Mercury: An Infrastructure-Free System for Network Localization and Navigation

Zhenyu Liu, *Student Member, IEEE*, Wenhan Dai, *Student Member, IEEE*, and Moe Z. Win, *Fellow, IEEE*

Abstract—Location-awareness enables a variety of emerging applications on mobile devices. For indoor applications, a desirable way of obtaining real-time locations is by combining different sources of positional information, such as the inertial measurements, ranging measurements, and map information with an infrastructure-free system that does not rely on any customized hardware. These sources of information can be incorporated into the paradigm of network localization and navigation (NLN). However, there still lacks an infrastructure-free localization system that applies the insights of NLN to effectively fuse different types of information. In this paper, we present the Mercury system, which realizes the key ideas of NLN, including the exploitation of spatiotemporal cooperation and the use of environmental knowledge. We design a real-time belief propagation algorithm to fuse inertial measurements as well as range measurements among different users with map information. We implement this algorithm in the Mercury system formed by a network of smartphones, and evaluate its localization accuracy through experimentation. Results show that Mercury provides reliable location information and that combining spatiotemporal cooperation with environmental knowledge remarkably reduces the location uncertainty of users. Moreover, the performance of Mercury is more robust to imperfect initial positional knowledge compared with that of existing systems.

Index Terms—Cooperative networks, localization, navigation, inertial tracking, belief propagation

1 Introduction

Location-awareness enables a variety of emerging applications such as pedestrian navigation, asset tracking, crowd sensing, and social networking [1], [2], [3], [4], [5], [6], [7]. For outdoor applications, the location information can be provided by the global navigation satellite systems (GNSSs). However, the performance of these systems is degraded in the indoor environment due to the complicated propagation conditions for radio frequency (RF) signals. It remains a challenging problem how to provide accurate and real-time location information for indoor applications.

Extensive research has been carried out on indoor localization [8], [9], [10], [11], [12]. Based on the hardware requirement, we classify existing indoor localization systems into two categories, namely, infrastructure-based systems and infrastructure-free systems. The former refers to systems that include devices specially designed for localization (e.g., ultra-wide band (UWB) radios and ultrasound transceivers) [13], [14], [15], [16], whereas the latter refers to systems that consist of only commercial devices (e.g., smartphones and tablets) and existing facilities (e.g., WiFi access points (APs) and cellular base stations) [17], [18], [19], [20], [21], [22], [23], [24], [25]. Between these two types of systems, the infrastructure-free ones are more amenable for wide-scale commercial use. These systems use measurements obtained from only commercial devices and thus avoid additional hardware cost. In infrastructure-free localization systems, the algorithm used for measurement fusion is critical in determining the overall system performance.

Network localization and navigation (NLN) is a recently proposed paradigm that can incorporate different types of positional information provided by sensors in commercial devices. This paradigm exploits spatiotemporal cooperation for position inference (see Fig. 1), and has been shown to improve localization performance [1], [2], [3], [26], [27]. Each user obtains its positional information by making intra-user and inter-user measurements in the temporal and spatial domain, respectively. Intra-user measurements include the acceleration and angular velocity of a user, and inter-user measurements include the ranges and angles among users. Another important source that can be exploited to improve the localization accuracy of NLN is the environmental knowledge, e.g., map information. In particular, if a map of the indoor environment is available, prior position information can be extracted from the map and the uncertainty of the user's position can be significantly reduced (see Fig. 1) [28], [29], [30]. The idea of exploiting spatiotemporal cooperation and environmental knowledge has been demonstrated by existing infrastructure-free systems to some extent [29], [30], [31], [32], [33], [34], [35], [36], [37], [38], [39], [40], [41]. In these systems, the intra-user measurements obtained from the inertial measurement unit (IMU) and the inter-user measurements obtained via acoustic signals are used to determine the position of the user. However, there still lacks a unified and principled framework to fully exploit the spatiotemporal cooperation and environmental knowledge.

In this paper, we present an infrastructure-free localization system, Mercury, that incorporates the key ingredients

• *The authors are with the Laboratory for Information and Decision Systems (LIDS), Massachusetts Institute of Technology, Cambridge, MA 02139. E-mail: {zliu14, whdai, moewin}@mit.edu.*

Manuscript received 14 July 2016; revised 9 May 2017; accepted 15 May 2017. Date of publication 18 July 2017; date of current version 2 Apr. 2018. (Corresponding author: Moe Z. Win.)
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TMC.2017.2725265

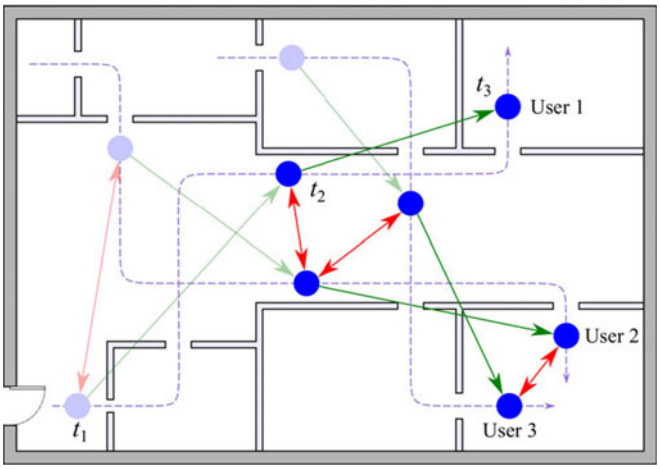


Fig. 1. Network localization and navigation: Three mobile users (blue dots) aim to localize themselves via temporal cooperation (green arrows), spatial cooperation (red arrows), and environmental knowledge (map information) in three different time instants t_1 (shaded), t_2, and t_3.

of NLN, including spatiotemporal cooperation and environmental knowledge, in a principled manner. In particular, we integrate the intra- and inter-user measurements as well as the map information under the Bayesian framework, design a belief propagation (BP) algorithm to infer the positions of users, and build a navigation system consisting of only smartphones accordingly. The main contributions of this paper are as follows.

- We propose a graphical model that represents the position, phone heading error, and the gyroscope bias of a user as a state vertex. Compared with models where the state vertex includes only the position, the proposed model enables more effective mitigation of the accumulated errors brought by the IMU.
- We develop a BP algorithm that fuses the inertial measurements obtained via the IMU in smartphones, as well as the range measurements among different users obtained via acoustic signals.
- We fuse the map knowledge with location information from spatiotemporal cooperation by imposing positional constraints in the dynamic model of the BP algorithm.
- We implement a real-time navigation system on smartphones, and demonstrate its performance improvements compared to existing systems via experimentation.

The remaining sections are organized as follows. Section 2 presents the system model. Section 3 presents the designed BP algorithm. Section 4 describes the implementation details of the infrastructure-free NLN system. Section 5 presents the experimental results. Section 6 describes the related work. Section 7 concludes the paper.

Notation. *Random variables are displayed in sans serif, upright fonts; their realizations in serif, italic fonts. Vectors and matrices are denoted by bold lowercase and uppercase letters, respectively. For example, a random variable and its realization are denoted by* x *and* x*, respectively; a random vector and its realization are denoted by* $\mathbf{x}$ *and* $\boldsymbol{x}$*, respectively.* $\boldsymbol{x}^{\mathrm{T}}$ *and* $\|\boldsymbol{x}\|$ *denote the transpose and the Euclidean norm of vector* $\boldsymbol{x}$*, respectively;* $\boldsymbol{x}_{k:n}$ *represents vector* $[\,\boldsymbol{x}_k^{\mathrm{T}}\ \boldsymbol{x}_{k+1}^{\mathrm{T}}\ \ldots\ \boldsymbol{x}_n^{\mathrm{T}}\,]^{\mathrm{T}}$*, where* $\boldsymbol{x}_i$ *is a column vector for* $i = k,\ k+1, \ldots, n$*;* $f(\boldsymbol{x})$ *denotes the probability density function (PDF)* $f_{\mathbf{x}}(\boldsymbol{x})$ *of the random vector* $\mathbf{x}$*, and* $f(\boldsymbol{x}|\boldsymbol{y})$ *denotes the conditional PDF* $f_{\mathbf{x}|\mathbf{y}}(\boldsymbol{x}|\boldsymbol{y})$ *of the random vector* $\mathbf{x}$ *conditioned on random vector* $\mathbf{y}$*;* $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ *denotes that random vector* $\mathbf{x}$ *follows the Gaussian distribution with mean* $\boldsymbol{\mu}$ *and covariance matrix* $\boldsymbol{\Sigma}$*, and* $g(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ *denotes its PDF;* $|\mathcal{S}|$ *denotes the cardinality of set* $\mathcal{S}$*.*

2 System Model

In this section, we present the graphical model for NLN, and describe details on the states and measurements in this model. This model serves as a basis for the design of Mercury.

2.1 Graphical Model

Consider a network consisting of N mobile users. Each user applies a discrete-time model so that it takes one step during each time interval. Note that at a certain time instant, different users may have taken a different number of steps. Suppose that the users in the network have taken $k_1, k_2, \ldots, k_N$ steps, respectively, at some time instant. For a particular user j, its positional information at the kth step is represented by a random vector $\mathbf{x}_k^{(j)}$ for $0 \le k \le k_j$, and this vector is called the state of the user. The trajectory of user j is then represented by the state sequence $\mathbf{x}_{0:k_j}^{(j)}$. Let $\mathbf{x}$ denote the concatenation of the state sequences of all users. The following assumptions on $\mathbf{x}$ are made throughout this paper:

1) The state sequences of different users are independent.
2) The state sequence of each user is a Markov chain.

