

# Accurate Indoor Localization with Ultra-Wideband using Spatial Models and Collaboration

Amanda Prorok and Alcherio Martinoli

Distributed Intelligent Systems and Algorithms Laboratory,  
School of Architecture, Civil and Environmental Engineering,  
Ecole Polytechnique Fédérale de Lausanne, Switzerland  
amanda.prorok@epfl.ch, alcherio.martinoli@epfl.ch

July 9, 2013

## Abstract

Ultra-wideband (UWB) localization is a recent technology that performs competitively with many indoor localization methods currently available. Despite its desirable traits, such as potential high accuracy and high material penetrability, the resolution of non-line-of-sight (NLOS) signals remains a very hard problem and has a significant impact on the localization performance. In this work, we address the peculiarities of UWB error behavior by building models that capture the spatiality as well as the multimodal statistics of the error behavior. Our framework utilizes tessellated maps that associate probabilistic error models to localities in space. In addition to our UWB localization strategy (which provides absolute position estimates), we investigate the effects of collaboration in the form of relative positioning. To this means, we develop a relative range and bearing model, and, together with the UWB model, present a unified localization technique based on a particle filter framework. We test our approach experimentally on a group of ten mobile robots equipped with UWB emitters and extension modules providing inter-robot relative range and bearing measurements. Our experimental insights highlight the benefits of collaboration, which are consistent over numerous experimental scenarios. Also, we show the relevance, in terms of positioning accuracy, of our multimodal UWB measurement model by performing systematic comparisons with two alternative measurement models. Our final results show median localization errors below 10 cm in cluttered environments, using a modest set of 50 particles in our filter.

## 1 Introduction

Due to its large frequency spectrum, UWB is able to penetrate through objects in NLOS scenarios, and thus alleviates the LOS constraint imposed by other sensor types relying on media such as infrared, ultrasound, visible light or narrow-band radio. This advantage ultimately enables localization over large ranges and in dynamic environments, which makes UWB an attractive candidate for indoor applications such as asset management, inventory tracking and assembly control, for a variety of different industries (Liu et al., 2007; Sahinoglu et al., 2008). Nevertheless, NLOS scenarios may cause biases in the signal propagation times, which leads to significant localization errors. In order to guarantee reliable and accurate performance, these biases need to be addressed by an effective localization strategy.

In this paper, we consider the problem of absolute localization of a team of mobile robots for unknown initial conditions. Since we aim for a portable, miniaturizable, cost-effective solution, with centimeter-level accuracy, we choose to solve the localization problem with a hybrid approach that combines UWB localization with collaborative localization. Our UWB localization strategy uses time-difference-of-arrival (TDOA) measurements from two or more base station pairs. We address the peculiarities of

UWB signal propagation with models that capture the spatiality as well as the multimodality 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. 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. This work investigates the combined roles of collaborative localization and UWB localization. Our experiments are able to validate our overall localization strategy, and show that the performance can be significantly improved when using collaboration: our insights indicate that relative positioning—even if through noisy sensors—is a useful tool to reduce absolute localization errors. Since the fusion of UWB positioning sensors with exteroceptive sensors has hardly been considered so far, our studies present pioneering work in this domain.

## 1.1 Related Work

Since our approach combines methods from two separate subdomains of localization research, we summarize the respective literature separately, in the two following subsections.

### 1.1.1 UWB Localization

UWB is a radio technology which is characterized by its very large bandwidth compared to conventional narrowband systems, and in particular features high positioning accuracy (due to a time resolution in the order of nanoseconds), and high material penetrability (due to a bandwidth typically larger than 0.5 GHz). Despite these desirable traits, the resolution of multipath signals remains a hard problem—the complexity of implementing state-of-the-art direct signal path detection algorithms is exacerbated by the necessity of maintaining very high sampling rates, in the order of several GHz (Sahinoglu et al., 2008).

Recently, UWB has received some attention within the robotics community. Hollinger et al. (2012) 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. (2009) 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 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. (2005), 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. (2010) 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 via constrained least squares minimization of the TDOA positioning equations, thus, without modeling the UWB error 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.

Indeed, when using UWB for localization, the basic measured quantity is time-of-flight. In practice, however, time-of-arrival (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 time-difference-of-arrival (TDOA) systems which are significantly more practical, since only the synchronization among base stations is required. This, in turn, enables a significant miniaturization of the emitter boards (to an order of a few centimeters in size), as well as a reduction of the consumption power (to an order of  $\mu\text{W}$  (Mercier et al., 2008)).

### 1.1.2 Collaborative Localization

There is abundant literature discussing various strategies toward solving the multi-robot localization problem. Our approach distinguishes itself by respecting the following design goals: cost-efficiency, full decentralization and scalability (Prorok and Martinoli, 2011; Prorok et al., 2012c). Our work relates to the body of literature that takes a *decentralized* approach to the multi-robot localization problem, where a robot maintains an estimate of only its own pose, versus a full-system state estimate (including all robots' individual pose estimates). We relate to the latter as a *multi-centralized* approach (Prorok et al., 2012b; Nerurkar and Roumeliotis, 2010)).

The category of work representing the *decentralized* approach has the following 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. (2000) first introduced a multi-robot Monte-Carlo localization algorithm for *global* localization, that also relaxes noise assumptions as well as inter-robot dependencies. They propose a method in 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. (2009a) 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 exchange rates. This method, however, allows cyclic updates and, thus, may suffer from overconfidence. In an addition to this work (Bahr et al., 2009b), 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).

## 1.2 Contributions of this Paper

This work is a culmination of our ongoing research efforts in the domain of indoor localization, and combines two complementary components: an UWB system capable of absolute positioning (Prorok et al., 2012c, 2011), and a collaborative multi-robot system capable of relative observations (Prorok and Martinoli, 2011; Prorok et al., 2012a). The current paper goes beyond our previous works on various fronts. In particular, to the best of our knowledge, this work is among the first to perform the fusion of UWB with on-board exteroceptive sensors—in our case, infrared-based relative positioning sensors. Our contributions are enumerated as follows.

- We develop a compact, closed-form TDOA measurement model that is a mixture of Gaussian and log-normal distributions. To the best of our knowledge, our model is the first UWB TDOA measurement model for mobile robot localization.
- The closed-form TDOA measurement model enables the development of an efficient estimation algorithm based on Expectation Maximization (EM) to determine the values of the model parameters. We show how the algorithm is applied in a batch mode as well as in an online mode, and evaluate the performance of the estimation algorithm quantitatively to show that it performs very well in comparison with an alternative, standard optimization method.
- We develop a mapping technique that allows us to tessellate space into discrete areas, and where each area is associated with a unique UWB measurement model. We exploit the ability of our model

to capture and adapt to all types of UWB propagation, and spatially customize the parametrization of this model using the underlying tessellation.

- We present an algorithm that configures the map of UWB measurement models in real-time, as a function of incoming data.
- We test our multimodal UWB measurement model against two variant models (a Gaussian model and a discrete histogram model) and discuss the advantages and disadvantages of these model alternatives with respect to practicality and performance.
- We review the formalisms of our previous works to present a new, unified framework that combines UWB and collaborative localization methods.
- We test our approach on a team of ten robots in a room-sized setup including obstacles that induce significant NLOS signal propagation. Our evaluations test the effect of varying collaboration frequencies, in order to better understand the impact of relative positioning on the overall localization accuracy.

The following text will elaborate the details of our contributions. After presenting the general framework, we first present our work on UWB localization (Sections 3—5). Subsequently, we present our collaborative localization approach (Section 6), as well as our final localization algorithm (Algorithm 5) that fuses the information from the two regarded sensing modalities. We conclude this paper by discussing the limitations of our approach and by bringing our work into a more general perspective.

## 2 Problem Statement

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, we develop an algorithm that fuses UWB TDOA measurements and relative positioning measurements with dead-reckoning information. The following paragraphs state our system and outline the notation for our two core elements, UWB positioning and relative positioning, by introducing probability densities  $p$  and  $q$  that implement the measurement models for the respective sensor modalities. The first goal of this paper is to develop  $p$  and  $q$ , and to detail how these models are practically employed on real robots. Subsequently, our goal is to detail our mapping method and to present an algorithm that, when deployed on robots, builds UWB measurement model maps in real-time. Finally, our goal is to show how the models for our two sensor modalities are embedded in an efficient localization filter, and to test the resulting performance through extensive experimental work.

### 2.1 General Framework and Performance Metric

Our multi-robot system is composed of  $N_{\mathcal{R}}$  robots  $\mathcal{R}_1, \mathcal{R}_2, \dots, \mathcal{R}_{N_{\mathcal{R}}}$ . Given its efficiency in solving the localization problem for unknown initial conditions and its ability to accommodate arbitrary probability density functions, our method of choice is the particle filter (otherwise known as Monte Carlo Localization (Thrun et al., 2005)). Indeed, as we will see in Sections 3 and 6, the model describing the relative range and bearing measurements is non-linear, and the model describing UWB measurements is both non-linear and multimodal. Hence, each robot runs an individual instance of the filter that keeps track of the current position estimate. The belief of a robot’s pose is formulated as

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

where  $M$  is the number of particles,  $\mathbf{x}_{n,t}^{[i]}$  is a sample of the random variable  $\mathbf{x}_{n,t} = [x_{n,t}, y_{n,t}, \psi_{n,t}]^T$ , the state of the robot  $\mathcal{R}_n$ , (where  $x_{n,t}$  and  $y_{n,t}$  are Euclidian 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  $\mathcal{R}_n$ . In order to discuss the localization performance in terms of the absolute positioning error (distance to 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  $\mathbf{x}_{n,t}$ :

$$\mathcal{E} = \left\| \left( \frac{1}{M} \sum_i^M \mathbf{x}_{n,t}^{[i]} \right) - \mathbf{x}_{n,t} \right\|. \quad (2)$$

## 2.2 Localization with UWB TDOA Measurements

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 1(a). 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:

$$\Delta\tau_{uv,n,t} \triangleq \Delta\tau_{uv}(\hat{\tau}_{uv,n,t}, \mathbf{x}_{n,t}) = \hat{\tau}_{uv,n,t} - \tau_{uv}(\mathbf{x}_{n,t}), \quad (3)$$

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  $N_A$  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 1(b). 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 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  $\Theta$ :

$$\begin{aligned} m_{uv}(\mathbf{x}_{n,t}) &= \boldsymbol{\theta}_{uv,a}, \text{ s.t.} \\ \exists \langle A_a, \boldsymbol{\theta}_{uv,a} \rangle &\in \mathcal{M}_{uv}, \text{ with } \mathbf{x}_{n,t} \in A_a. \end{aligned} \quad (4)$$

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  $\boldsymbol{\theta}_{uv,a}$ . We define our error model as

$$p(\Delta\tau_{uv,n,t}; \boldsymbol{\theta}_{uv,a}) = p(\Delta\tau_{uv,n,t}; m_{uv}(\mathbf{x}_{n,t})). \quad (5)$$

This concludes our problem formulation for UWB localization. The development of  $p$  is elaborated in detail in Section 3.

