel, W. Burgard, D. Fox, K. Fishkin, and M. Philipose, "Mapping and Localization with RFID Technology", *IEEE International Conference on Robotics and Automation*, pp. 1015–1020, 2004 (cit. on p. 15).
- [46] J. A. Hartigan, *Clustering Algorithms*. John Wiley and Sons, Inc., New York, NY, 1975 (cit. on p. 97).
- [47] J. Haverinen and A. Kemppainen, "Global indoor self-localization based on the ambient magnetic field", *Robotics and Autonomous Systems*, vol. 57, no. 10, pp. 1028–1035, 2009 (cit. on p. 14).
- [48] R. Herbrich, "Minimising the Kullback-Leibler Divergence", Tech. Rep., 2005 (cit. on p. 98).
- [49] J. Hightower and G. Borriello, "Location systems for ubiquitous computing", *Computer*, vol. 34, no. 8, pp. 57–66, 2001 (cit. on pp. 4, 5, 9).
- [50] G. A. Hollinger, J. Djughash, and S. Singh, "Target tracking without line of sight using range from radio", *Autonomous Robots*, vol. 32, no. 1, pp. 1–14, 2012 (cit. on pp. 15, 71, 72).
- [51] J. N. How, B. Bethke, A. Frank, D. Dale, and J. Vian, "Real-time indoor autonomous vehicle test environment", *IEEE Control Systems Magazine*, vol. 28, no. 2, pp. 51–64, 2008 (cit. on p. 16).
- [52] A. Howard, "Multi-robot Simultaneous Localization and Mapping using Particle Filters", *Int. Journal of Robotics Research*, vol. 25, no. 12, pp. 1243–1256, 2006 (cit. on p. 127).

## Bibliography

---

- [53] A. Howard, M. J. Mataric, and G. S. Sukhatme, "Localization for mobile robot teams: A distributed MLE approach", *Experimental Robotics VIII*, pp. 146–155, 2003 (cit. on pp. 81, 105).
- [54] A. Howard, S. Siddiqi, and G. S. Sukhatme, "An experimental study of localization using wireless ethernet", *Springer Tracts in Advanced Robotics, Field and Service Robotics*, vol. 24, pp. 145–153, 2006 (cit. on p. 15).
- [55] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data Clustering: A Review", *ACM Computing Surveys*, vol. 31, no. 3, pp. 265–323, 1999 (cit. on p. 96).
- [56] J. S. Jennings, G. Whelan, and W. F. Evans, "Cooperative search and rescue with a team of mobile robots", *International Conference on Advanced Robotics (ICAR)*, pp. 193–200, 1997 (cit. on p. 77).
- [57] D. Jourdan, J. J. Deyst, M. Z. Win, and N. Roy, "Monte Carlo localization in dense multipath environments using UWB ranging", *IEEE International Conference on Ultra-Wideband (ICU)*, pp. 314–319, 2005 (cit. on pp. 15, 71, 72).
- [58] G. Kantor, S. Singh, R. Peterson, D. Rus, A. Das, V. Kumar, G. Perreira, and J. Spletzer, "Distributed Search and Rescue with Robot and Sensor Teams", *Springer Tracts in Advanced Robotics*, vol. 24, pp. 529–538, 2006 (cit. on p. 77).
- [59] A. Kelly, "Fast and easy systematic and stochastic odometry calibration", *Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3188–3194, 2004 (cit. on p. 23).
- [60] K. Khoshelham, "Accuracy Analysis of Kinect Depth Data", *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 38, pp. 133–138, 2011 (cit. on pp. 13, 111).
- [61] L. Kleeman, "Optimal estimation of position and heading for mobile robots using ultrasonic beacons and dead-reckoning", *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2582–2587, 1992 (cit. on p. 15).
- [62] L. G. W. Koontz, P. Narendra, and K. Fukunaga, "A Branch and Bound Clustering Algorithm", *IEEE Transactions on Computers*, vol. C-24, no. 9, pp. 908–915, 1975 (cit. on p. 96).
- [63] Y. Koren and J. Borenstein, "Potential field methods and their inherent limitations for mobile robot navigation", *Proceedings of the 1991 IEEE International Conference on Robotics and Automation*, pp. 1398–1404, 1991 (cit. on p. 18).
- [64] A. Ladd, K. Bekris, A. Rudys, L. Kavraki, and D. Wallach, "Robotics-based location sensing using wireless ethernet", *Wireless Networks*, vol. 11, no. 1-2, pp. 189–204, 2005 (cit. on pp. 15, 18).
- [65] J. E. Lenz, "A review of magnetic sensors", *Proceedings of the IEEE*, vol. 78, pp. 973–989, 1990 (cit. on p. 14).
- [66] J. J. Leonard and H. F. Durrant-Whyte, *Directed Sonar Sensing for Mobile Robot Navigation*. Kluwer Academic Publishers, 1992 (cit. on p. 25).
- [67] —, "Mobile Robot Localization by Tracking Geometric Beacons", *IEEE Transactions on Robotics and Automation*, vol. 7, pp. 376–382, 1998 (cit. on p. 12).
- [68] K. Y. K. Leung, T. D. Barfoot, and H. Liu, "Decentralized Localization of Sparsely-Communicating Robot Networks: A Centralized-Equivalent Approach", *IEEE Transactions on Robotics*, vol. 26, no. 1, pp. 62–77, 2010 (cit. on p. 106).
- [69] Q. Lindsey, D. Mellinger, and V. Kumar, "Construction of cubic structures with quadrotor teams", *Proceedings of Robotics: Science & Systems VII*, pp. 198–205, 2012 (cit. on p. 77).
- [70] H. Liu, H. Darabi, P. Banerjee, and J. Liu, "Survey of Wireless Indoor Positioning Techniques and Systems", *IEEE Transactions on Systems, Man and Cybernetics*, vol. 37, no. 6, pp. 1067–1080, 2007 (cit. on pp. 4, 5, 7, 9, 73).