Based on these assumptions, the joint PDF $f(\boldsymbol{x})$ of the states can be written as

$$f(\boldsymbol{x}) = \prod_{j=1}^{N} f\big(\boldsymbol{x}_0^{(j)}\big) \prod_{k=1}^{k_j} f\big(\boldsymbol{x}_k^{(j)} \big| \boldsymbol{x}_{k-1}^{(j)}\big).$$

A user makes an intra-user measurement whenever it takes a new step, and it also makes inter-user measurements intermittently when there are other users in its communication range with line-of-sight propagation conditions. For a particular user j, the intra-user measurement obtained at its kth step is represented by a random vector $\mathbf{y}_k^{(j)}$ for $0 \le k \le k_j$. In addition, suppose that user j and user i make an inter-user measurement at their k_jth and k_ith step, respectively, and such measurement is denoted by $\mathbf{y}_{k_i,k_j}^{(i,j)}$. Let $\mathbf{y}$ denote the concatenation of the intra- and inter-user measurements made by all the users, and the following assumptions on $\mathbf{y}$ are used throughout this paper:

1) The intra-user measurement of a user depends only on its states of the current step and the previous step.
2) The inter-user measurement between two users depends only on the states of their current steps.

Based on these assumptions, the conditional PDF $f(\boldsymbol{y}|\boldsymbol{x})$ can be written as

$$\begin{aligned} f(\boldsymbol{y}|\boldsymbol{x}) = &\left(\prod_{j=1}^{N} \prod_{k=1}^{k_j} f\big(\boldsymbol{y}_k^{(j)} \big| \boldsymbol{x}_k^{(j)}, \boldsymbol{x}_{k-1}^{(j)}\big) \right) \\ &\times \prod_{i=1}^{N} \prod_{l=i+1}^{N} \prod_{u=1}^{k_i} \prod_{v=1}^{k_l} f\big(\boldsymbol{y}_{u,v}^{(i,l)} \big| \boldsymbol{x}_u^{(i)}, \boldsymbol{x}_v^{(l)}\big). \end{aligned} \tag{1}$$

Note that $f\big(\boldsymbol{y}_{u,v}^{(i,l)} \big| \boldsymbol{x}_u^{(i)}, \boldsymbol{x}_v^{(l)}\big)$ in (1) is replaced with 1 if user i and user l do not make an inter-user measurement at their uth and vth step, respectively. With the above assumptions

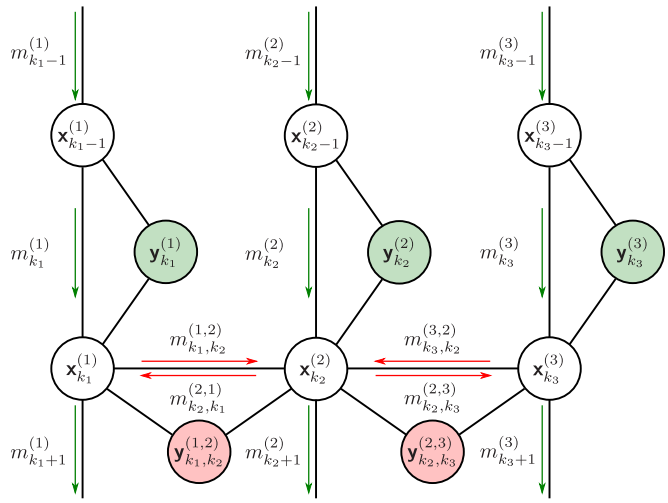


Fig. 2. Undirected graph describing the model of the positional states (white circles), intra-user measurements (green circles), and inter-user measurements (red circles) in a network consisting of three users. Inter-user measurements are available only when users perform spatial cooperation. Temporal cooperation messages (green arrows) and spatial cooperation messages (red arrows) are passed between connected states.

on the states and measurements, we can describe the system model using an undirected graph (see Fig. 2). The vertices in the graph represent states, intra-user measurements, or inter-user measurements.

2.2 Models for States and Measurements

We next present more details on the model of states and measurements. We omit the the index of the user in superscripts and subscripts for notational simplicity unless otherwise noted.

The state $\mathbf{x}_k := \left[\mathbf{p}_k^{\mathrm{T}}\ \mathbf{v}_k^{\mathrm{T}}\right]^{\mathrm{T}}$ in the graphical model consists of the user position $\mathbf{p}_k \in \mathbb{R}^2$ and a random vector $\mathbf{v}_k = \left[\mathsf{w}_k\ \mathsf{e}_k\right]^{\mathrm{T}}$ related to the errors introduced by the IMU. In particular, $\mathsf{w}_k \in \mathbb{R}$ denotes the phone heading error. It represents the estimation error for the horizontal orientation, referred to as heading, of the phone. Moreover, $\mathsf{e}_k \in \mathbb{R}$ represents the gyroscope bias in the horizontal plane. The motivation of incorporating w_k and e_k into the state is for mitigating the effects of the errors brought by the IMU. Specifically, the intra-user measurements are obtained via processing the inertial measurements provided by the IMU, including acceleration and angular velocity samples (the processing procedures are described in Section 4.2). These measurements are affected by the errors caused by the bias and noise in the IMU. If such errors are not appropriately handled, the estimated phone heading and the user's trajectory will deviate from the truth [30], [36]. To improve the localization performance, we incorporate the phone heading error and the gyroscope bias into the state vector, and propose a BP method to compute its distribution. With such a distribution, the impacts of the errors on the intra-user measurements can be mitigated.

It is assumed that sequences $\mathbf{p}_{0:k}$ and $\mathbf{v}_{0:k}$ are independent Markov chains,[1] and thus the dynamic model $f(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})$ can be expressed as follows:

$$f(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}) = f(\boldsymbol{p}_k|\boldsymbol{p}_{k-1}) f(\boldsymbol{v}_k|\boldsymbol{v}_{k-1}).$$

1. We adopt the Markov model for $\mathbf{p}_{0:k}$ because it does not require assumptions on the user's walking patterns.

The conditional PDF $f(\boldsymbol{p}_k|\boldsymbol{p}_{k-1})$ is obtained via the map information. Specifically, we partition the area of interest into a set of small disjoint squares and denote such set by $\mathcal{M} = \{1, 2, \ldots, M\}$, where M is the number of the squares. Let $\boldsymbol{c}^{(i)} \in \mathbb{R}^2$ denote the coordinate of the center of square i. For two different squares i and j, if the Euclidean distance $\|\boldsymbol{c}^{(i)} - \boldsymbol{c}^{(j)}\|$ between their centers is smaller than a predefined threshold, and there are no obstacles (e.g., walls) on the line segment connecting $\boldsymbol{c}^{(i)}$ with $\boldsymbol{c}^{(j)}$, then we say square i is a neighbor of square j and vice versa. The presence of obstacles can be determined based on the floor plan. For a position $\boldsymbol{p} \in \mathbb{R}^2$, let function $I(\boldsymbol{p})$ denote the index of the square that contains it, and let $\mathcal{N}(\boldsymbol{p})$ denote the index set containing all the neighbors of square $I(\boldsymbol{p})$. We also define $\boldsymbol{c}(\boldsymbol{p})$ as the center of the square that contains $\boldsymbol{p}$, i.e.,

$$\boldsymbol{c}(\boldsymbol{p}) := \boldsymbol{c}^{(I(\boldsymbol{p}))}. \tag{2}$$

It is assumed that at each step, a user moves to one of the neighboring squares with equal probability. The conditional PDF $f(\boldsymbol{p}_k|\boldsymbol{p}_{k-1})$ is thus given by

$$f(\boldsymbol{p}_k|\boldsymbol{p}_{k-1}) = \begin{cases} \frac{1}{|\mathcal{N}(\boldsymbol{p}_{k-1})|S}, & \text{if } I(\boldsymbol{p}_k) \in \mathcal{N}(\boldsymbol{p}_{k-1}) \\ 0, & \text{otherwise,} \end{cases} \tag{3}$$

where constant S is the area of each square.

The conditional PDF $f(\boldsymbol{v}_k|\boldsymbol{v}_{k-1})$ is obtained as follows. First, w_k is modeled as

$$\mathsf{w}_k = \mathsf{w}_{k-1} + \mathsf{e}_{k-1}\Delta t_k + \mathsf{n}_k^{\mathrm{w}},$$

where Δt_k is the duration of the kth step, and $\mathsf{n}_k^{\mathrm{w}} \sim \mathcal{N}(0, \sigma_{\mathrm{w}}^2)$ is noise with known variance σ_{w}^2. The second term is due to the fact that at the kth step, the gyroscope bias brings additional error of $\mathsf{e}_{k-1}\Delta t_k$ in the phone heading estimate. Second, e_k is modeled as [42], [43]

$$\mathsf{e}_k = \mathsf{e}_{k-1} + \mathsf{n}_k^{\mathrm{e}},$$

where $\mathsf{n}_k^{\mathrm{e}} \sim \mathcal{N}(0, \sigma_{\mathrm{e}}^2)$ is noise with known variance σ_{e}^2. The conditional PDF $f(\boldsymbol{v}_k|\boldsymbol{v}_{k-1})$ can thus be written as

$$f(\boldsymbol{v}_k|\boldsymbol{v}_{k-1}) = g(w_k; w_{k-1} + e_{k-1}\Delta t_k, \sigma_{\mathrm{w}}^2) g(e_k; e_{k-1}, \sigma_{\mathrm{e}}^2).$$