## 2.3 Localization with Relative Range and Bearing Measurements

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 our 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 1(a) 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

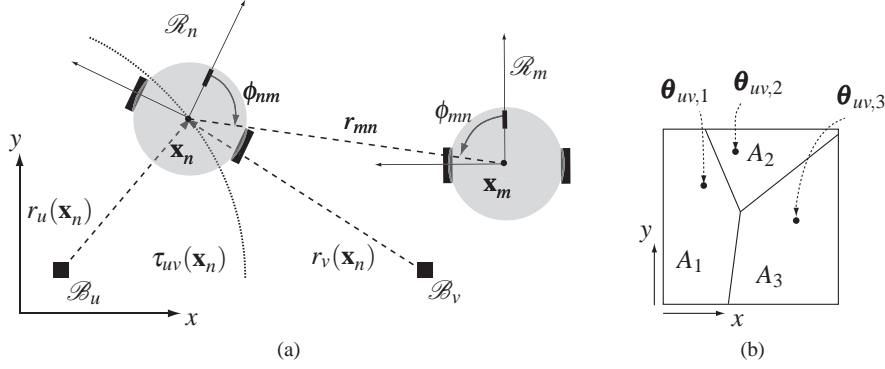


Figure 1: (a) System of  $N_{\mathcal{R}} = 2$  robots at positions  $\mathbf{x}_n, \mathbf{x}_m$  and two well-localized UWB base stations  $\mathcal{B}_u$  and  $\mathcal{B}_v$ . The figure shows the true ranges  $r_u(\mathbf{x}_n)$  and  $r_v(\mathbf{x}_n)$  of robot  $\mathcal{R}_n$  to the respective base stations, as well as a segment of the hyperbola resulting from the range-difference  $\tau_{uv,n}$ . 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  $p(\Delta\tau_{uv,n,t}; \boldsymbol{\theta}_{uv,a})$  are mapped to individual areas  $A_a$ .

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  $\hat{\phi}_{mn,t}$

$$\begin{aligned} \Delta r(\hat{r}_{mn,t}, \mathbf{x}_{m,t}, \mathbf{x}_{n,t}) &= \hat{r}_{mn,t} - r(\mathbf{x}_{m,t}, \mathbf{x}_{n,t}) \\ \Delta \phi(\hat{\phi}_{mn,t}, \mathbf{x}_{m,t}, \mathbf{x}_{n,t}) &= \hat{\phi}_{mn,t} - \phi(\mathbf{x}_{m,t}, \mathbf{x}_{n,t}), \end{aligned} \quad (6)$$

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:

$$\begin{bmatrix} r_{mn,t} \\ \phi_{mn,t} \end{bmatrix} \triangleq \mathbf{T}_e^p(\mathbf{x}_m, \mathbf{x}_n) = \begin{bmatrix} r(\mathbf{x}_{m,t}, \mathbf{x}_{n,t}) \\ \phi(\mathbf{x}_{m,t}, \mathbf{x}_{n,t}) \end{bmatrix} = \begin{bmatrix} \sqrt{(x_n - x_m)^2 + (y_n - y_m)^2} \\ \text{atan2}((y_n - y_m), (x_n - x_m)) - \Psi_m \end{bmatrix}. \quad (7)$$

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

$$\Delta r_{mn,t}^{[j]} \triangleq \Delta r(\hat{r}_{mn,t}, \mathbf{x}_m^{[j]}, \mathbf{x}_{n,t}) \quad (8)$$

$$\Delta \phi_{mn,t}^{[j]} \triangleq \Delta \phi(\hat{\phi}_{mn,t}, \mathbf{x}_m^{[j]}, \mathbf{x}_{n,t}). \quad (9)$$

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

$$q^{[j]}(\Delta r_{mn,t}^{[j]}, \Delta \phi_{mn,t}^{[j]}; \xi). \quad (10)$$

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

$$q(\Delta r_{mn,t}, \Delta \phi_{mn,t}; \xi) = \eta \cdot \sum_{\langle \mathbf{x}_m^{[j]}, w_{m,t}^{[j]} \rangle \in \mathcal{X}_{m,t}} q^{[j]}(\Delta r_{mn,t}^{[j]}, \Delta \phi_{mn,t}^{[j]}; \xi) \cdot w_{m,t}^{[j]} \quad (11)$$



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  $\mathcal{R}_m$ . This concludes our problem formulation for collaborative localization. The development of  $q$  is elaborated in detail in Section 6.1.

### 3 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.

#### 3.1 General UWB TDOA Measurement Model

For the sake of brevity, our following derivations omit the subscript  $t$ . We begin by detailing a general model  $p^*(\Delta \tau_{uv,n}; \cdot)$ , which, however, is analytically non-tractable. The subsequent subsection develops an approximation to this general model, producing the closed-form model  $p(\Delta \tau_{uv,n,t}; \boldsymbol{\theta}_{uv,a})$  (introduced in Section 2.2). 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

$$\hat{r}_{u,n} \triangleq r_u(\mathbf{x}_n) + \varepsilon + Yb_u \quad (12)$$

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  $\varepsilon \sim p_{\mathcal{N}}(0, \sigma_{\mathcal{N}}^2)$  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  $(0, \infty)$  (Alsindi et al., 2009; Qi, 2004). In particular, Alsindi et al. (2009) 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  $b_u \sim p_{\ln \mathcal{N}}(\mu_u, \sigma_u)$ , 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

$$\Delta r_{u,n} \triangleq \hat{r}_{u,n} - r_u(\mathbf{x}_n) \quad (13)$$

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

$$p_u(\Delta r_{u,n} | \bar{L}_u) = (p_{\ln \mathcal{N}, u} * p_{\mathcal{N}})(\Delta r_{u,n}) \quad (14)$$

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 density of an error

<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).

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

$$p_u(\Delta r_{u,n}|L_u) = p_{\mathcal{N}}(\Delta r_u) \quad (15)$$

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

$$p_u(\Delta r_{u,n}) = p_u(\Delta r_u|L_u) \cdot P_{L_u} + p_u(\Delta r_u|\bar{L}_u) \cdot (1 - P_{L_u}). \quad (16)$$

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

$$\hat{\tau}_{uv,n} \triangleq \hat{r}_{u,n} - \hat{r}_{v,n} \quad (17)$$

and then easily model the TDOA error  $\Delta \tau_{uv}$  as previously shown in Equation (3). 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 (12), resulting in

$$\Delta \tau_{uv,n} = \Delta r_{u,n} - \Delta r_{v,n}. \quad (18)$$

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 random variables drawn from the probability densities describing the TOA error models of the two respective base stations. We use the results of Equations (16) and (18) to model this resulting probability density as

$$p^*(\Delta \tau_{uv,n}) \triangleq (p_u * p_v^-)(\Delta \tau_{uv,n}) \quad (19)$$

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.,  $p_v^-(\Delta r_{v,n}) = p_v(-\Delta r_{v,n})$ ).

### 3.2 Efficient TDOA Measurement Model

Although numerical implementations for the TDOA measurement model of Equation (19) 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 (19).

Using basic algebraic properties of the convolution and inserting Equation (16) into Equation (19), we have

$$\begin{aligned} p^*(\Delta \tau_{uv,n}) = & \left( P_{L_u} P_{L_v} (p_{\mathcal{N}} * p_{\mathcal{N}}^-) + P_{L_u} (1 - P_{L_v}) (p_{\mathcal{N}} * p_{\mathcal{N}}^- * p_{\ln \mathcal{N}, v}^-) + \right. \\ & P_{L_v} (1 - P_{L_u}) (p_{\mathcal{N}} * p_{\mathcal{N}}^- * p_{\ln \mathcal{N}, u}) + \\ & \left. (1 - P_{L_u}) (1 - P_{L_v}) (p_{\mathcal{N}} * p_{\mathcal{N}}^- * p_{\ln \mathcal{N}, u} * p_{\ln \mathcal{N}, v}^-) \right) (\Delta \hat{\tau}_{uv,n}). \end{aligned} \quad (20)$$



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 (20) can be rewritten as

$$p^*(\Delta\tau_{uv,n}) = \left( P_{L_u} P_{L_v} p_{\sqrt{2}\mathcal{N}} + P_{L_u} (1 - P_{L_v}) (p_{\sqrt{2}\mathcal{N}} * p_{\ln\mathcal{N},v}^-) + P_{L_v} (1 - P_{L_u}) (p_{\sqrt{2}\mathcal{N}} * p_{\ln\mathcal{N},u}) + (1 - P_{L_u}) (1 - P_{L_v}) (p_{\sqrt{2}\mathcal{N}} * p_{\ln\mathcal{N},u} * p_{\ln\mathcal{N},v}^-) \right) (\Delta\tau_{uv,n}). \quad (21)$$

UWB measurement campaigns have shown that  $\sigma_{\mathcal{N}} \ll 1$  (Alsindi et al., 2009; Prorok et al., 2011). 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 was numerically verified in (Prorok et al., 2012c),  $p_{\ln\mathcal{N},u} * p_{\ln\mathcal{N},v}^-$  can be approximated by the density function  $p_{\mathcal{N}}$  of a normal distribution  $\mathcal{N}(\tilde{\mu}, \tilde{\sigma}^2)$ , under reasonable assumptions for the parameter ranges. 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  $\tilde{\mu} = E[X_u - X_v]$  and  $\tilde{\sigma}^2 = \text{Var}(X_u - X_v)$ . This leads to

$$\begin{aligned} \tilde{\mu} &= E[X_u - X_v] = e^{\mu_u + \sigma_u^2/2} - e^{\mu_v + \sigma_v^2/2} \\ \tilde{\sigma}^2 &= \text{Var}(X_u) + \text{Var}(-X_v) = e^{2\mu_u + \sigma_u^2} (e^{\sigma_u^2} - 1) + e^{2\mu_v + \sigma_v^2} (e^{\sigma_v^2} - 1). \end{aligned} \quad (22)$$

Finally, using the results obtained above, we further simplify Equation (21) and redefine the density  $p^*$ , our TDOA error model, in closed-form as a sum of four terms:

$$p(\Delta\tau_{uv,n}; \boldsymbol{\theta}_{uv,a}) = \left( P_{L_u} P_{L_v} p_{\sqrt{2}\mathcal{N}} + P_{L_u} (1 - P_{L_v}) p_{\ln\mathcal{N},v}^- + P_{L_v} (1 - P_{L_u}) p_{\ln\mathcal{N},u} + (1 - P_{L_u}) (1 - P_{L_v}) p_{\mathcal{N}} \right) (\Delta\tau_{uv,n}) \quad (23)$$

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

$$\boldsymbol{\theta}_{uv,a} = [\mu_u, \sigma_u, \mu_v, \sigma_v, P_{L_u}, P_{L_v}]^\top$$

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 (22) to complete the model (23). We note that in the final form of our model (23), 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 2 provides an intuition of the multimodal nature of TDOA error data, and illustrates how our error model proposes to capture this. In particular, Figure 2(b) illustrates how each of the four terms of Equation (23) represents one of the four possible modes of operation for a base station pair.