- [71] T. Lochmatter, P. Roduit, C. Cianci, N. Correll, J. Jacot, and A. Martinoli, "SwisTrack - A Flexible Open Source Tracking Software for Multi-Agent Systems", *Proceedings of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4004–4010, 2008 (cit. on p. 21).
- [72] K. Lorincz, D. J. Malan, T. R. F. Fulford-Jones, A. Nawoj, A. Clavel, V. Shnayder, G. Mainland, M. Welsh, and S. Moulton, "Sensor networks for emergency response: challenges and opportunities", *Pervasive Computing*, vol. 3, no. 4, pp. 16–23, 2004 (cit. on p. 3).
- [73] A. Martinelli, F. Pont, and R. Siegwart, "Multi-Robot Localization Using Relative Observations", *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2797–2802, 2005 (cit. on p. 105).
- [74] J. McQueen, "Some methods for classification and analysis of multivariate observations", *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, pp. 281–297, 1967 (cit. on p. 97).
- [75] P. P. Mercier, D. C. Daly, M. Bhardwaj, D. D. Wentzloff, F. S. Lee, and A. P. Chandrakasan, "Ultra-low-power UWB for sensor network applications", *IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 2562–2565, 2008 (cit. on p. 35).
- [76] G. Mermoud, **A. Prorok**, L. Matthey, C. M. Cianci, N. Correll, and A. Martinoli, "Self-Organized Robotic Systems: Large-Scale Experiments in Aggregation and Self-Assembly using Miniature Robots", in *Handbook of Collective Robotics*, S. Kernbach, Ed., Pan Stanford Publishing, 2012 (cit. on p. 158).
- [77] N. Michael, D. Mellinger, Q. Lindsey, and V. Kumar, "The GRASP Multiple Micro-UAV Testbed", *Robotics & Automation Magazine, IEEE*, vol. 17, no. 3, 2010 (cit. on p. 16).
- [78] D. Moore, J. Leonard, D. Rus, and S. Teller, "Robust distributed network localization with noisy range measurements", *Proceedings of the 2nd international conference on Embedded networked sensor systems (SENSYS)*, pp. 50–61, 2004 (cit. on p. 81).
- [79] R. Mosberger and H. Andreasson, "Estimating the 3D Position of Humans wearing a Reflective Vest using a Single Camera System", *International Conference on Field and Service Robotics (FSR)*, 2012 (cit. on p. 3).
- [80] A. Mourikis and S. I. Roumeliotis, "Optimal sensor scheduling for resource-constrained localization of mobile robot formations", *IEEE Transactions on Robotics*, vol. 22, no. 5, pp. 917–931, 2006 (cit. on p. 106).
- [81] B. Neal, "Slice Sampling", *The Annals of Statistics*, vol. 31, pp. 705–757, 2003 (cit. on p. 93).
- [82] E. Nerurkar and S. Roumeliotis, "Asynchronous Multi-Centralized Cooperative Localization", *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4352–4359, 2010 (cit. on p. 105).
- [83] E. Nerurkar, S. Roumeliotis, and A. Martinelli, "Distributed maximum a posteriori estimation for multi-robot cooperative localization", *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1402–1409, 2009 (cit. on p. 106).
- [84] K. Pahlavan, X. Li, and J. P. Makela, "Indoor geolocation science and technology", *IEEE Communications Magazine*, vol. 40, no. 2, pp. 112–118, 2002 (cit. on p. 4).
- [85] S. Park and S. Hashimoto, "Autonomous Mobile Robot Navigation Using Passive RFID in Indoor Environment", *IEEE Transactions on Industrial Electronics*, vol. 56, no. 7, pp. 2366–2373, 2009 (cit. on p. 15).
- [86] P. N. Pathirana, N. Bulusu, A. V. Savkin, and S. Jha, "Node localization using mobile robots in delay-tolerant sensor networks", *IEEE Transactions on Mobile Computing*, vol. 4, no. 3, pp. 285–296, 2005 (cit. on p. 77).
- [87] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, "The cricket location-support system", *International Conference on Mobile Computing and Networking (MOBICOM)*, pp. 32–43, 2000 (cit. on p. 15).