The prior distribution of $\mathbf{x}_0$ is modeled as follows. Random variables $\mathbf{p}_0$, w_0, and e_0 are independent. It is assumed that $\mathsf{e}_0 \sim \mathcal{N}(0, \varsigma_{\mathrm{e}}^2)$ with variance ς_{e}^2, whereas the distributions of $\mathbf{p}_0$ and w_0 depend on the availability of the initial positional knowledge. Specifically, $f(\boldsymbol{p}_0)$ is assumed to be constant in each square, and its value depends on whether the initial position of the user is known. If the initial position is known to be in square $i \in \mathcal{M}$, then $f(\boldsymbol{p}_0)$ is

$$f(\boldsymbol{p}_0) = \begin{cases} 1/S & \text{if } I(\boldsymbol{p}_0) = i; \\ 0 & \text{otherwise.} \end{cases}$$

If the initial position is unknown, then $f(\boldsymbol{p}_0)$ is uniform in the entire area of interest, and it can be written as

$$f(\boldsymbol{p}_0) = \frac{1}{SM}.$$

The distribution of w_0 depends on whether the initial phone heading is known. If such heading is known, there is no error in the phone heading estimate initially, and thus

$\mathsf{w}_0 = 0$ with probability 1; otherwise we model w_0 as $\mathsf{w}_0 \sim \mathcal{N}(0, \varsigma_{\mathrm{w}}^2)$, where ς_{w}^2 is a constant.[2]

The intra-user measurement $\mathbf{y}_k$ consists of the step length measurement l_k and the step direction measurement u_k, i.e., $\mathbf{y}_k := [\,\mathsf{l}_k\ \mathsf{u}_k\,]^{\mathrm{T}}$. These measurements are obtained via a processing of the acceleration and angular velocity samples. It is assumed that l_k and u_k are independent given $\mathbf{x}_{k-1:k}$, and therefore the conditional PDF $f(\boldsymbol{y}_k|\boldsymbol{x}_{k-1:k})$ can be expressed as

$$f(\boldsymbol{y}_k|\boldsymbol{x}_{k-1:k}) = f(l_k|\boldsymbol{x}_{k-1:k}) f(u_k|\boldsymbol{x}_{k-1:k}).$$

The measurement l_k is modeled as

$$\mathsf{l}_k = \|\mathbf{p}_k - \mathbf{p}_{k-1}\| + \mathsf{n}_k^{\mathrm{l}},$$

where $\mathsf{n}_k^{\mathrm{l}} \sim \mathcal{N}(0, \sigma_{\mathrm{l}}^2)$ is noise with known variance σ_{l}^2, and the conditional PDF $f(l_k|\boldsymbol{x}_{k-1:k})$ can then be written as

$$\begin{aligned} f(l_k|\boldsymbol{x}_{k-1:k}) &= f(l_k|\boldsymbol{p}_{k-1:k}) \\ &= g(l_k; \|\boldsymbol{p}_k - \boldsymbol{p}_{k-1}\|, \sigma_{\mathrm{l}}^2). \end{aligned} \tag{4}$$

The measurement u_k is modeled as

$$\mathsf{u}_k = \angle[\,\mathbf{p}_k - \mathbf{p}_{k-1}\,] + \mathsf{w}_k + \mathsf{n}_k^{\mathrm{u}},$$

where $\mathsf{n}_k^{\mathrm{u}} \sim \mathcal{N}(0, \sigma_{\mathrm{u}}^2)$ is noise with known variance σ_{u}^2, and $\angle[\boldsymbol{p}]$ is the angle of vector $\boldsymbol{p} \in \mathbb{R}^2$. The second term is due to the fact that u_k is obtained based on the phone heading estimate at the kth step, and therefore it contains the phone heading error w_k. The conditional PDF $f(u_k|\boldsymbol{x}_{k-1:k})$ can thus be written as

$$\begin{aligned} f(u_k|\boldsymbol{x}_{k-1:k}) &= f(u_k|\boldsymbol{p}_{k-1:k}, w_k) \\ &= g(u_k; \angle[\,\boldsymbol{p}_k - \boldsymbol{p}_{k-1}\,] + w_k, \sigma_{\mathrm{u}}^2). \end{aligned} \tag{5}$$

The inter-user measurement $\mathsf{y}_{k_i,k_j}^{(i,j)}$ between user i and j is the range measurement obtained via acoustic signals. It is modeled as

$$\mathsf{y}_{k_i,k_j}^{(i,j)} = \big\|\mathbf{p}_{k_i}^{(i)} - \mathbf{p}_{k_j}^{(j)}\big\| + \mathsf{n}_{k_i,k_j}^{(i,j)},$$

where $\mathsf{n}_{k_i,k_j}^{(i,j)} \sim \mathcal{N}(0, \sigma_{\mathrm{r}}^2)$ is noise with known variance of σ_{r}^2. The conditional PDF $f\big(y_{k_i,k_j}^{(i,j)}\big|\boldsymbol{x}_{k_i}^{(i)}, \boldsymbol{x}_{k_j}^{(j)}\big)$ can thus be written as

$$\begin{aligned} f\big(y_{k_i,k_j}^{(i,j)}\big|\boldsymbol{x}_{k_i}^{(i)}, \boldsymbol{x}_{k_j}^{(j)}\big) &= f\big(y_{k_i,k_j}^{(i,j)}\big|\boldsymbol{p}_{k_i}^{(i)}, \boldsymbol{p}_{k_j}^{(j)}\big) \\ &= g\big(y_{k_i,k_j}^{(i,j)}; \big\|\boldsymbol{p}_{k_i}^{(i)} - \boldsymbol{p}_{k_j}^{(j)}\big\|, \sigma_{\mathrm{r}}^2\big). \end{aligned} \tag{6}$$

Other types of sensor measurements, such as the compass measurements and WiFi received signal strength (RSS) measurements, can also be incorporated into the system by extending the measurement model. For example, the compass measurement can be incorporated as follows. Consider that $\mathbf{y}_k := [\,\mathsf{l}_k\ \mathsf{u}_k\ \mathsf{z}_k\,]^{\mathrm{T}}$. Specifically, z_k is defined as $\mathsf{z}_k := \mathsf{h}_k - \mathsf{s}_k$, where h_k is the phone heading estimate based on the acceleration and angular velocity samples, and s_k is the phone heading estimate based on the compass reading. The conditional distribution $f(\boldsymbol{y}_k|\boldsymbol{x}_{k-1:k})$ can be written as

$$f(\boldsymbol{y}_k|\boldsymbol{x}_{k-1:k}) = f(l_k|\boldsymbol{x}_{k-1:k}) f(u_k|\boldsymbol{x}_{k-1:k}) f(z_k|\boldsymbol{x}_{k-1:k}). \tag{7}$$

The conditional PDF $f(z_k|\boldsymbol{x}_{k-1:k})$ is obtained according to the following measurement model of z_k

$$\mathsf{z}_k = \mathsf{w}_k + \mathsf{n}_k^{\mathrm{z}}, \tag{8}$$

where $\mathsf{n}_k^{\mathrm{z}} \sim \mathcal{N}(0, \sigma_{\mathrm{z}}^2)$ is noise with known variance σ_{z}^2. Equation (8) is justified as follows. The variable h_k is the sum of the true phone heading with the heading error w_k, whereas s_k is modeled as the sum of the true phone heading with noise $\mathsf{n}_k^{\mathrm{z}}$. Therefore, their difference $\mathsf{z}_k = \mathsf{h}_k - \mathsf{s}_k$ can be written as (8). The conditional PDF $f(z_k|\boldsymbol{x}_{k-1:k})$ is thus

$$f(z_k|\boldsymbol{x}_{k-1:k}) = g(z_k; w_k, \sigma_{\mathrm{z}}^2). \tag{9}$$

3 Measurement Fusion with Belief Propagation Algorithm

In this section, we describe the BP algorithm for spatiotemporal cooperation [27], and present efficient methods for computing the messages and beliefs when intra- and inter-user measurements are available.

3.1 BP Algorithm for Spatiotemporal Cooperation

We use the BP algorithm to compute the posterior distribution of the positional states given the available intra- and inter-user measurements. We adopt the BP algorithm for three reasons. First, the BP algorithm is efficient in computing such distribution based on the graphical model. In particular, the BP algorithm fuses the intra- and inter-user measurements as well as the map constraints in a principled manner, and the uncertainty in the position estimate of each user is also taken into account. Second, the BP algorithm facilitates the implementation of cooperation, especially spatial cooperation. Specifically, a user exchanges information only with users in its communication range, called neighboring users, when it performs spatial cooperation. Third, the BP algorithm enables a distributed implementation of the system. With the BP algorithm, a user can process the measurements and estimate its position locally, and thus a central processing unit is not required.