### 3.3 Localization using UWB Measurements

The previous section established the UWB error model  $p(\Delta\tau_{uv,n,t}; \boldsymbol{\theta}_{uv,a})$ . The reader will recall from Section 2.2 that, in addition to a UWB measurement model, our modeling approach includes a set of maps  $\mathcal{M}$ . 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 (4) to retrieve the parameter vector  $\boldsymbol{\theta}_{uv,a}$  used to formulate the probability density function  $p(\Delta\tau_{uv,n,t}; \boldsymbol{\theta}_{uv,a})$ . 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

$$P(\mathbf{x}_{n,t}^{[i]}) = \eta \cdot \prod_{\langle \hat{\tau}_{uv,n,t}, \langle \mathcal{B}_u, \mathcal{B}_v \rangle \rangle \in T_{n,t}} p(\Delta\tau_{uv}(\hat{\tau}_{uv,n,t}, \mathbf{x}_{n,t}^{[i]}; m_{uv}(\mathbf{x}_{n,t}^{[i]})), \quad (24)$$

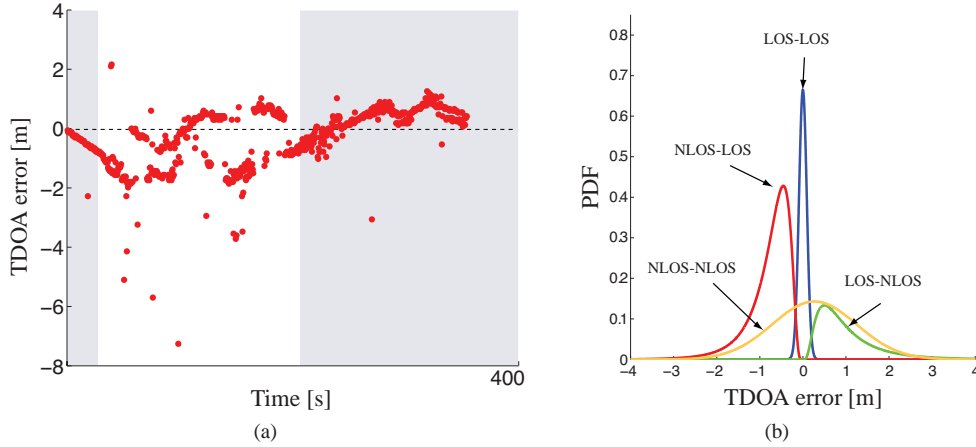


Figure 2: (a) Example of the multimodal nature of TDOA error data. The data was collected by a robot driving through an indoor environment with obstacles. The data points in the white (non-shaded) area exhibit a multimodal behavior. (b) We consider a base station pair  $\langle \mathcal{B}_1, \mathcal{B}_2 \rangle$ . The plot illustrates the four modes which form the complete multimodal probability density function shown in Eq. (23). The model parameters are set to:  $\mu_1 = -0.43$ ,  $\mu_2 = -0.2$ ,  $\sigma_1 = 0.6$ ,  $\sigma_2 = 0.7$ ,  $P_{L_1} = 0.3$ ,  $P_{L_2} = 0.5$ .

where  $\eta$  is a normalization constant. This likelihood function is formulated algorithmically in Algorithm 1, and is embedded in the overall localization filter as seen later in Algorithm 5, on line 4. In practice, the application of the 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 5.

---

**Algorithm 1** 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  $w_{n,t}^{[i]}$
- 

## 4 Estimation of Model Parameters

Due to its efficiency in achieving the Cramér-Rao lower bound 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.

## 4.1 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

$$\hat{\boldsymbol{\theta}}_{uv} = \arg \max_{\boldsymbol{\theta}_{uv,a}} \frac{1}{K} \sum_{k=1}^K \log p(\Delta\tau_{uv,k}; \boldsymbol{\theta}_{uv,a}). \quad (25)$$

First, let us simplify our model (23) by reformulating its four terms as

$$\begin{aligned} f_1(x; \boldsymbol{\theta}_{uv,a}) &= p_{\sqrt{2}\mathcal{N}}(x; \boldsymbol{\theta}_{uv,a}) = p_{\sqrt{2}\mathcal{N}}(x) \\ f_2(x; \boldsymbol{\theta}_{uv,a}) &= p_{\text{In}\mathcal{N},v}^-(x; \boldsymbol{\theta}_{uv,a}) = p_{\text{In}\mathcal{N},v}^-(x; \boldsymbol{\mu}_v, \boldsymbol{\sigma}_v) \\ f_3(x; \boldsymbol{\theta}_{uv,a}) &= p_{\text{In}\mathcal{N},u}(x; \boldsymbol{\theta}_{uv,a}) = p_{\text{In}\mathcal{N},u}(x; \boldsymbol{\mu}_u, \boldsymbol{\sigma}_u) \\ f_4(x; \boldsymbol{\theta}_{uv,a}) &= p_{\tilde{\mathcal{N}}}(x; \boldsymbol{\theta}_{uv,a}) = p_{\tilde{\mathcal{N}}}(x; \tilde{\boldsymbol{\mu}}, \tilde{\boldsymbol{\sigma}}). \end{aligned} \quad (26)$$

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

$$p(x; \boldsymbol{\theta}_{uv,a}) = \sum_{j=1}^4 \alpha_j f_j(x; \boldsymbol{\theta}_{uv,a}). \quad (27)$$

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

$$f(x, z; \boldsymbol{\theta}_{uv,a}) = \sum_{j=1}^4 \alpha_j \delta_{zj} f_j(x; \boldsymbol{\theta}_{uv,a}) \quad (28)$$

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  $\boldsymbol{\theta}_{uv,a}$ . The following notations follow the conventions introduced by Cappé and Moulines (2009). We are now able to write  $f$  in exponential family form:

$$f(x, z; \boldsymbol{\theta}_{uv,a}) = h(x, z) \exp \left( \sum_{j=1}^{11} S_j(x, z) \phi_j(\boldsymbol{\theta}_{uv,a}) \right) \quad (29)$$

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:

$$\begin{aligned}
S(x, z) &= [\delta_{z1}, x^2 \delta_{z1}, \chi_{(-\infty, 0)}(x) \delta_{z2}, \chi_{(-\infty, 0)}(x) \log(-x) \delta_{z2}, \\
&\quad \chi_{(-\infty, 0)}(x) \log(-x)^2 \delta_{z2}, \chi_{(0, \infty)}(x) \delta_{z3}, \chi_{(0, \infty)}(x) \log(x) \delta_{z3}, \\
&\quad \chi_{(0, \infty)}(x) \log(x)^2 \delta_{z3}, \delta_{z4}, x \delta_{z4}, x^2 \delta_{z4}]^\top \\
[\phi_1, \phi_2]^\top(\boldsymbol{\theta}_{uv,a}) &= \left[ \log(\alpha_1) - \log(2\sqrt{\pi}\sigma), -\frac{1}{4\sigma^2} \right]^\top \\
[\phi_3, \phi_4, \phi_5]^\top(\boldsymbol{\theta}_{uv,a}) &= \left[ -\log(2\sqrt{\pi}\sigma_v) + \log(\alpha_2) - \frac{\mu_v^2}{2\sigma_v^2}, -1 + \frac{\mu_v}{\sigma_v^2}, -\frac{1}{2\sigma_v^2} \right]^\top \\
[\phi_6, \phi_7, \phi_8]^\top(\boldsymbol{\theta}_{uv,a}) &= \left[ -\log(2\sqrt{\pi}\sigma_u) + \log(\alpha_3) - \frac{\mu_u^2}{2\sigma_u^2}, -1 + \frac{\mu_u}{\sigma_u^2}, -\frac{1}{2\sigma_u^2} \right]^\top \\
[\phi_9, \phi_{10}, \phi_{11}]^\top(\boldsymbol{\theta}_{uv,a}) &= \left[ -\log(2\sqrt{\pi}\tilde{\sigma}) + \log(\alpha_4) - \frac{\tilde{\mu}^2}{2\tilde{\sigma}^2}, \frac{\tilde{\mu}}{\tilde{\sigma}^2}, -\frac{1}{2\tilde{\sigma}^2} \right]^\top
\end{aligned} \tag{30}$$

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

$$\bar{w}_j(x; \boldsymbol{\theta}_{uv,a}) := \frac{\alpha_j f_j(x; \boldsymbol{\theta}_{uv,a})}{\sum_{m=1}^4 \alpha_m f_m(x; \boldsymbol{\theta}_{uv,a})} \tag{31}$$

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

$$\begin{aligned}
\bar{s}(x; \boldsymbol{\theta}_{uv,a}) &= [\bar{w}_1(x; \boldsymbol{\theta}_{uv,a}), \bar{w}_1(x; \boldsymbol{\theta}_{uv,a})x^2, \bar{w}_2(x; \boldsymbol{\theta}_{uv,a}), \\
&\quad \bar{w}_2(x; \boldsymbol{\theta}_{uv,a}) \log(-x), \bar{w}_2(x; \boldsymbol{\theta}_{uv,a}) \log(-x)^2, \\
&\quad \bar{w}_3(x; \boldsymbol{\theta}_{uv,a}), \bar{w}_3(x; \boldsymbol{\theta}_{uv,a}) \log(x), \bar{w}_3(x; \boldsymbol{\theta}_{uv,a}) \log(x)^2, \\
&\quad \bar{w}_4(x; \boldsymbol{\theta}_{uv,a}), \bar{w}_4(x; \boldsymbol{\theta}_{uv,a})x, \bar{w}_4(x; \boldsymbol{\theta}_{uv,a})x^2]^\top.
\end{aligned} \tag{32}$$

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

$$\begin{aligned}
\bar{\boldsymbol{\theta}}_{uv,a}(s) &= \left[ \frac{s_7}{s_6}, \sqrt{\frac{-s_7^2 + s_6 s_8}{s_6^2}}, \frac{s_4}{s_3}, \sqrt{\frac{-s_4^2 + s_3 s_5}{s_3^2}}, \frac{s_{10}}{s_9}, \sqrt{\frac{-s_{10}^2 + s_9 s_{11}}{s_9^2}}, \right. \\
&\quad \left. \frac{s_1 + s_3}{s_1 + s_3 + s_6 + s_9}, \frac{s_1 + s_6}{s_1 + s_3 + s_6 + s_9} \right]^\top.
\end{aligned} \tag{33}$$

We note that Equations (31) and (32) relate to the E-step and that Equation (33) 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.