- [88] **A. Prorok**, A. Arfire, A. Bahr, J. R. Farserotu, and A. Martinoli, “Indoor Navigation Research with the Khepera III Mobile Robot: An Experimental Baseline with a Case-Study on Ultra-Wideband Positioning”, *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2010. DOI: 10.1109/IPIN.2010.5647880 (cit. on pp. 20, 28, 158).
- [89] **A. Prorok**, A. Bahr, and A. Martinoli, “Low-Cost Collaborative Localization for Large-Scale Multi-Robot Systems”, *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4236–4241, 2012 (cit. on pp. 28, 158).
- [90] —, “Low-Cost Multi-Robot Localization”, in *Redundancy in Robot Manipulators and Multi-Robot Systems*, D. Milutinovic and J. Rosen, Eds., Lecture Notes in Electrical Engineering, Springer, 2012, pp. 15–34 (cit. on pp. 28, 158).
- [91] **A. Prorok**, C. Cianci, and A. Martinoli, “Towards Optimally Efficient Field Estimation with Threshold-Based Pruning in Real Robotic Sensor Networks”, *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pp. 5453–5459, 2010 (cit. on p. 158).
- [92] **A. Prorok**, W. C. Evans, and A. Martinoli, “An Adaptive Field Estimation Algorithm for Sensor Networks in Dynamic Environments”, *Workshop on Robotics for Environmental Monitoring: International Conference on Intelligent Robots and Systems (IROS)*, 2011 (cit. on p. 159).
- [93] **A. Prorok**, L. Gonon, and A. Martinoli, “Online Model Estimation of Ultra-Wideband TDOA Measurements for Mobile Robot Localization”, *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 807–814, 2012 (cit. on pp. 28, 158).
- [94] **A. Prorok** and A. Martinoli, “A Reciprocal Sampling Algorithm for Lightweight Distributed Multi-Robot Localization”, *Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3241–3247, 2011 (cit. on pp. 28, 158).
- [95] —, “Accurate Indoor Localization with Ultra-Wideband using Spatial Models and Collaboration”, *International Journal of Robotics Research (IJRR)*. To appear, 2013 (cit. on pp. 28, 158).
- [96] —, “Accurate Localization with Ultra-Wideband: Tessellated Spatial Models and Collaboration”, *Proceedings of the 13th International Symposium on Experimental Robotics (ISER), Springer Tracts in Advanced Robotics*. To appear, 2014 (cit. on pp. 29, 158).
- [97] **A. Prorok**, P. Tomé, and A. Martinoli, “Accommodation of NLOS for Ultra-Wideband TDOA Localization in Single- and Multi-Robot Systems”, *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2011. DOI: 10.1109/IPIN.2011.6071927 (cit. on pp. 29, 48, 50, 158).
- [98] **A. Prorok**, N. Correll, and A. Martinoli, “Multi-level Spatial Modeling for Stochastic Distributed Robotic Systems”, *The International Journal of Robotics Research (IJRR)*, vol. 30, pp. 574–589, Apr. 2011 (cit. on p. 158).
- [99] J. Pugh, X. Raemy, C. Favre, R. Falconi, and A. Martinoli, “A Fast On-Board Relative Positioning Module for Multi-Robot Systems”, *IEEE Transactions on Mechatronics*, vol. 14, no. 2, pp. 151–162, 2009 (cit. on pp. 81, 82).
- [100] Y. Qi, “Wireless geolocation in a non-line-of-sight environment”, PhD thesis, Princeton University, 2004 (cit. on p. 46).
- [101] M. Quigley, D. Stavens, A. Coates, and S. Thrun, “Sub-Meter Indoor Localization in Unmodified Environments with Inexpensive Sensors”, *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2039–2046, 2010 (cit. on pp. 18, 73).
- [102] A. N. Raghavan, H. Ananthapadmanaban, M. S. Sivamurugan, and B. Ravindran, “Accurate mobile robot localization in indoor environments using bluetooth”, *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4391–4396, 2010 (cit. on p. 15).
- [103] A. Ranganathan, J. Al-Muhtadi, S. Chetan, R. Campbell, and M. D. Mickunas, *MiddleWhere: a middleware for location awareness in ubiquitous computing applications*. ACM/IFIP/USENIX International Conference on Middleware, 2004 (cit. on p. 4).