The posterior distribution of a state given the available measurements in the BP algorithm is called the belief of that state, and we denote the belief of state $\mathbf{x}_{k_j}^{(j)}$ by $b_{k_j}^{(j)}(\boldsymbol{x}_{k_j}^{(j)})$. Moreover, the distribution $b_{k_j}^{(j)}(\boldsymbol{p}_{k_j}^{(j)})$ of $\mathbf{p}_{k_j}^{(j)}$ is obtained by marginalizing $\mathbf{v}_{k_j}^{(j)}$ out, i.e.,

$$b_{k_j}^{(j)}(\boldsymbol{p}_{k_j}^{(j)}) = \int b_{k_j}^{(j)}(\boldsymbol{x}_{k_j}^{(j)}) d\boldsymbol{v}_{k_j}^{(j)}.$$

Let $b_{k_j}^{(j)}(\boldsymbol{v}_{k_j}^{(j)}|\boldsymbol{p}_{k_j}^{(j)})$ denote the conditional PDF of $\mathbf{v}_{k_j}^{(j)}$ given $\mathbf{p}_{k_j}^{(j)}$ and the available measurements. The belief $b_{k_j}^{(j)}(\boldsymbol{x}_{k_j}^{(j)})$ can then be written as

$$b_{k_j}^{(j)}(\boldsymbol{x}_{k_j}^{(j)}) = b_{k_j}^{(j)}(\boldsymbol{p}_{k_j}^{(j)}) b_{k_j}^{(j)}(\boldsymbol{v}_{k_j}^{(j)}|\boldsymbol{p}_{k_j}^{(j)}).$$

In the BP algorithm, the belief is computed according to the messages passed in the graphical model (see Fig. 2) [44]. Messages are passed between connected states, and a message is a function of the destination state. There are two types of messages in the proposed graphical model, namely temporal cooperation messages and spatial cooperation messages. The former are passed from one time instant to the succeeding one

2. When the initial phone heading is unknown, Mercury chooses a value in $[-\pi, \pi)$ randomly according to the uniform distribution and sets it as the initial phone heading estimate. Therefore, $[\mathsf{w}_0]_\pi$ is uniformly distributed in $[-\pi, \pi)$, where $[\mathsf{w}_0]_\pi$ denotes the value in $[-\pi, \pi)$ that is congruent to w_0 modulo 2π. In other words, $[w_0]_\pi = w_0 + 2k\pi$, where k is an integer such that $w_0 + 2k\pi \in [-\pi, \pi)$. To facilitate the derivation of our algorithm, we make an approximation on w_0 such that $\mathsf{w}_0 \sim \mathcal{N}(0, \varsigma_{\mathrm{w}}^2)$. When $\varsigma_{\mathrm{w}}^2 > \pi^2$, the PDF of $[\mathsf{w}_0]_\pi$ is close to that of the uniform distribution in $[-\pi, \pi)$, and such approximation is tight.

when intra-user measurements are available, whereas the latter are passed between two users performing spatial cooperation when inter-user measurements are available. The belief of a state is proportional to the product of all the messages passed to it. Specifically, let $m_{k_j}^{(j)}(\boldsymbol{x}_{k_j}^{(j)})$ denote the temporal cooperation message passed from state $\mathbf{x}_{k_j-1}^{(j)}$ to $\mathbf{x}_{k_j}^{(j)}$. In addition, suppose user j cooperates with users in a subset $\mathcal{C}_{k_j}^{(j)} \subset \{1,2,\ldots,N\}$ at the k_jth step, and let $m_{k_i,k_j}^{(i,j)}(\boldsymbol{x}_{k_j}^{(j)})$ $(i \in \mathcal{C}_{k_j}^{(j)})$ denote the spatial cooperation message passed from state $\mathbf{x}_{k_i}^{(i)}$ to $\mathbf{x}_{k_j}^{(j)}$. The belief $b_{k_j}^{(j)}(\boldsymbol{x}_{k_j}^{(j)})$ can then be written as

$$b_{k_j}^{(j)}(\boldsymbol{x}_{k_j}^{(j)}) = \frac{1}{Z_{k_j}^{(j)}} m_{k_j}^{(j)}(\boldsymbol{x}_{k_j}^{(j)}) \prod_{i \in \mathcal{C}_{k_j}^{(j)}} m_{k_i,k_j}^{(i,j)}(\boldsymbol{x}_{k_j}^{(j)}),$$

where $Z_{k_j}^{(j)}$ is a normalization constant such that

$$\int b_{k_j}^{(j)}(\boldsymbol{x}_{k_j}^{(j)}) d\boldsymbol{x}_{k_j}^{(j)} = 1.$$

The messages and beliefs in the BP algorithm are computed in a recursive manner when new intra- and inter-user measurements are available. When intra-user measurement $\boldsymbol{y}_{k_j}^{(j)}$ is available, the temporal cooperation message $m_{k_j}^{(j)}(\boldsymbol{x}_{k_j}^{(j)})$ is computed as

$$\begin{aligned} m_{k_j}^{(j)}(\boldsymbol{x}_{k_j}^{(j)}) = \int & f(\boldsymbol{x}_{k_j}^{(j)}|\boldsymbol{x}_{k_j-1}^{(j)}) f(\boldsymbol{y}_{k_j}^{(j)}|\boldsymbol{x}_{k_j-1}^{(j)}, \boldsymbol{x}_{k_j}^{(j)}) \\ & \times b_{k_j-1}^{(j)}(\boldsymbol{x}_{k_j-1}^{(j)}) d\boldsymbol{x}_{k_j-1}^{(j)}, \end{aligned} \tag{10}$$

and when inter-user measurement $\boldsymbol{y}_{k_i,k_j}^{(i,j)}$ is available, the spatial cooperation message $m_{k_i,k_j}^{(i,j)}(\boldsymbol{x}_{k_j}^{(j)})$ is computed as

$$\begin{aligned} m_{k_i,k_j}^{(i,j)}(\boldsymbol{x}_{k_j}^{(j)}) = \int & f(\boldsymbol{y}_{k_i,k_j}^{(i,j)}|\boldsymbol{x}_{k_i}^{(i)}, \boldsymbol{x}_{k_j}^{(j)}) m_{k_i}^{(i)}(\boldsymbol{x}_{k_i}^{(i)}) \\ & \times \prod_{l \in \mathcal{C}_{k_i}^{(i)} \setminus j} m_{k_l,k_i}^{(l,i)}(\boldsymbol{x}_{k_i}^{(i)}) d\boldsymbol{x}_{k_i}^{(i)}. \end{aligned} \tag{11}$$

The integrals in (10) and (11), however, are analytically intractable in general. To address this issue, we propose efficient methods for computing the messages and beliefs described in the following section.

3.2 Approximate Belief Update in the BP Algorithm

We next present an efficient method for updating the beliefs when intra- and inter-user measurements are available. We make a few approximations on the measurement model. Specifically, $\boldsymbol{p}_k$ and $\boldsymbol{p}_{k-1}$ in (4) and (5) are approximated by $\boldsymbol{c}(\boldsymbol{p}_k)$ and $\boldsymbol{c}(\boldsymbol{p}_{k-1})$, respectively, where function $\boldsymbol{c}(\boldsymbol{p})$ is defined in (2). In addition, $\boldsymbol{p}_{k_i}^{(i)}$ and $\boldsymbol{p}_{k_j}^{(j)}$ in (6) are approximated by $\boldsymbol{c}(\boldsymbol{p}_{k_i}^{(i)})$ and $\boldsymbol{c}(\boldsymbol{p}_{k_j}^{(j)})$, respectively. Then we have

$$f(l_k|\boldsymbol{p}_{k-1:k}) \approx g(l_k; \|\boldsymbol{c}(\boldsymbol{p}_k) - \boldsymbol{c}(\boldsymbol{p}_{k-1})\|, \sigma_{\mathrm{l}}^2) \tag{12}$$

$$f(u_k|\boldsymbol{p}_{k-1:k}, w_k) \approx g(u_k; \angle[\boldsymbol{c}(\boldsymbol{p}_k) - \boldsymbol{c}(\boldsymbol{p}_{k-1})] + w_k, \sigma_{\mathrm{u}}^2) \tag{13}$$

$$f(y_{k_i,k_j}^{(i,j)}|\boldsymbol{p}_{k_i}^{(i)}, \boldsymbol{p}_{k_j}^{(j)}) \approx g(y_{k_i,k_j}^{(i,j)}; \|\boldsymbol{c}(\boldsymbol{p}_{k_i}^{(i)}) - \boldsymbol{c}(\boldsymbol{p}_{k_j}^{(j)})\|, \sigma_{\mathrm{r}}^2). \tag{14}$$

Note that (12), (13), and (14) are common approximations [30], [34]. Moreover, (12), (13), and (14) are asymptotically accurate as the size of each square goes to zero. Based on these approximations, we show an important property of belief $b_{k_j}^{(j)}(\boldsymbol{x}_{k_j})$ in the following proposition. This property will be used for designing an efficient belief update method when intra- and inter-user measurements are available.

Proposition 1. *With approximations (12), (13), and (14), $b_{k_j}^{(j)}(\boldsymbol{x}_{k_j}^{(j)})$ can be expressed as*

$$b_{k_j}^{(j)}(\boldsymbol{x}_{k_j}^{(j)}) = b_{k_j}^{(j)}(\boldsymbol{c}(\boldsymbol{p}_{k_j}^{(j)})) b_{k_j}^{(j)}(\boldsymbol{v}_{k_j}^{(j)}|\boldsymbol{c}(\boldsymbol{p}_{k_j}^{(j)})),$$

for $j = 1,2,\ldots,N$ and $k_j \geq 0$, where the conditional distribution $b_{k_j}^{(j)}(\boldsymbol{v}_{k_j}^{(j)}|\boldsymbol{c}(\boldsymbol{p}_{k_j}^{(j)}))$ is a Gaussian mixture distribution, with all its components having the same covariance matrix that does not depend on $\boldsymbol{x}_{1:k_j}^{(j)}$.