## 4.2 Batch Estimation Algorithm

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

$$\hat{\boldsymbol{\theta}}_{uv,a}^{(k+1)} = \bar{\boldsymbol{\theta}} \left( \frac{1}{N} \sum_{i=1}^N \bar{s}(\Delta\tau_{uv}^{(i)}; \hat{\boldsymbol{\theta}}_{uv,a}^{(k)}) \right). \tag{34}$$

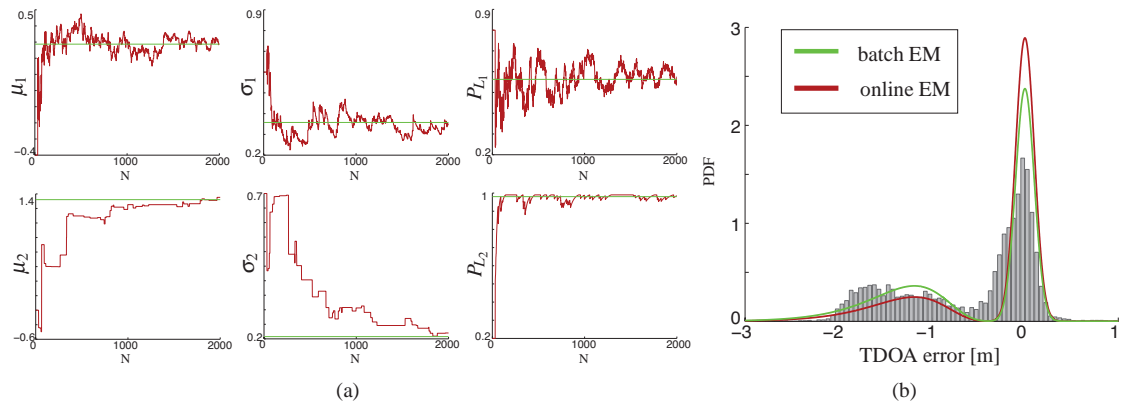


Figure 3: (a) Online EM estimation of model parameters for a real base station pair  $\langle \mathcal{B}_2, \mathcal{B}_1 \rangle$ , for a number of  $N = 2000$  data points sampled in a  $1\text{m} \times 1\text{m}$  area  $A_1$  in our experimental setup, and  $\gamma = 1/k^{0.65}$ . The batch estimates (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  $\bar{\mu}$  and  $\bar{\sigma}$  (not shown here) are defined as in Eq. (22) to form the full parameter vector  $\theta_{12,1}$ . (b) Normalized histogram of the data points. We superimpose the estimated models (online and batch).

### 4.3 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 and Moulines (2009), 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

$$\begin{aligned} \hat{s}^{(k+1)} &= \hat{s}^{(k)} + \gamma^{(k+1)} \left[ \bar{s} \left( \Delta\tau_{uv,k}; \hat{\theta}_{uv,a}^{(k)} \right) - \hat{s}^{(k)} \right] \\ \hat{\theta}_{uv,a}^{(k+1)} &= \bar{\theta}_{uv,a}(\hat{s}^{(k+1)}) \end{aligned} \quad (35)$$

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 Cappé and Moulines (2009) 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 2, where line 6 corresponds to the application of Equation (35). The real-time estimation of UWB error maps constitutes an example application of the online EM algorithm.

### 4.4 Performance

We perform several tests to assess the efficiency of our estimation algorithms. In order to do this, we will assume that any set of true UWB model parameter values lies in the ranges reported in Table 1 (an assumption supported by the experimental measurement campaigns performed by Alsindi et al. (2009) and Prorok et al. (2011)). 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

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

Table 1: Model parameter ranges for a base station  $\mathcal{B}_u$

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, the existence of such parameter ranges facilitates the task of calibrating our models.

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 3(a) 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 3(b) 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 4 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 1, and used in the non-approximated model, Equation (19), 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 4(b) 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 1, and draw a sample size of  $N = 1000$ . Figure 5 shows the comparison of our method with a Sequential Quadratic Programming (SQP) algorithm, evaluated over 100 simulations. Figure 5(a) shows that, for an equal number of maximum iterations, the batch estimation algorithm produces significantly lower Kolmogorov-Smirnov distance values. Moreover, Figure 5(b) shows that running the batch EM is significantly faster than running the SQP algorithm.

## 5 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 2.2), 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. Each area is associated to an error model, which is initialized with parameters randomly sampled from the ranges reported in Table 1. The models can be calibrated using either our online method or our batch method.

### 5.1 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 model in the state when the robot last exited



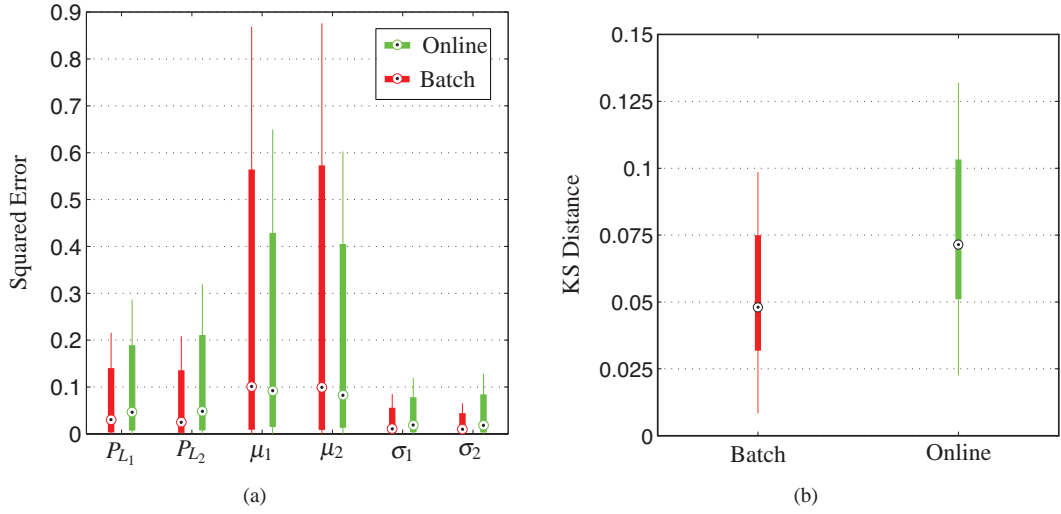


Figure 4: Comparison of batch and online EM estimation algorithms, evaluated over 1000 simulations. For the online algorithm, the final estimate of  $\hat{\theta}_{w,a}$  is found by averaging the last 50 estimates (a) Squared-error  $(\theta_i - \hat{\theta}_i)^2$ , with  $\hat{\theta}$  resulting from 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.

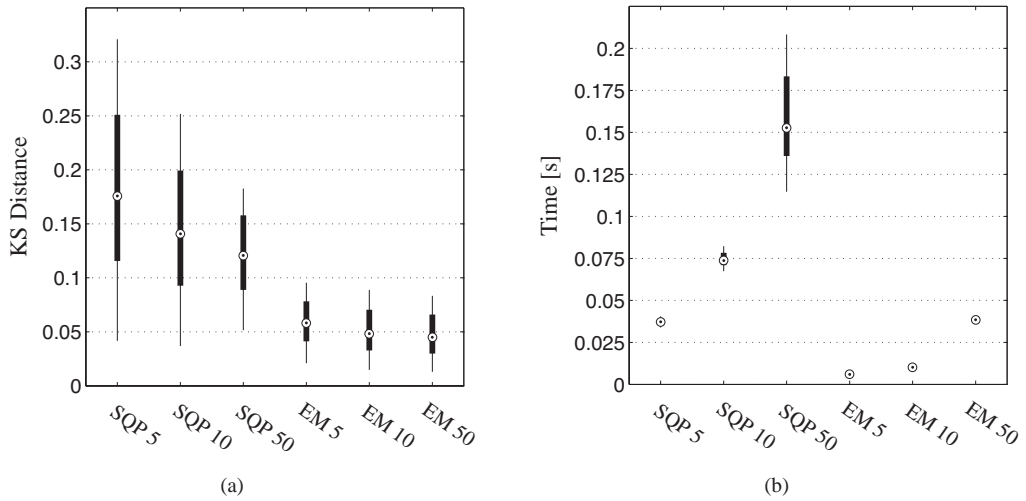


Figure 5: Comparison of a standard numerical optimization algorithm and our proposed batch EM estimation algorithm, evaluated over 100 simulations. The standard numerical optimization is performed using SQP based on the non-approximated model of Equation (19). The maximum number of iterations are marked by a number following the algorithm acronym (e.g., 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.