- [104] V. Renaudin, B. Merminod, and M. Kasser, "Optimal Data Fusion for Pedestrian Navigation based on UWB and MEMS", *Position Location and Navigation System Conference (PLANS)*, pp. 341–349, 2008 (cit. on p. 72).
- [105] C. Rizos, R. Gethin, J. Barnes, and N. Gambale, "Locata: A new high accuracy indoor positioning system", *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2010. DOI: 10.1109/IPIN.2010.5647717 (cit. on pp. 9, 18).
- [106] J. F. Roberts, T. S. Stirling, J. C. Zufferey, and D. Floreano, "2.5D infrared range and bearing system for collective robotics", *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2009. DOI: 10.1109/IROS.2009.5354263 (cit. on p. 81).
- [107] S. I. Roumeliotis and G. A. Bekey, "Distributed Multirobot Localization", *IEEE Transactions on Robotics and Automation*, vol. 14, no. 5, pp. 781–795, 2002 (cit. on pp. 81, 105).
- [108] N. Roy, W. Burgard, D. Fox, and S. Thrun, "Coastal navigation-mobile robot navigation with uncertainty in dynamic environments", *Proceedings of the 1999 IEEE International Conference on Robotics and Automation*, pp. 35–40, 1999 (cit. on p. 18).
- [109] N. Roy and S. Thrun, "Online self-calibration for mobile robots", *Proceedings of the 1999 IEEE International Conference on Robotics and Automation*, pp. 2292–2297, 1999 (cit. on pp. 19, 23).
- [110] Z. Sahinoglu, S. Gezici, and I. Guvenc, *Ultra-wideband Positioning Systems. Theoretical Limits, Ranging Algorithms, and Protocols*. Cambridge University Press, 2008 (cit. on pp. 34–36).
- [111] C. Savarese, J. M. Rabaey, and J. Beutel, "Location in distributed ad-hoc wireless sensor networks", *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 2037–2040, 2001 (cit. on p. 4).
- [112] M. Segura, H. Hashemi, C. Sisterna, and V. Mut, "Experimental demonstration of self-localized Ultra Wideband indoor mobile robot navigation system", *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2010. DOI: 10.1109/IPIN.2010.5647457 (cit. on pp. 15, 18, 71).
- [113] B. Siciliano and O. Khatib, Eds., *Handbook of Robotics*. Springer, 2008 (cit. on pp. 77, 111, 112).
- [114] R. Siegwart, I. Nourbakhsh, and D. Scaramuzza, *Introduction to Autonomous Mobile Robots*, 2nd edition. The MIT Press, 2011 (cit. on pp. 12–14).
- [115] S. Tekinay, E. Chao, and R. Richton, "Performance Benchmarking for Wireless Location Systems", *IEEE Communications Magazine*, vol. 36, no. 4, pp. 72–76, 1999 (cit. on p. 7).
- [116] S. Thrun, M. Bennewitz, W. Burgard, A. Cremers, F. Dellaert, D. Fox, D. Hähnel, C. Rosenberg, N. Roy, J. Schulte, and D. Schulz, "MINERVA: a second-generation museum tour-guide robot", *Proceedings of the 1999 IEEE International Conference on Robotics and Automation*, pp. 1999–2005, 1999 (cit. on p. 18).
- [117] S. Thrun, W. Burgard, and D. Fox, "A real-time algorithm for mobile robot mapping with applications to multi-robot and 3D mapping", *IEEE International Conference on Robotics and Automation (ICRA)*, vol. 1, 321–328 vol. 1, 2000 (cit. on p. 14).
- [118] —, "A real-time algorithm for mobile robot mapping with applications to multi-robot and 3D mapping", *IEEE International Conference on Robotics and Automation (ICRA)*, vol. 1, 321–328 vol. 1, 2000 (cit. on p. 19).
- [119] —, *Probabilistic Robotics*. The MIT Press, 2005 (cit. on pp. 14, 54, 55, 144).
- [120] S. Thrun, D. Fox, W. Burgard, and F. Dellaert, "Robust Monte Carlo Localization for Mobile Robots", *Artificial Intelligence*, vol. 128, no. 1-2, pp. 99–141, 2000 (cit. on pp. 18, 106).
- [121] S. Thrun and Y. f. Liu, "Multi-robot SLAM with sparse extended information filters", *Robotics Research*, pp. 254–266, 2005 (cit. on p. 127).