Proof. See the Appendix, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/10.1109/TMC.2017.2725265. □

Remark 1. Proposition 1 indicates that the approximations (12), (13), and (14) enable the efficient computation of belief $b_{k_j}^{(j)}(\boldsymbol{x}_{k_j}^{(j)})$. First, $b_{k_j}^{(j)}(\boldsymbol{x}_{k_j}^{(j)})$ can be represented by the product of $b_{k_j}^{(j)}(\boldsymbol{c}^{(l)})$ and $b_{k_j}^{(j)}(\boldsymbol{v}_{k_j}^{(j)}|\boldsymbol{c}^{(l)})$ with $l = 1,2,\ldots,M$. In particular, $b_{k_j}^{(j)}(\boldsymbol{c}^{(l)})$ is a scalar, and $b_{k_j}^{(j)}(\boldsymbol{v}_{k_j}^{(j)}|\boldsymbol{c}^{(l)})$ is a Gaussian mixture distribution parameterized by the weights, means, and covariance matrices of its components. Second, these scalars and the Gaussian mixture parameters can be computed easily, as shown in the Appendix, available in the online supplemental material.

Remark 2. The proof in the Appendix, available in the online supplemental material, considers only the step direction and step length measurements. It can be shown that Proposition 1 still holds when the compass measurements given by (7), (8) and (9) are incorporated. In particular, the proof is still valid with the equations for computing the Gaussian mixture parameters modified.

An efficient method for belief update can be designed based on Proposition 1. Consider the belief update of user j at its k_jth step, and for simplicity, assume that user j performs spatial cooperation with user i, i.e., $\mathcal{C}_{k_j}^{(j)} = \{i\}$. The belief update consists of two stages. In the first stage, user j obtains intra-user measurements $l_{k_j}^{(j)}$ and $u_{k_j}^{(j)}$. User j then evaluates the temporal cooperation message and obtains an intermediate belief $\tilde{b}_{k_j}^{(j)}(\boldsymbol{x}_{k_j}^{(j)})$ represented by the product of $\tilde{b}_{k_j}^{(j)}(\boldsymbol{c}^{(l)})$ and Gaussian mixture distribution $\tilde{b}_{k_j}^{(j)}(\boldsymbol{v}_{k_j}^{(j)}|\boldsymbol{c}^{(l)})$ with $l = 1,2,\ldots,M$ (see the Appendix, available in the online supplemental material). In the second stage, user j obtains inter-user range measurements $y_{k_i,k_j}^{(i,j)}$, sends $\tilde{b}_{k_j}^{(j)}(\boldsymbol{c}^{(l)})$ to user i, and receives $\tilde{b}_{k_i}^{(i)}(\boldsymbol{c}^{(l)})$ from user i. User j then evaluates the spatial cooperation message and obtains the updated belief $b_{k_j}^{(j)}(\boldsymbol{x}_{k_j}^{(j)})$ represented by the product of $b_{k_j}^{(j)}(\boldsymbol{c}^{(l)})$ and $b_{k_j}^{(j)}(\boldsymbol{v}_{k_j}^{(j)}|\boldsymbol{c}^{(l)}) = \tilde{b}_{k_j}^{(j)}(\boldsymbol{v}_{k_j}^{(j)}|\boldsymbol{c}^{(l)})$ with $l = 1,2,\ldots,M$ (see the Appendix, available in the online supplemental material).[3] If no user performs spatial cooperation with user j at its k_jth step, i.e., $\mathcal{C}_{k_j}^{(j)} = \emptyset$, then the belief update in the second stage is not required, and $b_{k_j}^{(j)}(\boldsymbol{c}^{(l)}) = \tilde{b}_{k_j}^{(j)}(\boldsymbol{c}^{(l)})$.

3. The proof in the Appendix, available in the online supplemental material, shows that inter-user measurements affect the posterior distribution of the user's position, but they do not affect the distribution of the phone heading error and gyroscope bias. Therefore, we have $b_{k_j}^{(j)}(\boldsymbol{v}_{k_j}^{(j)}|\boldsymbol{c}^{(l)}) = \tilde{b}_{k_j}^{(j)}(\boldsymbol{v}_{k_j}^{(j)}|\boldsymbol{c}^{(l)})$.

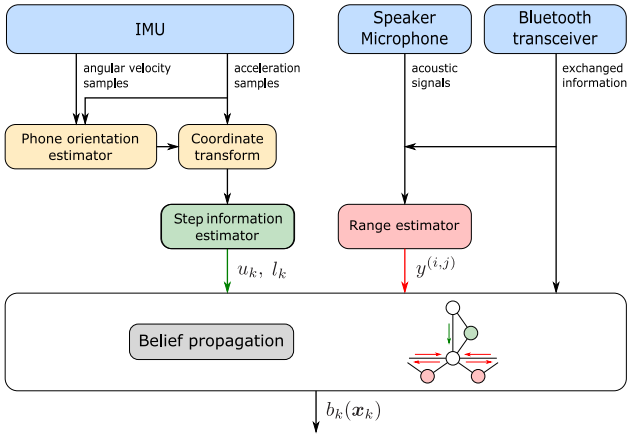


Fig. 3. System architecture for Mercury.

Moreover, we make three other approximations in the belief update process to reduce the computation and communication overhead. Consider the belief update of user j at its k_jth step. First, user i sends only the largest $N_\mathrm{c} < M$ elements in the set $\{\tilde{b}_{k_i}^{(i)}(\boldsymbol{c}^{(l)}), l = 1, 2, \ldots, M\}$ to user j when performing spatial cooperation. Second, user j keeps the largest $N_\mathrm{b} < M$ elements in the set $\{b_{k_j}^{(j)}(\boldsymbol{c}^{(l)}), l = 1, 2, \ldots, M\}$, and sets the rest to 0. Third, user j approximates the Gaussian mixture distribution $b_{k_j}^{(j)}(\boldsymbol{v}_{k_j}^{(j)}|\boldsymbol{c}^{(l)})$ by keeping N_g of its components with the largest weights, where N_g is a constant. The computational complexity of the proposed belief update method can be obtained as follows. Since the beliefs of N_b squares are tracked, and for each square the parameters of a Gaussian mixture distribution with N_g components are computed, the total computational complexity scales as $\mathcal{O}(N_\mathrm{b}N_\mathrm{g})$.[4] The efficiency of our algorithm is demonstrated by experimental results, since desirable localization performance can be achieved with a small N_g and a reasonable N_b.

Combining the results in this section, we design a distributed BP algorithm that incorporates spatiotemporal cooperation and map information for NLN. The details of this algorithm are presented in Algorithm 1 using user j as an example: lines 3-18 describe the first stage of the belief update with intra-user measurements; lines 19-24 describe the second stage of the belief update with inter-user measurements.

Algorithm 1. Distributed BP Algorithm for Mercury

Input: Prior distribution $b_0^{(j)}(\boldsymbol{x}_0^{(j)})$; step length and direction measurements $l_{k_j}^{(j)}$ and $u_{k_j}^{(j)}$; range measurements $y_{k_i,k_j}^{(i,j)}$;

Output: Belief $b_{k_j}^{(j)}(\boldsymbol{c}^{(l)})$ and $b_{k_j}^{(j)}(\boldsymbol{v}_{k_j}^{(j)}|\boldsymbol{c}^{(l)})$, for $l \in \mathcal{M}$ and $1 \le k_j \le K$.

1: $k_j \leftarrow 1$;
2: **while** $k_j \le K$ **do**
3: Compute $\boldsymbol{\Sigma}_{k_j}$ according to (20) in the Appendix, available in the online supplemental material;
4: **for all** $l \in \mathcal{M}$ **do**
5: **for all** $i \in \mathcal{N}(\boldsymbol{c}^{(l)})$ **do**
6: **for all** $m = 1, 2, \ldots, N_\mathrm{g}$ **do**
7: Compute $\boldsymbol{\mu}_{k_j}^{(i,m)}(\boldsymbol{c}^{(l)})$ and $\tilde{\alpha}_{k_j}^{(i,m)}(\boldsymbol{c}^{(l)})$ according to (21) and (22) in the Appendix, available in the online supplemental material, respectively;
8: **end for**
9: **end for**
10: **for all** $n = 1, 2, \ldots, N_\mathrm{g}|\mathcal{N}(\boldsymbol{c}^{(l)})|$ **do**
11: Obtain $\tilde{\alpha}_k^{(n)}(\boldsymbol{p}_k)$ and $\boldsymbol{\mu}_k^{(n)}(\boldsymbol{p}_k)$ by reordering $\{\tilde{\alpha}_{k_j}^{(i,m)}(\boldsymbol{c}^{(l)})\}$ and $\{\tilde{\mu}_{k_j}^{(i,m)}(\boldsymbol{c}^{(l)})\}$ as described in the Appendix, available in the online supplemental material;
12: **end for**
13: $\tilde{b}_{k_j}^{(j)}(\boldsymbol{c}^{(l)}) \leftarrow \sum_{n=1}^{N_\mathrm{g}|\mathcal{N}(\boldsymbol{c}^{(l)})|} \tilde{\alpha}_{k_j}^{(n)}(\boldsymbol{c}^{(l)})$;
14: **for all** $1 \le n \le N_\mathrm{g}$ **do**
15: $\alpha_{k_j}^{(n)}(\boldsymbol{c}^{(l)}) \leftarrow \tilde{\alpha}_{k_j}^{(n)}(\boldsymbol{c}^{(l)}) / \sum_{v=1}^{N_g} \tilde{\alpha}_{k_j}^{(v)}(\boldsymbol{c}^{(l)})$;
16: **end for**
17: $b_{k_j}^{(j)}(\boldsymbol{v}_{k_j}^{(j)}|\boldsymbol{c}^{(l)}) \leftarrow \sum_{n=1}^{N_\mathrm{g}} \alpha_{k_j}^{(n)}(\boldsymbol{c}^{(l)}) g(\boldsymbol{v}_{k_j}^{(j)}; \boldsymbol{\mu}_{k_j}^{(n)}(\boldsymbol{c}^{(l)}), \boldsymbol{\Sigma}_{k_j})$;
18: **end for**
19: **for all** $i \in \mathcal{C}_{k_j}^{(j)}$ **do**
20: Transmit $\tilde{b}_{k_j}^{(j)}(\boldsymbol{c}^{(l)})$ to user i, and receive $\tilde{b}_{k_i}^{(i)}(\boldsymbol{c}^{(l)})$ from user i;
21: **for all** $l \in \mathcal{M}$ **do**
22: $b_{k_j}^{(j)}(\boldsymbol{c}^{(l)}) \leftarrow \tilde{b}_{k_j}^{(j)}(\boldsymbol{c}^{(l)}) \sum_{n=1}^{M} f(y_{k_i,k_j}^{(i,j)}|\boldsymbol{c}^{(n)}, \boldsymbol{c}^{(l)}) \times \tilde{b}_{k_i}^{(i)}(\boldsymbol{c}^{(n)})$;
23: **end for**
24: **end for**
25: $k_j \leftarrow k_j + 1$;
26: **end while**