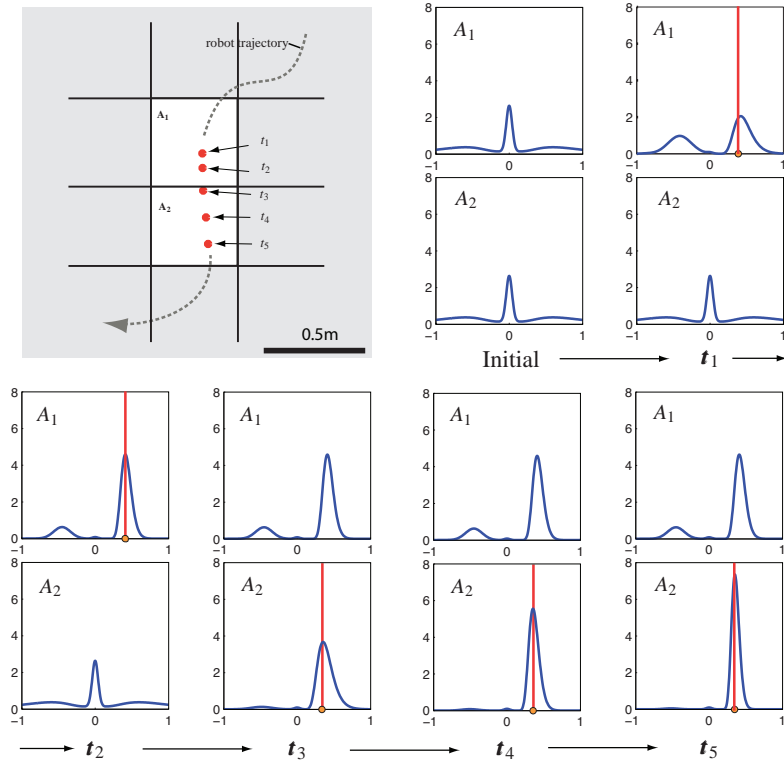


Figure 6: Illustration of the online estimation algorithm. An excerpt of the robot path is shown, passing through areas  $A_1$  and  $A_2$ , which belong to an evenly tessellated map. Five measurements (vertical red lines) from base stations  $\langle \mathcal{B}_u, \mathcal{B}_v \rangle$  are taken 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).

that area, and applies the new updates to this model. Figure 6 illustrates a simple example of how two models are updated. Let us describe this process formally. The mapping function in Equation (4) 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 2. Line 6 refers to the online estimation algorithm, which we elaborated in detail in the previous paragraphs (and showed in Equation (35)).

---

**Algorithm 2** 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  $\mathcal{M}_t$ 

```

---

## 5.2 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 9(a), and configuration **B**, as depicted in Figure 9(b) (the experimental setup is detailed in Section 7.1).

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 2.2). 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

$$\mathcal{T}_{uv,a} = \{\Delta\tau_{uv,n,t} \mid \forall \mathcal{R}_n, \text{ with } \mathbf{x}_{n,t} \in A_a\}. \quad (36)$$

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 (34)).

Figure 7 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 numbered circles 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

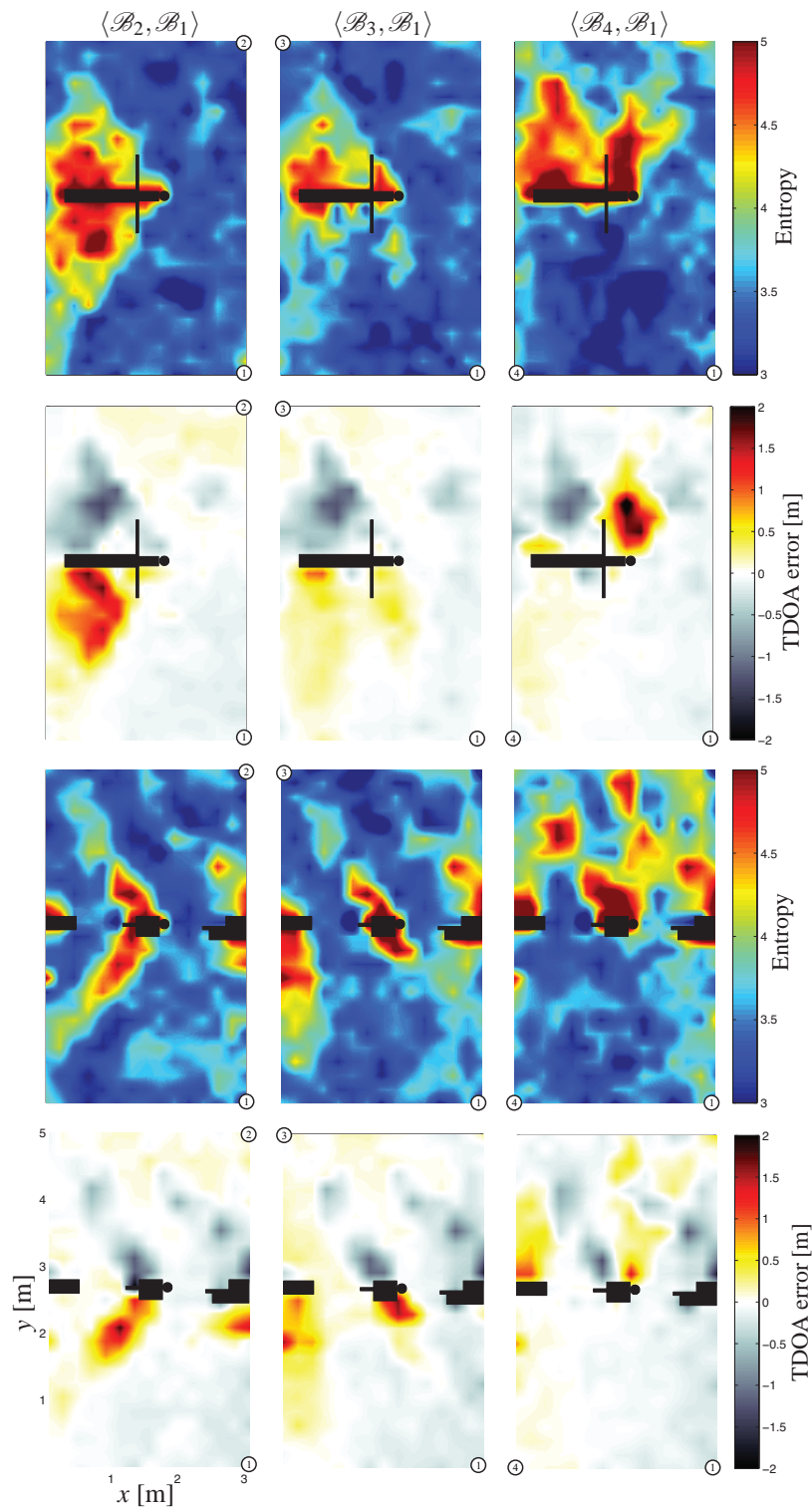


Figure 7: Evaluation of experimental data as a function of space. The numbered circles schematically represent the positions of the (numbered) UWB base stations.

calculated average TDOA error values (rows 2 and 4),

$$\Delta\bar{\tau}_{uv,a} = \frac{1}{|\mathcal{T}_{uv,a}|} \cdot \sum_{\Delta\tau_{uv,n,t} \in \mathcal{T}_{uv,a}} \Delta\tau_{uv,n,t}, \quad (37)$$

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

$$H_{uv,a} = \sum_{\Delta\tau_{uv,n,t} \in \mathcal{T}_{uv,a}} p(\Delta\tau_{uv,n,t}; \hat{\boldsymbol{\theta}}_{uv,a}) \log \frac{1}{p(\Delta\tau_{uv,n,t}; \hat{\boldsymbol{\theta}}_{uv,a})}. \quad (38)$$

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 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 Collaborative Localization

In what follows, we present a robot detection model based on noisy relative range and bearing measurements, which builds upon the notation introduced in Section 2.3. In addition to using the detection model for updating the belief representation  $\mathbf{Bel}(\mathbf{x}_{n,t})$ , our approach utilizes a *reciprocal* sampling method. Finally, in Section 6.3, we present the complete localization algorithm.

### 6.1 Range & Bearing Model

The detection model  $q(\Delta r_{mn,t}, \Delta \phi_{mn,t}; \xi)$  of Equation (11) 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 (6)–(10)). 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  $\mathbf{x}_{m,t}^{[j]}$ :

$$q^{[j]}(\Delta r_{mn,t}^{[j]}, \Delta \phi_{mn,t}^{[j]}; \xi) = \Phi \left( \mathbf{T}_e^p(\mathbf{x}_{m,t}^{[j]}, \mathbf{x}_{n,t}); \begin{bmatrix} \hat{r}_{mn} \\ \hat{\phi}_{mn} \end{bmatrix}, \xi \right) \quad (39)$$

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

$$\xi = \begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\phi^2 \end{bmatrix} \quad (40)$$

where we assume the independence of range and bearing measurements. This assumption is supported by empirical work from Goyal and Martinoli (2011), which provides experimental evidence for our platform showing that a range measurement and a bearing measurement behave like two independent Gaussian variables.

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. 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

$$\mathcal{Q}(\mathbf{x}_{n,t}^{[i]}) = \eta \cdot \prod_{d_{mn} \in D_{n,t}} q(\Delta r_{mn,t}, \Delta \phi_{mn,t}; \xi) \quad (41)$$

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 3.

---

**Algorithm 3** `Detection_Model( $D_{n,t}, \mathbf{x}_t^{[i]}, \overline{w}_t^{[i]}$ )`

---

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

We note that for  $M$  particles, Algorithm 3 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. Previous work (Prorok et al., 2012a) elaborates a clustering strategy to mitigate these effects, ultimately reducing the overall complexity to  $O(|\mathcal{N}_n|MK)$  where  $K$  is a design parameter (number of clusters).

## 6.2 Reciprocal Sampling

Let us refer to the iterative process described in Algorithm 5: instead of sampling a new particle pose  $\mathbf{x}_{n,t}^{[i]}$  from  $\mathbf{Bel}(\mathbf{x}_{n,t-1}^{[i]})$  in line 11, the reciprocal MCL routine in line 13 samples from the detection model, according to Equation (41). Thus, samples are drawn at poses which are probable given reciprocal robot observations, and which are independent of the previous belief  $\mathbf{Bel}(\mathbf{x}_{n,t-1}^{[i]})$ . 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 continuously creates particles in areas of the pose space which are likely to be significant, and thus it allows for very small particle set sizes (also shown by Prorok and Martinoli (2011)). 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 4



shows the 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 (multimodal Gaussians), sampling from the detection model is cheap. For more complex probability density functions, sophisticated and efficient methods such as slice sampling (Neal, 2003) can be employed.

---

**Algorithm 4** `Reciprocal_Sampling( $D_{n,t}, \bar{\mathcal{X}}_{n,t}$ )`

---

```

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

```

---

### 6.3 Sensor Fusion

Algorithm 5 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  $\mathcal{R}_n$ . In other words, the detected robot will opportunistically apply the detection model using data received from the robots that made the detection, if any. 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  $\alpha \leq 1$ . In our work, we implement the sampling routine in line 11 with the low-variance sampling algorithm (described by Thrun et al. (2005)). 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 sampled particle set remains unchanged.

## 7 Experiments

The following sections detail our experimental setup and the configurations of our algorithm. Finally, we report our experimental results.

### 7.1 Hardware Setup

Our experimental setup, shown in Figure 9, consists of three main elements (i) a group of ten mobile robots, (ii) an UWB positioning system composed of four base stations, and (iii) two overhead cameras with overlapping views of a  $5 \times 3$  m<sup>2</sup> arena. To perform experiments, we use ten Khepera III robots that drive (randomly) in the arena at a speed of one robot size per second. The robots run a Braitenberg controller (Braitenberg, 1984) on-board for basic obstacle avoidance based on information obtained from the robots' infrared proximity sensors (which are located roughly 2 cm above the ground). The Khepera III is a differential drive robot of 12cm diameter produced by K-Team corporation<sup>2</sup>, see Figure 8. We use the robot with a KoreBot II extension board providing a standard embedded Linux operating system on

---

<sup>2</sup><http://www.k-team.com/>

---

**Algorithm 5** 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  $X_{n,t}$ 
```

---



Figure 8: The Khepera III robot is equipped with a range and bearing extension module which utilizes sixteen infrared LEDs. On top of this board, we mounted an active marker (LED) tracking module, which simultaneously carries the UWB emitter tag.