## Bibliography

---

- [122] R. Y. Tsai, "A Versatile Camera Calibration Technique for High-accuracy 3D machine Vision Metrology using Off-the-Shelf TV Cameras and Lenses", *IEEE Journal of Robotics and Automation*, vol. 3, no. 4, pp. 323–344, 1987 (cit. on p. 21).
- [123] United Nations, *Population Distribution, Urbanization, Internal Migration and Development: An Internal Perspective*. United Nations, Departement of Economic and Social Affairs, 2011 (cit. on p. 3).
- [124] A. Varshavsky, E. de Lara, J. Hightower, A. LaMarca, and V. Otsason, "GSM indoor localization", *Pervasive and Mobile Computing*, vol. 3, no. 6, pp. 698–720, 2007 (cit. on p. 10).
- [125] A. Waegli, S. Guerrier, and J. Skaloud, "Redundant MEMS-IMU integrated with GPS for Performance Assessment in Sports", *IEEE/ION Position, Location and Navigation Symposium*, pp. 1260–1268, 2008 (cit. on p. 72).
- [126] C. Wang, "Location estimation and uncertainty analysis for mobile robots", *Proceedings of the 1988 IEEE International Conference on Robotics and Automation*, pp. 1231–1235, 1988 (cit. on p. 23).
- [127] J. Werb and C. Lanzl, "Designing a positioning system for finding things and people indoors", *IEEE Spectrum*, vol. 35, no. 9, pp. 71–78, 1998 (cit. on pp. 3, 4).
- [128] J. Werfel, Y. Bar-Yam, D. Rus, and R. Nagpal, "Distributed construction by mobile robots with enhanced building blocks", *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2787–2794, 2006 (cit. on p. 77).
- [129] M. Youssef and A. Agrawala, "The Horus location determination system", *Wireless Networks*, vol. 14, no. 3, pp. 357–374, Jan. 2007 (cit. on p. 10).
- [130] H. Zhan, "Impulse Radio Ultra-WideBand Ranging and Localization for Sensor Networks", PhD thesis, Ecole Polytechnique Fédérale de Lausanne (EPFL), 2009 (cit. on p. 36).
- [131] X. S. Zhou and S. I. Roumeliotis, "Multi-robot SLAM with Unknown Initial Correspondence: The Robot Rendezvous Case", *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1785–1792, 2006 (cit. on p. 127).