4 System Implementation

In this section, we present implementation details of Mercury. We first describe the system architecture, and then show how to obtain phone orientation estimates, step direction measurements, and range measurements.

4.1 System Architecture

Fig. 3 shows the system architecture of Mercury. In Mercury, a Bluetooth transceiver and built-in sensors including the IMU, speaker, and microphone are used to obtain intra- and inter-user measurements (green and red arrows in Fig. 3). These measurements are fused by the BP algorithm to obtain the positional belief of a user. The location at which the maximum belief is achieved is the position estimate of the user.

The IMU provides acceleration and angular velocity samples, and these samples are used to estimate the three-dimensional phone orientation [45]. Based on the phone orientation estimate, the acceleration samples are transformed from the phone coordinate system to the earth coordinate system. The transformed acceleration samples are then used to compute the step direction and length, which are used as the intra-user measurements in the BP algorithm. In Mercury, the difference between the phone heading and the step direction, known as heading offset, is assumed to be time-varying during the localization procedure. This assumption is more practical compared with the assumption in existing systems that the heading offset is a constant [34], [35], [36], since the user may rotate the phone and the heading offset may change.

The speaker and microphone are used to transmit and record acoustic signals, respectively, when a user performs

4. With spatial cooperation, there is additional computational complexity for incorporating the inter-user measurements that scales as $\mathcal{O}(N_\mathrm{b}N_\mathrm{c})$. However, since spatial cooperation is performed only intermittently in Mercury, its contribution to the overall computational complexity is negligible.

spatial cooperation. Based on the recorded acoustic signal, the user computes its range with respect to another user [41], and the result is used as the inter-user measurement in the BP algorithm. Moreover, the Bluetooth transceiver is used to exchange the information that is required for the range computation and belief update.

A map of the accessible area is pre-processed and serves as an input for Mercury. Specifically, the borders of the accessible area are labeled, and then the area inside the borders is partitioned into identical squares. The size of each square affects both the localization accuracy and the computational complexity. On the one hand, the approximations (12), (13), and (14) become more accurate, and the localization accuracy is higher with a smaller square size. On the other hand, a larger number of squares are required to cover the area of interest, and thus the computational complexity is increased. As a tradeoff, we set the size of each square as $0.7\ \text{m} \times 0.7\ \text{m}$ for the results presented in this paper.

4.2 Phone Orientation Estimation and Step Direction Measurement

The method proposed in [45] is used in Mercury for estimating the phone orientation. In this method, a calibration term obtained according to the acceleration samples is considered in the phone orientation estimation. However, such calibration reduces the error only in the estimate of the angle between the phone and the direction of the gravity. It does not reduce the error in the horizontal orientation estimate, i.e., the phone heading estimate. Such limitation motivates the incorporation of the phone heading error into the state vector.

The direction of a step is computed based on the spectrum of the acceleration samples obtained during this step as follows. First, the acceleration samples obtained in this step are transformed to the earth coordinate system based on the phone orientation estimate. Then a discrete Fourier transform (DFT) is performed on the transformed samples to obtain their spectrum. Let $\boldsymbol{a}_{\text{N}} = [\, a_{\text{N},0}\ a_{\text{N},1}\ \ldots\ a_{\text{N},L-1}\,]^{\text{T}}$ and $\boldsymbol{a}_{\text{W}} = [\, a_{\text{W},0}\ a_{\text{W},1}\ \ldots\ a_{\text{W},L-1}\,]^{\text{T}}$ denote the acceleration spectrum of the kth step in the north and west direction, respectively, where L is the number of acceleration samples obtained in that step. We make a similar observation as in [46] that the first and second frequency components, $\boldsymbol{a}_1 := [\, a_{\text{N},1}\ a_{\text{W},1}\,]^{\text{T}}$ and $\boldsymbol{a}_2 := [\, a_{\text{N},2}\ a_{\text{W},2}\,]^{\text{T}}$, capture most of the energy in the spectrum.[5] We also observe that $\boldsymbol{a}_1$ has a large vector component in the direction $u_k + \pi/2$ that is perpendicular to the step direction, whereas $\boldsymbol{a}_2$ has a large vector component in the step direction u_k. Based on these observations, we compute the step direction u_k by solving the following optimization problem analytically

$$u_k = \underset{\theta\in[0,2\pi)}{\arg\max} \left(\left|\boldsymbol{a}_1^{\text{T}}\boldsymbol{e}(\theta+\pi/2)\right|^2 + \left|\boldsymbol{a}_2^{\text{T}}\boldsymbol{e}(\theta)\right|^2\right),$$

where $\boldsymbol{e}(\theta) := [\, \cos\theta\ \sin\theta\,]^{\text{T}}$.

The step direction measurement u_k contains the phone heading error caused by the bias and noise in the IMU, since it is computed based on the phone orientation estimate. To mitigate the impacts of the phone heading error, we incorporate w_k and e_k into the state vector $\mathbf{x}_k$. In addition, while the gyroscope bias exists along all the three axes of the phone coordinate system, only the component along the Z-axis (i.e., the axis perpendicular to the screen of the smartphone) is considered in Mercury. The reason is explained as follows. For simplicity, we assume that the Z-axis is perpendicular to the ground, and therefore the biases with respect to the other two axes, X- and Y-axis, do not affect the phone heading estimate and the step direction measurement.[6] Experimental results show that incorporating the gyroscope bias with respect to the Z-axis leads to sufficiently accurate position estimates. Moreover, removing this assumption and extending the system model to include the biases with respect to all the three axes is straightforward.

4.3 Spatial Cooperation via Acoustic Signals

Users perform spatial cooperation by making range measurements and exchanging positional information. The spatial cooperation between a pair of users, user j and user i, consists of four stages: cooperation request, carrier sensing, acoustic ranging, and belief exchange. User j starts the spatial cooperation by setting up a Bluetooth connection with user i. After the connection is established, user j sends a cooperation request to user i and waits for its response. When the response is received, user j starts carrier sensing by turning on the microphone to detect the presence of acoustic signals transmitted by other users. If no such signals are detected, user j performs two-way ranging with user i by measuring the round-trip propagation time of acoustic signals [41]. When the ranging is finished, the two users exchange their positional beliefs using Bluetooth, update their position estimates as described in Algorithm 1, and close the Bluetooth connection. If there are multiple neighboring users available, user j performs spatial cooperation with them in succession. Note that the spatial cooperation requires only the sensors embedded in smartphones. Moreover, since the propagation speed of acoustic signals is slow compared to that of RF signals, desirable ranging accuracy can be achieved with the acoustic sampling rate of a smartphone. The latency of the spatial cooperation is as follows. The delay of the cooperation request ranges from 300 ms to 1.8 s, and the duration of carrier sensing is set to 400 ms. The delay of two-way ranging and belief exchange is 1.1 s. Thus, the total delay of the spatial cooperation is moderate for typical indoor applications.

5 Experimental Evaluation

In this section, we evaluate the performance of Mercury via experiments. The results are shown for both single-user scenarios and multi-user scenarios.

5.1 Experiment Setup

We implement Mercury using three different types of smartphones, namely Samsung Galaxy S4, LG Nexus 5, and Motorola Moto X (2nd Generation). The Galaxy S4 and Moto X models run Android version 4.4.4, whereas the Nexus 5 model runs Android version 5.0.1. The built-in sensors in these phones are used to obtain intra- and inter-user measurements for localization. In particular, the IMU provides acceleration and angular velocity samples at rate of

5. Since the number of points for the DFT is the same as the number of acceleration samples contained in a step, the first and second frequency component correspond to the first and second harmonic of the acceleration measurements, respectively, with the step frequency as the fundamental frequency.

6. The aim of this assumption is only for simplifying the model of the gyroscope biases. Our system also operates well if the Z-axis is not perpendicular to the ground.

TABLE 1
Values of Parameters in the System Model

Parameter	σ_{w}^2	ς_{e}^2 [s^{-2}]	σ_{l}^2 [m^2]	σ_{u}^2	σ_{r}^2 [m^2]
Value	7.62×10^{-5}	2.74×10^{-5}	0.09	0.024	0.036

100 Hz for the Galaxy S4 model, and 200 Hz for the Moto X and Nexus 5 models. In the spatial cooperation, we adopt acoustic chirp sequences consisting of 4,096 samples under the sampling rate of 44.1 kHz for making range measurements. The duration and bandwidth of such a sequence are 93 ms and 2.2 kHz, respectively. Based on the intra- and inter-user measurements, we obtain empirically the values of the parameters in the system model described in Section 2.2. These values are listed in Table 1.