Figure 9: (a) The  $5 \times 3 \text{ m}^2$  experimental arena contains an obstacle (configuration **A**) composed of various elements made of brick, plaster, metal, wood, and a 2 meter high tube covered in aluminum. Four UWB base stations are mounted on the ceiling in the corners of the lab room. Two overhead cameras provide ground truth positioning in the experimental area. (b) Experimental configuration **B**.

an Intel XSCALE PXA-270 processor running at 624 MHz. Communication is enabled through an IEEE 802.11b wireless card which is installed in a built-in CompactFlash slot. The robot uses wheel encoders to provide odometry readings at 5 Hz. It also uses a relative range and bearing module (Pugh et al., 2009), which is set to provide the measurements used by the detection model at a maximum frequency of 2 Hz (but can operate robustly up to a frequency of 100 Hz in a system of two robots). The modules' noise characteristics were empirically determined in our actual setup ( $\sigma_r = 0.15 \cdot r_{mn}$ , and  $\sigma_\phi = 0.15 \text{ rad}$ ). 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). Lastly, the robot is also equipped with an LED-based active marker module for tracking that also carries the UWB emitter tag, which emits positioning pulses at a frequency of 10 Hz. In practice, UWB positioning updates are received at a frequency of 2.6 Hz. The UWB localization system employed in this work is commercially available from Ubisense<sup>3</sup>, Series 7000 (sensors and compact tags). It is installed on the ceiling, in the corners of our  $40 \text{ m}^2$  laboratory. The overhead camera system runs on a central processor which also enacts the synchronization of available ground truth positioning data with all incoming raw sensor data (from the UWB system as well as from the robots). In order to compute the ground truth robot positions, the camera system utilizes the open source tracking software *SwisTrack*<sup>4</sup> (Lochmatter et al., 2008). The average error of the resulting ground truth is roughly 1cm, as reported by Prorok et al. (2010).

In order to create a NLOS setting 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 obstacles, and implement it in two variations. In experimental configuration **A**, we install a cross-shaped obstacle in our arena (see Figure 9(a)). 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

<sup>3</sup><http://www.ubisense.net>

<sup>4</sup><http://en.wikibooks.org/wiki/SwisTrack>

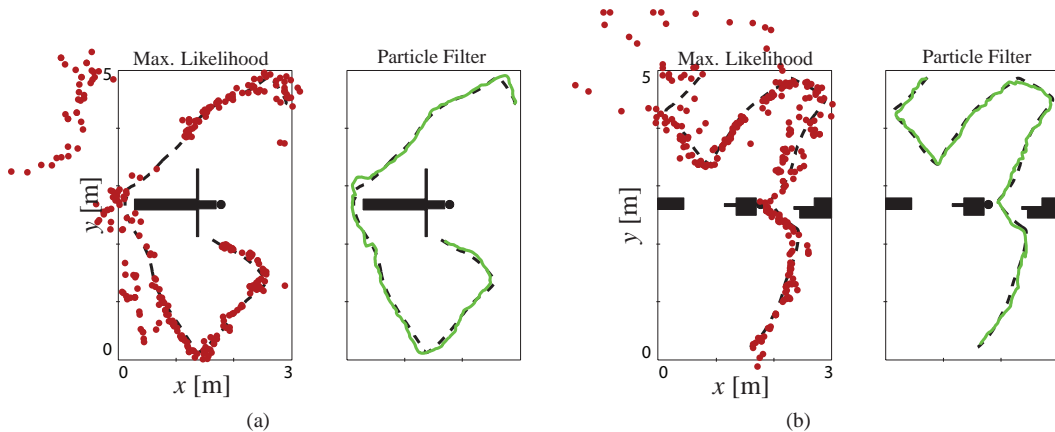


Figure 10: The panels show a 160 s excerpt of the trajectory of one robot. Maximum likelihood position estimates (calculated with the raw TDOA values) are plotted in red, and the center of mass of the robot trajectory generated by our particle filter (using  $\mathcal{M}_{M,375} + C_0$ ) is plotted in green. The ground truth trajectory is shown by a dashed black line. (a) Experimental configuration **A**. (b) Experimental configuration **B**.

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. In configuration **B**, we rearrange the same modules to form three separate obstacles, as depicted in Figure 9(b), with one obstacle in the center, and two obstacles aligned with the arena walls, creating two “doorways” for the robots to navigate through.

## 7.2 Experimental Configuration

As a way of validating our multimodal error model, we calculate two additional sets of maps: (i) one composed of Gaussian distributions instead of our proposed error model (Eq. (23)), 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 multimodal map with 375 cells).

Also, we will test the effect of collaboration on the 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. Finally, 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.

## 7.3 Overall localization error

Our localization algorithm is evaluated repeatedly 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 5 with 50 particles, and a reciprocal proportion  $\alpha = 0.02$  (unless otherwise stated). Apart from the modalities described in this paper (including odometry), the robots use no other

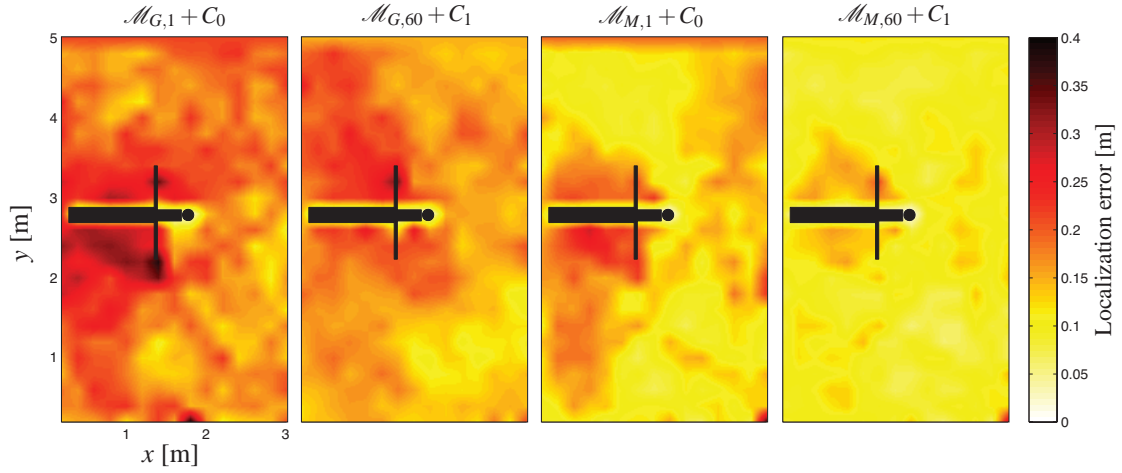


Figure 11: The graphs show the average localization error as a function of space, over a 40 minute experiment with ten robots. Four different maps ( $\mathcal{M}_{G,1}$ ,  $\mathcal{M}_{G,60}$ ,  $\mathcal{M}_{M,1}$ ,  $\mathcal{M}_{M,60}$ ) are tested with two different collaboration frequencies ( $C_0$ ,  $C_1$ ).

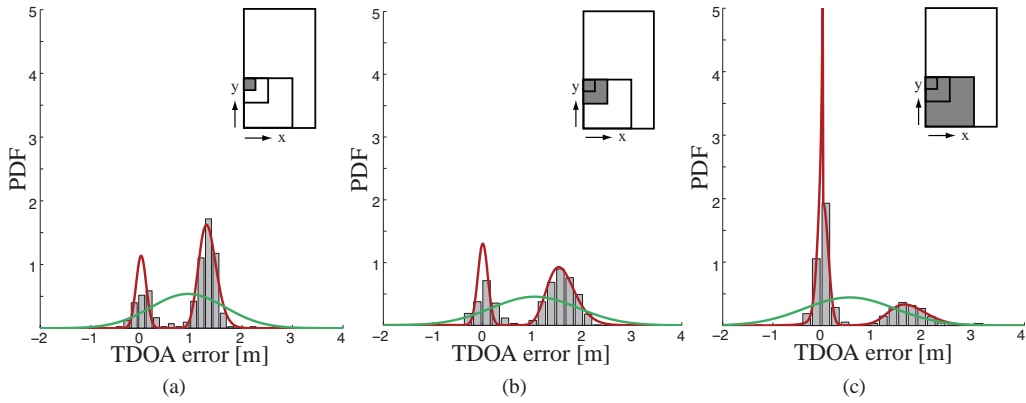


Figure 12: Normalized histograms of TDOA error data for base station pair  $\langle \mathcal{B}_1, \mathcal{B}_2 \rangle$ . We fit a Gaussian (in green) onto the data, as well as our proposed error model of Eq. (23) (in red). The data is collected over a (a)  $0.5 \times 0.5 \text{ m}^2$  large square, (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.