# Curriculum Vitae

## Amanda Prorok

### Education

2008-2013	PhD in Computer, Communication and Information Sciences <i>Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland</i>
2001-2006	MS in Computer Science <i>Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland</i>
1994-2001	Swiss Baccalaureate <i>Kantonsschule Zürcher Unterland, Bülach, Switzerland</i>

### Experience

2007	Diplomatic Intern <i>Science &amp; Technology Office, Embassy of Switzerland, Tokyo, Japan</i>
2006	Research Intern <i>Mitsubishi Electric Corporation, Amagasaki, Japan</i>
2004-2005	Teaching Assistant <i>School of Computer and Communication Sciences, EPFL, Switzerland</i>
2004	Research Intern <i>Informatics and Applied Mathematics Department, University of Girona, Spain</i>

### Honors and Awards

2008	Travel award, IEEE Swarm Intelligence Symposium
2006	Scholarship, Swiss-Japanese Chamber of Commerce
2004	Trainee of the year, IAESTE, sponsored by ABB Switzerland

## Publications

### Journal Articles

1. **A. Prorok** and A. Martinoli, “Accurate Indoor Localization with Ultra-Wideband using Spatial Models and Collaboration”, *International Journal of Robotics Research (IJRR)*. To appear, 2013
2. **A. Prorok**, N. Correll, A. Martinoli, “Multi-level Spatial Modeling for Stochastic Distributed Robotic Systems”, *The International Journal of Robotics Research (IJRR)*, vol. 30, pp. 574–589, Apr. 2011

### Refereed Conference Proceedings

1. **A. Prorok** and A. Martinoli, “Accurate Localization with Ultra-Wideband: Tessellated Spatial Models and Collaboration”, *Proceedings of the 13th International Symposium on Experimental Robotics (ISER)*, Springer Tracts in Advanced Robotics. To appear, 2014
2. **A. Prorok**, L. Gonon, A. Martinoli, “Online Model Estimation of Ultra-Wideband TDOA Measurements for Mobile Robot Localization”, *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 807–814, 2012
3. **A. Prorok**, A. Bahr, A. Martinoli, “Low-Cost Collaborative Localization for Large-Scale Multi-Robot Systems”, *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4236–4241, 2012
4. **A. Prorok** and A. Martinoli, “A Reciprocal Sampling Algorithm for Lightweight Distributed Multi-Robot Localization”, *Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3241–3247, 2011
5. **A. Prorok**, P. Tomé, A. Martinoli, “Accommodation of NLOS for Ultra-Wideband TDOA Localization in Single- and Multi-Robot Systems”, *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2011. DOI: 10.1109/IPIN.2011.6071927
6. S. Goyal, **A. Prorok**, A. Martinoli, “Two-Phase Online Calibration for Infrared-Based Inter-Robot Positioning Modules”, *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3313–3319, 2011
7. **A. Prorok**, A. Arfire, A. Bahr, J. R. Farserotu, A. Martinoli, “Indoor Navigation Research with the Khepera III Mobile Robot: An Experimental Baseline with a Case-Study on Ultra-Wideband Positioning”, *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2010. DOI: 10.1109/IPIN.2010.5647880
8. **A. Prorok**, C. Cianci, A. Martinoli, “Towards Optimally Efficient Field Estimation with Threshold-Based Pruning in Real Robotic Sensor Networks”, *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pp. 5453–5459, 2010
9. C. M. Cianci, J. Nembrini, **A. Prorok**, A. Martinoli, “Assembly of Configurations in a Networked Robotic System: A Case Study on a Reconfigurable Interactive Table Lamp”, *IEEE Swarm Intelligence Symposium (SIS)*, 2008. DOI: 10.1109/SIS.2008.4668318