To evaluate the performance of Mercury, we conduct experiments on the first floor and the sixth floor in the Stata Center at MIT. The first floor contains large open space, whereas the sixth floor is a typical office environment. The accessible areas of the first and sixth floor are partitioned into around 2,000 and 500 squares, respectively. The experiments are conducted in both single-user and multi-user scenarios. In single-user scenarios, only temporal cooperation is available, whereas in multi-user scenarios both temporal and spatial cooperation are available. In addition, the experiments are conducted under different settings of the prior distribution depending on whether the initial positions of the users and the phone headings are known. If the initial position of a user is unknown, the prior distribution of its position is uniform in the area of interest, and thus we set $N_{\mathrm{b}} = M$. If the initial position is known, we set N_{b} as 500 and 150 in the first and the sixth floor, respectively. Moreover, we set N_{g} and N_{c} to 3 and 30, respectively, unless otherwise noted (see Section 3.2 for the explanation of N_{b}, N_{g}, and N_{c}).

Mercury is compared with the systems described in [30] and [34] in the single-user scenarios, as spatial cooperation is not considered in the compared systems. The system in [30] is called MapCraft; the system in [34] uses the particle filtering (PF) technique and thus is referred to as PF. Both of the compared systems require discretized maps for the areas of interest, and for a fair comparison, they both use the ones obtained via the pre-processing described in Section 4.1. The step length and step direction measurements obtained by Mercury are used as the inputs of the two compared systems, and their variances, σ_{l}^2 and σ_{u}^2, are set to the same value as those in Mercury (see Table 1). Furthermore, even though WiFi RSS measurements can be incorporated in all the three systems, they are not taken into consideration in this paper for simplicity.

In MapCraft, the posterior distribution of the user trajectory is represented by a set of functions called potential functions. The logarithm of a potential function is the weighted sum of several feature functions that describe the dynamic and measurement models. We use the first two feature functions in [30], since the other feature functions are optional and require WiFi measurements. In particular, the first feature function is the product of (3), (4), and

$$g(u_k; \angle[\boldsymbol{p}_k - \boldsymbol{p}_{k-1}], \sigma_{\mathrm{u}}^2). \tag{15}$$

Compared with (5) used by Mercury, (15) does not consider the phone heading error w_k and its impact on the step direction measurement u_k. To mitigate such impact, the second feature function is adopted in MapCraft, and it is the product of (3), (4), and

$$g(u_k - \hat{w}_k; \angle[\boldsymbol{p}_k - \boldsymbol{p}_{k-1}], \sigma_{\mathrm{u}}^2),$$

where

$$\hat{w}_k = \frac{1}{W} \sum_{n=k-W}^{k-1} \big(u_n - \angle[\hat{\boldsymbol{p}}_n - \hat{\boldsymbol{p}}_{n-1}]\big),$$

with $\hat{\boldsymbol{p}}_n$ being an estimate of $\boldsymbol{p}_n$ and W being a constant. In other words, Mercury calibrates the step direction measurement u_k by subtracting the error $\hat{w}_k$ from it in the second feature function. The error $\hat{w}_k$ is computed using the position estimates in a previous time window containing W steps. The Viterbi algorithm is used in MapCraft to compute the state sequence of a user that achieves the maximum posterior distribution, and the complexity of updating the posterior distribution at each step scales as $\mathcal{O}(N_{\mathrm{b}})$. The performance of MapCraft depends on the value of W and the weights of the two feature functions. In this paper, W is chosen from the set $\{20, 40\}$. The weight λ_1 of the first feature function is chosen from $\{0.05, 0.1, 0.2, 0.4\}$, and the weight of the second feature is set to $1 - \lambda_1$. The best localization accuracy of MapCraft when W, λ_1, and λ_2 are chosen from the above values is compared with the performance of Mercury.[7]

In PF, the posterior distribution of a positional state is characterized by a set of weighted particles, and the values of these particles are chosen from a set of grids obtained via partitioning the area of interest. When a step is detected, a set of new particles are generated via a prediction stage, and their weights are computed in an update stage. The particle with the largest weight is the position estimate corresponding to the detected step. Specifically, in the prediction stage, the values of the new particles are computed based on the connectivity of the grids in the area of interest, whereas in the update stage, the weights of the new particles are computed according to the step direction and length measurements. The complexity of the PF technique scales linearly with respect to the number of particles, and we set such number to $N_{\mathrm{b}} N_{\mathrm{g}}$ so that the complexity of PF is similar to that of Mercury.

5.2 Experimental Results in Single-User Scenarios

We first conduct experiments to evaluate the accuracy of the step direction measurements. Specifically, a user walks along the corridor on the sixth floor for two consecutive laps as shown in Fig. 4a.[8] The user rotates the phone horizontally by 90 degrees after finishing the first lap. Fig. 4b shows the phone heading estimates and step direction measurements in one experiment. First, one can observe that desirable accuracy is obtained both before and after the phone is rotated. The average error of the step direction measurements is 8.9 degrees. Second, it can be seen that in the first lap, the step direction measurements and phone heading estimates are close to each other, whereas in the second lap there is a difference of about 90 degrees between them. This demonstrates the necessity to consider the heading offset to be time-varying.

We next evaluate the localization accuracy of Mercury in single-user scenarios. The user walks along the trajectory shown in Fig. 5. Fig. 5 also shows the estimated trajectory in one experiment when the initial position and phone

7. The feature function weights are obtained through a training process in [30]. Such a training process is not considered in this paper for fair comparisons.

8. The floor plans used in this paper are provided by the Department of Facilities, MIT.

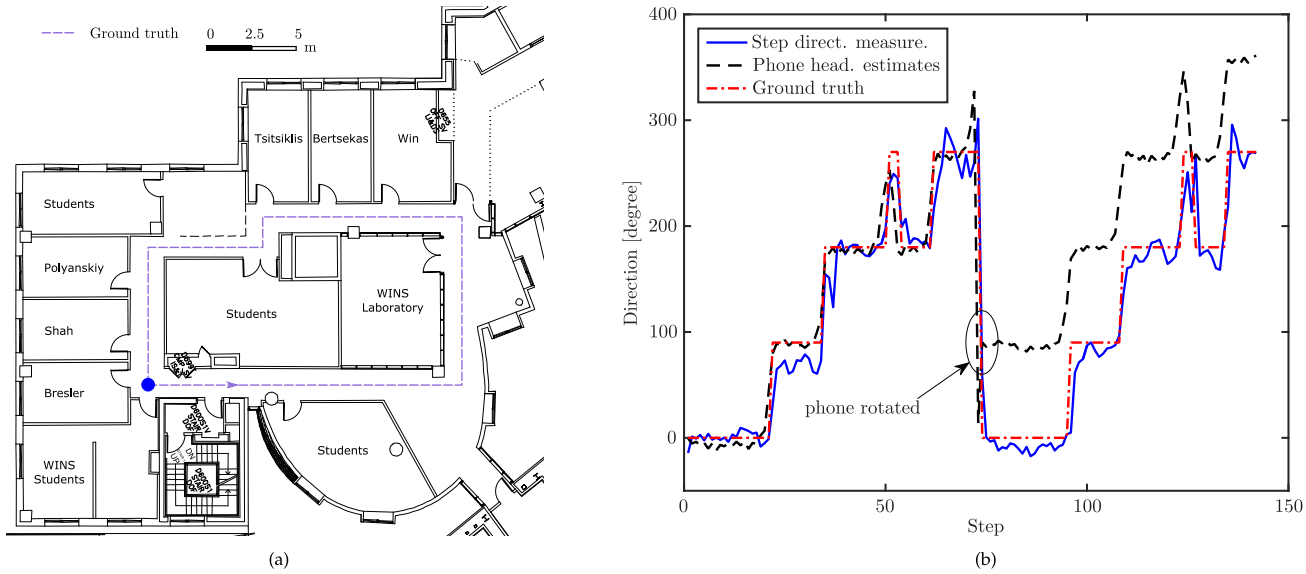


Fig. 4. Phone heading estimates and step direction measurements: (a) the trajectory of the user starting at the blue dot; (b) phone heading estimates and step direction measurements.