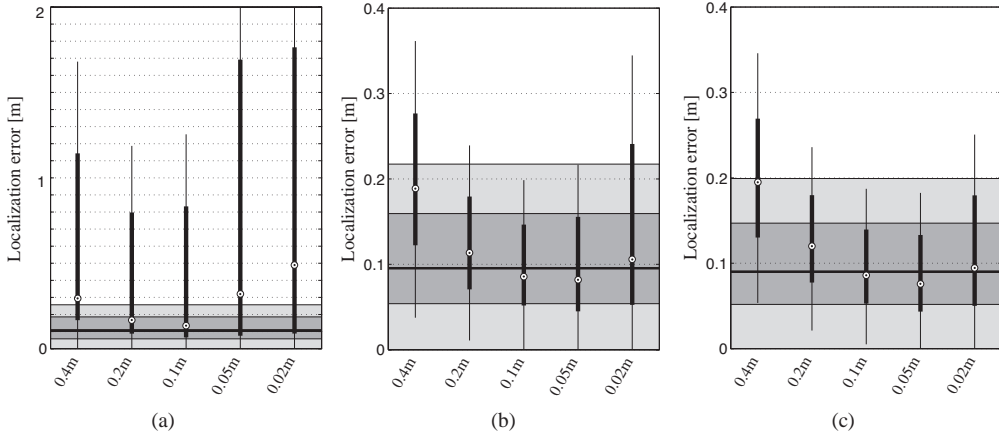


Figure 13: Localization performance using  $\mathcal{M}_{H,375}$ , with varying bin-sizes (5 boxplots along the x-axis), as well as  $\mathcal{M}_{M,375}$  (1 boxplot spanning the x-axis, shaded in grey). The boxplots show the median, 25th and 75th percentile of all localization errors for all robots 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)  $C_1$ .

sensors to localize. We discuss the localization performance in terms of the 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 2.1, Equation (2)).

Figure 10 shows two excerpts of the robot trajectory for both experimental configurations, and superimposes (i) the maximum likelihood trilateration estimates (computed with raw TDOA measurements), and (ii) the center of mass of the set of particles resulting from our localization algorithm. Finally, for comparison, we note that the error of the maximum likelihood trilateration estimates (computed with raw 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**.

Figure 11 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 ( $\mathcal{M}_{M,\cdot}$ ) is better than the Gaussian ( $\mathcal{M}_{G,\cdot}$ ) model. They also indicate that a high resolution map ( $\mathcal{M}_{\cdot,60}$ ) combined with collaboration is better than a low resolution map ( $\mathcal{M}_{\cdot,1}$ ) without collaboration. Moreover, the results indicate that the improvements help mitigate errors in areas prone to NLOS (confer Figure 7): 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.

## 7.4 Evaluation of the UWB model

To illustrate the concept of our mapping strategy, Figure 12 shows fits of (i) the probability density function proposed by our error model in Equation (23), (ii) a Gaussian probability density function, and (iii) a normalized histogram with a bin-size of 0.2 m. The three panels present data gathered in a cell of 3 different sizes, a)  $0.5 \times 0.5 \text{ m}^2$ , b)  $1 \times 1 \text{ m}^2$  and c)  $2 \times 2 \text{ m}^2$ . We note that in all cases, the multimodal approach (as proposed by our error model) suggests a good fit to the data. Also, we see that the multimodal nature of the error statistics is preserved across different scales, which is a unique characteristic of the UWB bias.

In order to assess the relevance of our multimodal model in the maps  $\mathcal{M}_{M,\cdot}$ , we compare its resulting localization performance with that of the Gaussian model employed in  $\mathcal{M}_{G,\cdot}$ , and with that of the



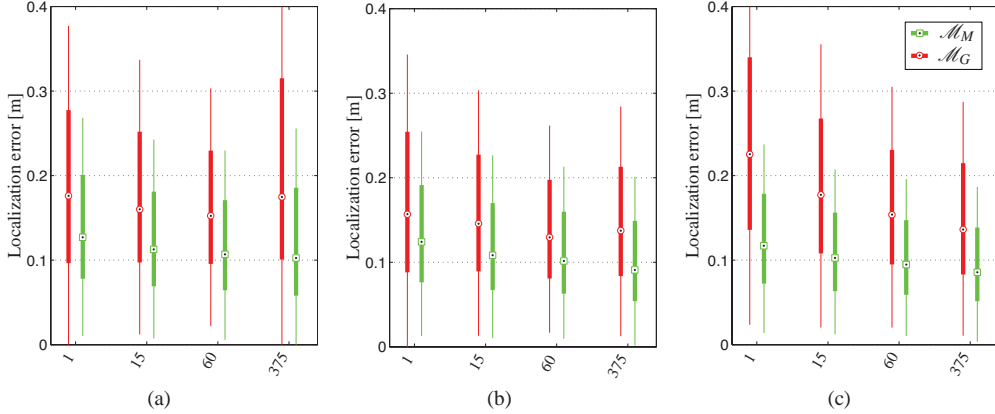


Figure 14: Localization performance using  $\mathcal{M}_M$  and  $\mathcal{M}_G$  with four grid resolutions  $N_a$  (along the x-axis). The boxplots show the median, 25th and 75th percentile of all localization errors for all robots throughout the experiment. The plots show results for configuration **A** and **B** combined. The collaboration frequency increases from left to right. (a)  $C_0$ , (b)  $C_{0.1}$ , (c)  $C_1$ .

histogram model employed in  $\mathcal{M}_{H,\cdot}$ .

Figure 13 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  $\mathcal{M}_{M,375}$ . The plots show results for configuration **A** and **B** combined, with an increasing collaboration frequency from left to right. The figure shows a significant performance difference 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 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. Also, 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 over-fit the data and the overall localization error increases.

Figure 14 shows a comparison of the localization performance obtained for  $\mathcal{M}_{M,\cdot}$  and  $\mathcal{M}_{G,\cdot}$ , for all four grid resolutions,  $N_a = \{1, 15, 60, 375\}$ . Again, the plots show results for configuration **A** and **B** combined, with an increasing collaboration frequency from left to right. We see that the multimodal models systematically produce better results. Also, for an increasing map resolution, the localization errors tend to decrease. In average and across all map resolutions, the multimodal models provide a 60% performance improvement over the Gaussian models for configuration **A**, and a 32% performance improvement for configuration **B**. Finally, we note that increasing the map granularity *systematically* improves localization performance for the multimodal models, whereas for the Gaussian models, the performance systematically improves only in Figure 14(c), when the collaboration rate is high. This points out that the Gaussian model does not reliably improve performance for increasing map granularities, showing a fundamental mismatch of the underlying data.

## 7.5 Evaluation of Collaboration

A significant difference in performance is seen in Figure 13 for the histogram model map, where the median of the non-collaborative scheme ( $C_0$ ) in plot (a) ranges in between 0.13 m to 0.48 m, compared

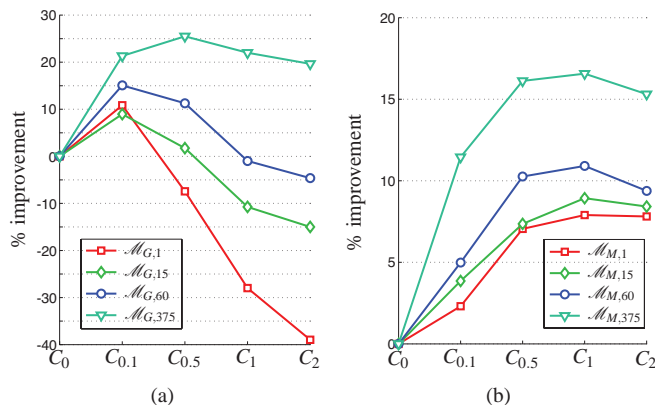


Figure 15: Improvement in percent of the localization accuracy for *collaborative* versions of the algorithm over the *non-collaborative* version ( $C_0$ ). The data points were calculated by considering the respective median error values. The collaboration frequencies vary from 0.1 Hz to 2 Hz:  $C_{0.1}$ ,  $C_{0.5}$ ,  $C_1$ ,  $C_2$ . We test (a) the Gaussian model with maps  $\mathcal{M}_{G,\cdot}$  and (b) multimodal models with maps  $\mathcal{M}_{M,\cdot}$ .

to the collaborative schemes in figures (b) and (c) ( $C_{0.1}$  and  $C_1$ ), with a range in between 0.075 m 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 14 qualitatively indicates that, when employing the multimodal error maps  $\mathcal{M}_{M,\cdot}$ , a performance increase is obtained for increasing collaboration frequencies. This result is represented quantitatively in Figure 15: 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 15(a). 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 an 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 15(b). 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 effect of increasing the uncertainty in the particle distribution is thereby exacerbated, 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.

Figure 16 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 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

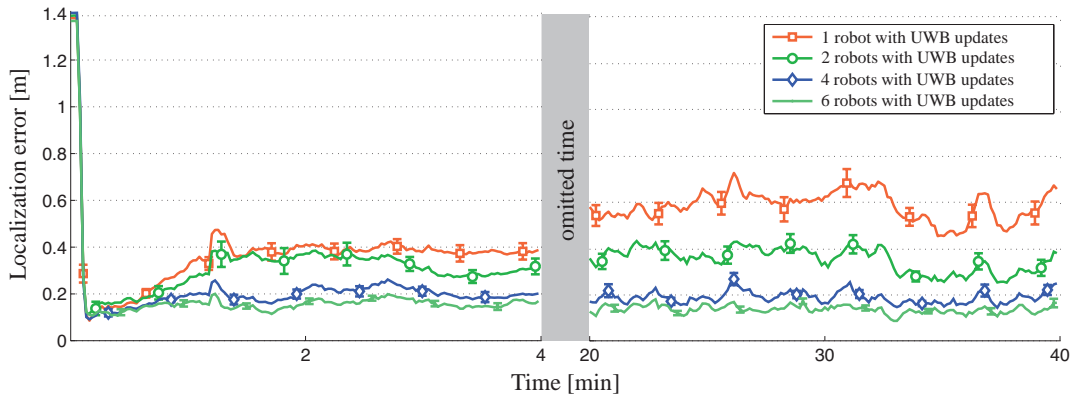


Figure 16: Average localization error (for all robots) obtained over 50 evaluations of a data set for an experiment of 40 min duration, performed on 10 Khepera III robots with 50 particles each, and a reciprocal sampling proportion of  $\alpha = 0.02$ . We vary the number of robots that receive UWB positioning updates. The error bars show a 95% confidence interval.

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.

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 sampling ( $\alpha = 0$ ). We vary the number of robots that receive UWB updates. Figure 17 reports the results. In Figure 17(a) 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 17(b), 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 the means to increase robustness.

## 8 Discussion

Our results show that, with our method, 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 (Hollinger et al., 2012; Gonzalez et al., 2009; Jourdan et al., 2005), in terms of localization accuracy, our method performs very well while bearing a small computational load. Additionally, we show quantitatively how our method compares with a baseline technique (maximum likelihood), reducing the error by a factor of 2.5.

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. (2010), 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* (Liu et al., 2007).