### Book Chapters

1. **A. Prorok**, A. Bahr, A. Martinoli, “Low-Cost Multi-Robot Localization”, in *Redundancy in Robot Manipulators and Multi-Robot Systems*, D. Milutinovic and J. Rosen, Eds., Lecture Notes in Electrical Engineering, Springer, 2012, pp. 15–34
2. G. Mermoud, **A. Prorok**, L. Matthey, C. M. Cianci, N. Correll, A. Martinoli, “Self-Organized Robotic Systems: Large-Scale Experiments in Aggregation and Self-Assembly using Miniature Robots”, in *Handbook of Collective Robotics*, S. Kernbach, Ed., Pan Stanford Publishing, 2012

## Refereed Workshop Proceedings

1. **A. Prorok**, W. C. Evans, A. Martinoli, "An Adaptive Field Estimation Algorithm for Sensor Networks in Dynamic Environments", *Workshop on Robotics for Environmental Monitoring; International Conference on Intelligent Robots and Systems (IROS)*, 2011

## Invited Talks

1. "Towards Better Indoor Localization for Large-Scale Systems: Ultra-Wideband and Collaborative Strategies", CSAIL, MIT, Cambridge, USA, May 2012
2. "Any-Time Low-Cost Localization for Large-Scale Multi-Robots Systems", Workshop on Redundancy in Robot Manipulators and Multi-Robot Systems, IROS, San Francisco, USA, October 2011
3. "An Adaptive Field Estimation Algorithm for Sensor Networks in Dynamic Environments", Workshop on Robotics for Environmental Monitoring, IROS, San Francisco, USA, October 2011
4. "Accommodation of NLOS for Ultra-Wideband TDOA Localization in Single- and Multi-Robot Systems", NCCR Mobile Information & Communication Systems, Scientific Workshop, ETH, Zurich, Switzerland, September 2011

## Project Supervision

1. Christophe Schild, Master's Thesis (2012)  
Development of Microphone and Speaker Models in Webots using OpenAL
2. Rafael Mosberger, Master's Thesis (2012)  
Vision-based People Detection Using Infrared Reflective Vests
3. Lukas Gonon, Summer Internship (2011)  
An Efficient Ultra-Wideband Time-Difference-Of-Arrival Measurement Model
4. Dimitrios Chalikopoulos, Semester Project (2011)  
Towards a Spatial Model of Ultra-Wideband Ranging Biases
5. Rafael Mosberger, Semester Project (2010)  
Collective Robot Localization: Exploring Information Merging and Sensor Fusion Strategies
6. Louis Raymond, Summer Internship (2010)  
Dynamic Object Tracking for Mobile Robot Localization
7. Adrian Arfire, Summer Internship (2009)  
Towards Ultra-Wideband Technology for Mobile Robot Navigation: Algorithms and Performance Evaluation using an Overhead Camera
8. Yuangfang Ma, Summer Internship (2009)  
Implementation of TinyOS-2 compliant Firmware for a Robotic Communication Module

## Academic Service (Reviews)

1. International Journal of Robotics Research (IJRR), 2012-2013
2. IEEE Transactions on Evolutionary Computation, 2009
3. IEEE Communication Letters, 2012
4. Electronics and Telecommunications Research Institute Journal (ETRI), 2013

5. Symposium on Distributed Autonomous Robotic Systems (DARS), 2008-2012
6. Swarm Intelligence Symposium (SIS), 2008
7. International Conference on Autonomous Agents and Multiagent Systems (AAMAS), 2009-2010
8. Robotics: Science and Systems Conference (RSS), 2009
9. Swarm Intelligence Journal (SIJ), 2009
10. International Conference on Intelligent Robots and Systems (IROS), 2011-2012
11. International Conference on Robotics and Automation (ICRA), 2012-2013
12. International Conference on Indoor Positioning and Navigation (IPIN), 2010

## Languages

English	native
German	native
French	excellent
Japanese	fluent, certified (Japanese Language Proficiency Test, N2)
Portuguese	conversational
Spanish	conversational
Russian	elementary

## Personal Details

Date of birth:	21 January 1982
Place of birth:	São Paulo, Brazil
Citizenship:	Great Britain, Switzerland