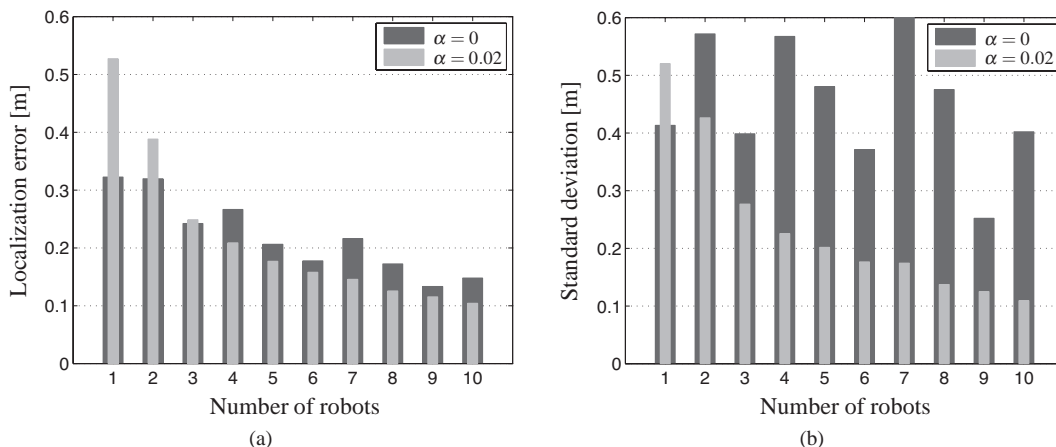


Figure 17: Localization error (for all robots, throughout the whole run) obtained over 50 evaluations of a data set for an experiment of 40 min duration performed on 10 Khepera III robots with 50 particles each. We test our algorithm with and without reciprocal sampling, i.e., once with a proportion 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.

In order to reduce the cost of building a priori error maps one should explore the simultaneous localization and mapping of UWB measurement models. An extension of our method would bear similarities to the FastSLAM algorithm (Montemerlo et al., 2002). Another trade-off of our method is that we assume our base stations to be well-localized. Possible extensions to this work could consider an augmented parameter set (including the unknown base station positions). It still remains to be explored, however, if our EM estimation framework is able to account for the additional variables, and whether it will still remain robust to local minima. Additionally, Kalman-based filtering techniques remain to be considered, at the cost of, however, altering a number of features belonging to our current methodology. Firstly, our reciprocal sampling strategy is inherently built on a particle-filter based approach, and no straight-forward equivalent for Kalman-based methods is known at this point. Secondly, a linearization of our multimodal measurement model will very likely result in poorer performance (as pointed out by Ho and Xu (2004)). Finally, our quantitative comparisons to Maximum Likelihood positioning estimates (in Section 7.3), as well as our comparisons with the Gaussian model maps (in Section 7.4) indicate that the usage of (unimodal) Gaussian measurement models leads to poorer performance. Indeed, numerous measurement campaigns (Alsindi et al., 2009; Prorok et al., 2011) have shown that the more physical space is covered, the higher the variability in NLOS biases, and the harder it is for a Gaussian model to capture the error statistics (the log-normal distribution is a heavy-tail distribution and, thus, can cope with such cases).

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 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 from NLOS areas, as well as separate differing NLOS cells from each other. Such advanced tessellation methods promise to significantly reduce the mapping effort (by allowing coarser granularities) while maintaining the same level of localization accuracy. Further, the validity of our static model still needs to be investigated for highly dynamic environments. It is to be assumed that moderately busy environments would only produce very sporadic anomalies, and that the overall localization quality would be maintained. In highly dynamic environments, however, it would be of great interest to investigate the role of collaboration, to determine whether and to what extent it is able to compensate for the static maps.

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 in our previous work (Prorok et al., 2012c) that for initial conditions sampled in a pre-defined, physically feasible range, 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 with which to compare 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.

Lastly, this work has 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, however, optimal collaboration rates are to be defined remains to be explored. Equally, the effect of heterogeneity in robot teams also remains to be explored in greater depth (beyond the results reported in Figure 16). By varying the proportion of robots with UWB sensors versus those without, an additional degree of system configurability is provided, and should be examined in conjunction with variable collaboration rates, and variable robot connectivity.

## 9 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. Our experiments showed that the error behavior of UWB measurements based on time-of-flight is dependent on the configuration of the environment, and thus, can be modeled as a function of space. This work constitutes the first steps towards studying the effects of collaboration on UWB localization. In conclusion, this work has allowed us to make two main insights: **(I)** Accurate localization with UWB in cluttered environments is possible, and can be achieved methodically. The degree of accuracy is configurable at a user level. The resulting framework is efficient, and can be deployed on very resource constrained embedded platforms. **(II)** Collaboration in the form of relative positioning improves the overall localization, provided the UWB measurement model (which is used in conjunction) faithfully captures the underlying UWB data. The integration of cheap relative positioning hardware is relevant and feasible, and represents a cost-efficient way of improving absolute positioning systems.

## Acknowledgements

The work presented in this paper was supported by the National Competence Center in Research on Mobile Information and Communication Systems (NCCR-MICS), a center supported by the Swiss National Science Foundation under grant number 51NF40-111400. The authors would like to thank the anonymous referees for their constructive feedback in improving this manuscript.

## References

- N. Alsindi, B. Alavi, and K. Pahlavan. Measurement and Modeling of Ultrawideband TOA-Based Ranging in Indoor Multipath Environments. *IEEE Transactions on Vehicular Technology*, 58:1046–1058, 2009.
- A. Bahr, J. J. Leonard, and M. F. Fallon. Cooperative Localization for Autonomous Underwater Vehicles. *Int. Journal of Robotics Research*, 28(6):714–728, 2009a.
- A. Bahr, M. R. Walter, and J. J. Leonard. Consistent Cooperative Localization. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3415–3422, 2009b.
- V. Braitenberg. *Vehicles: Experiments in synthetic psychology*. MIT Press, 1984.
- O. Cappé and E. Moulines. On-line Expectation-Maximization Algorithm for Latent Data Models. *Journal of the Royal Statistical Society. Series B (Methodological)*, 71:593–613, 2009.
- D. Fox, W. Burgard, H. Kruppa, and S. Thrun. A Probabilistic Approach to Collaborative Multi-Robot Localization. *Autonomous Robots*, 8:325–344, 2000.
- 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*, 57(5): 496–507, 2009.
- S. Goyal and A. Martinoli. Bayesian Rendezvous for Distributed Robotics Systems. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2765–2771, 2011.
- K. C. Ho and W. Xu. An accurate algebraic solution for moving source location using TDOA and FDOA measurements. *Signal Processing, IEEE Transactions on*, 52(9):2453–2463, 2004.
- G. A. Hollinger, J. Djughash, and S. Singh. Target Tracking Without Line of Sight Using Range from Radio. *Autonomous Robots*, 32(1):1–14, 2012.
- D. B. Jourdan, J. J. Jr. Deyst, M. Z. Win, and N. Roy. Monte Carlo Localization in Dense Multipath Environments Using UWB Ranging. In *IEEE International Conference on Ultra-Wideband (ICU)*, pages 314–319, 2005.
- H. Liu, H. Darabi, P. Banerjee, and J. Liu. Survey of Wireless Indoor Positioning Techniques and Systems. *IEEE Transactions on Systems, Man and Cybernetics*, 37(6):1067–1080, 2007.
- T. Lochmatter, P. Roudit, C. Cianci, N. Correll, J. Jacot, and A. Martinoli. SwisTrack - A Flexible Open Source Tracking Software for Multi-Agent Systems. In *Proceedings of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4004–4010, 2008.
- 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)*, pages 2562–2565, 2008.
- M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem. *Proceedings of the National conference on Artificial Intelligence (AAAI)*, pages 593–598, 2002.
- B. Neal. Slice Sampling. *The Annals of Statistics*, 31:705–757, 2003.
- E. Nerurkar and S. Roumeliotis. Asynchronous Multi-Centralized Cooperative Localization. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4352–4359, 2010.
- A. Prorok and A. Martinoli. A Reciprocal Sampling Algorithm for Lightweight Distributed Multi-Robot Localization. In *Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3241–3247, 2011.
- 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. In *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2010. doi: 10.1109/IPIN.2010.5647880.
- A. Prorok, P. Tomé, and A. Martinoli. Accommodation of NLOS for Ultra-Wideband TDOA Localization in Single- and Multi-Robot Systems. In *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2011. doi: 10.1109/IPIN.2011.6071927.
- A. Prorok, A. Bahr, and A. Martinoli. Low-Cost Collaborative Localization for Large-Scale Multi-Robot Systems. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 4236–4241, 2012a.



- A. Prorok, A. Bahr, and A. Martinoli. Low-Cost Multi-Robot Localization. In D Milutinovic and J Rosen, editors, *Redundancy in Robot Manipulators and Multi-Robot Systems*, pages 15–34. Lecture Notes in Electrical Engineering, Springer, 2012b. ISBN 978-3-642-33970-7.
- A. Prorok, L. Gonon, and A. Martinoli. Online Model Estimation of Ultra-Wideband TDOA Measurements for Mobile Robot Localization. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 807–814, 2012c.
- 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*, 14(2):151–162, 2009.
- Y. Qi. *Wireless Geolocation in a Non-Line-of-Sight Environment*. PhD thesis, Princeton University, 2004.
- M. Quigley, D. Stavens, A. Coates, and S. Thrun. Sub-Meter Indoor Localization in Unmodified Environments with Inexpensive Sensors. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2039–2046. IEEE, 2010. ISBN 978-1-4244-6674-0.
- Z. Sahinoglu, S. Gezici, and I. Guvenc. *Ultra-wideband Positioning Systems. Theoretical Limits, Ranging Algorithms, and Protocols*. Cambridge University Press, 2008.
- M. Segura, H. Hashemi, C. Sisterna, and V. Mut. Experimental demonstration of self-localized Ultra Wideband indoor mobile robot navigation system. In *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2010. doi: 10.1109/IPIN.2010.5647457.
- S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. The MIT Press, 2005.

## A Notation

Table 2: 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 $\mathcal{R}_n$

Table 3: UWB 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 in system
$A_a$	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 $\mathcal{B}_u$
$\boldsymbol{\mu}_u, \boldsymbol{\sigma}_u$	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 associated to area $A_a$ and base station pair $\langle \mathcal{B}_u, \mathcal{B}_v \rangle$
$p$	UWB measurement model

Table 4: 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 $t$
$\sigma_r$	Standard deviation of relative range measurements
$\sigma_\phi$	Standard deviation of relative bearing measurements
$\xi$	Covariance matrix of range and bearing 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