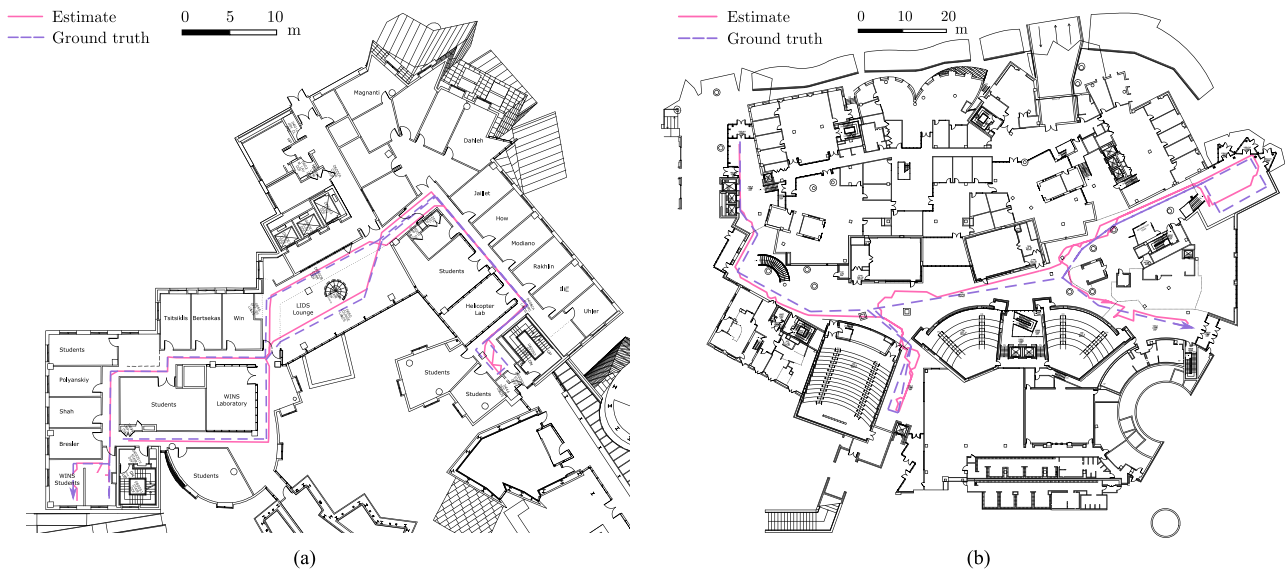


Fig. 5. The ground truth and estimated trajectory at the Stata Center, MIT: (a) on the sixth floor and (b) on the first floor.

heading are known. It can be seen that the estimated trajectory is very close to the ground truth.

Fig. 6 shows the cumulative distribution function (CDF) of the localization error when the initial state of the user is known. First, the 80th percentile of the localization error for Mercury is 1.6 and 3.5 m in the sixth and first floor, respectively. Second, it can be seen that both Mercury and the MapCraft system outperform the PF system. This is because the PF system does not take the accumulated error in the step direction measurements into account. Third, the performance on the sixth floor is better than that on the first floor. This is because more positional constraints can be extracted from the map information on the sixth floor.

Fig. 7 shows the CDF of the localization error when the initial position of the user is unknown and the initial phone heading is known. First, the 80th percentile of the localization error for Mercury is 1.7 and 3.9 m in the sixth and first floor, respectively. Second, since the initial position is unavailable, the localization accuracy is degraded compared with the scenario that the initial position is known. Fortunately, the position estimate quickly converges to the true position after a few steps, and therefore the overall performance is almost the same.

Fig. 8 shows the CDF of the localization error when the initial position of the user is known and the initial phone heading is unknown. The 80th percentile of the localization error for Mercury is 1.7 and 3.8 m in the sixth and first floor, respectively. Compared to the MapCraft system, Mercury is more robust to the unavailability of the initial phone heading information. In particular, the MapCraft system uses the position estimates in a previous time window to compute and correct the step direction measurement errors. When the previous position estimates are close to the true positions, such errors can be corrected. However, the errors cannot be correctly computed if the previous position estimates are unreliable, e.g., if the initial phone heading is unknown. In contrast, Mercury does not use position estimates in a previous window for measurement calibration directly.

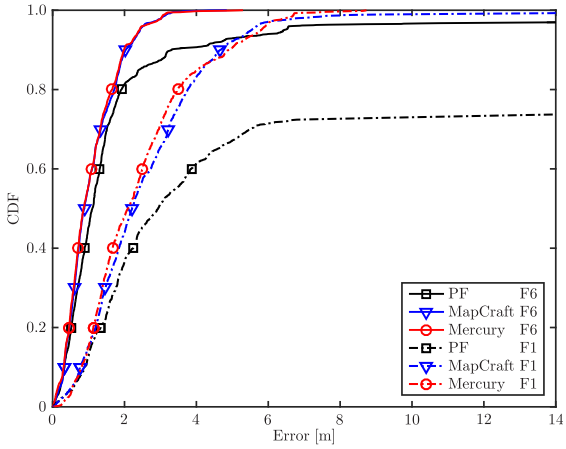


Fig. 6. The CDF of the localization error for Mercury, MapCraft, and the PF system in the sixth (F6) and first (F1) floor. The initial position and phone heading are known.

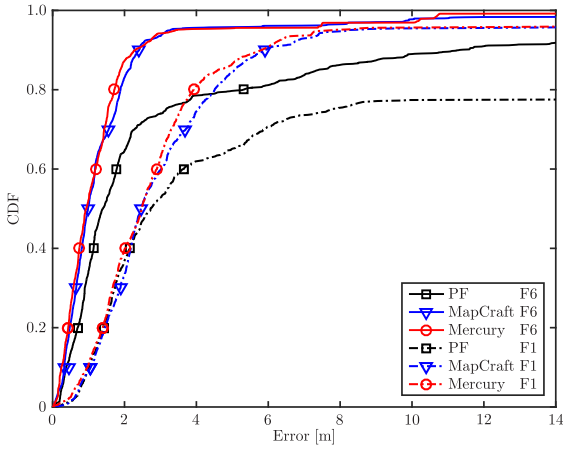


Fig. 7. The CDF of the localization error for Mercury, MapCraft, and the PF system in the sixth (F6) and first (F1) floor. Only the initial phone heading is known.

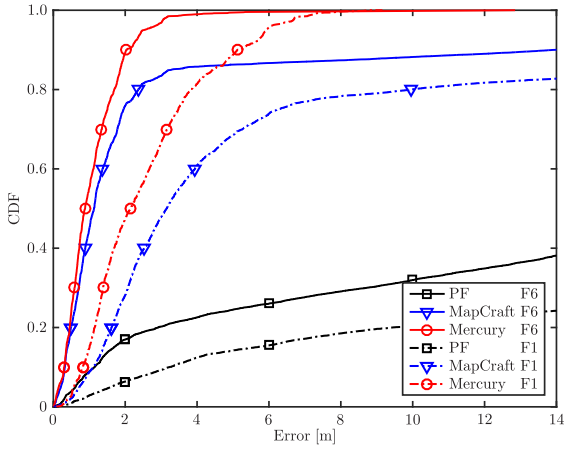


Fig. 8. The CDF of the localization error for Mercury, MapCraft, and the PF system in the sixth (F6) and first (F1) floor. Only the initial position is known.

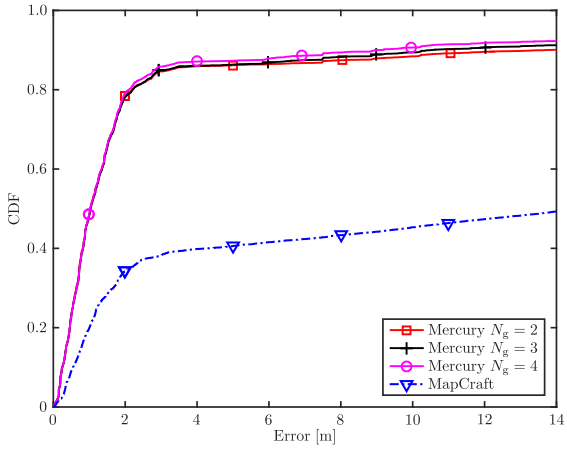


Fig. 9. The CDF of the localization error for Mercury and MapCraft on the sixth floor. Neither the initial position nor phone heading is known.

Instead, it incorporates the phone heading error in the state vector and updates its posterior distribution when intra-user measurements are available. With such a distribution, the effects of step direction measurement errors are mitigated when the positional belief is updated using the proposed BP algorithm. Therefore, Mercury is less sensitive to the information of the initial phone heading.

Fig. 9 shows the CDF of the localization error when neither the initial position nor the phone heading of the user is known. Different values of N_g are used in this scenario. First, even when there is no information about the initial state, Mercury still achieves 80th percentile of localization error of about 2.1 m when N_g is small. Second, the improvement from increasing N_g is not remarkable. In particular, the 80th percentile of localization error is 2.1 and 2.2 m when N_g is 4 and 2, respectively. Therefore, we use a small value of N_g in Mercury.

The experimental results in single-user scenarios demonstrate the efficiency of the BP algorithm in Mercury compared with conventional filtering techniques for indoor localization. Specifically, we extend the state vector in Mercury by incorporating the phone heading error and gyroscope bias. Since the conditional distribution of these two random variables given the user's position can be represented by a Gaussian mixture distribution, and the parameters of such a distribution can be computed easily, the extra computational overhead of extending the state vector is small. Compared with filtering techniques (e.g., the PF technique) whose complexity grows fast with respect to the dimension of the state vector, our algorithm is more amenable for infrastructure-free localization systems. In addition, experimental results show that Mercury is more robust than MapCraft when the initial positional knowledge is imperfect, and its computational complexity is only slightly higher than that of MapCraft.

5.3 Experimental Results in Multi-User Scenarios

We next evaluate the localization accuracy of Mercury in multi-user scenarios with two experiments. The first experiment is conducted on the sixth floor with user 1 and user 2, and the second one is conducted on the first floor with user 1, user 2, and user 3. User 1 walks along the trajectory in Fig. 5, and user 2 and user 3 walk along the trajectories in Fig. 10. The users perform spatial cooperation intermittently as they walk, and the cooperation between a pair of users is represented by a red arrow in Fig. 10. In the first experiment, user 1 knows both its initial position and phone heading, whereas user 2 knows neither. In the second experiment, user 1 knows both its initial position and phone heading; user 2 knows only its initial phone heading; user 3 knows neither.

Fig. 10. The trajectories of user 2 and user 3 in the multi-user scenario: (a) on the sixth floor and (b) on the first floor. Solid red arrows represent spatial cooperation.

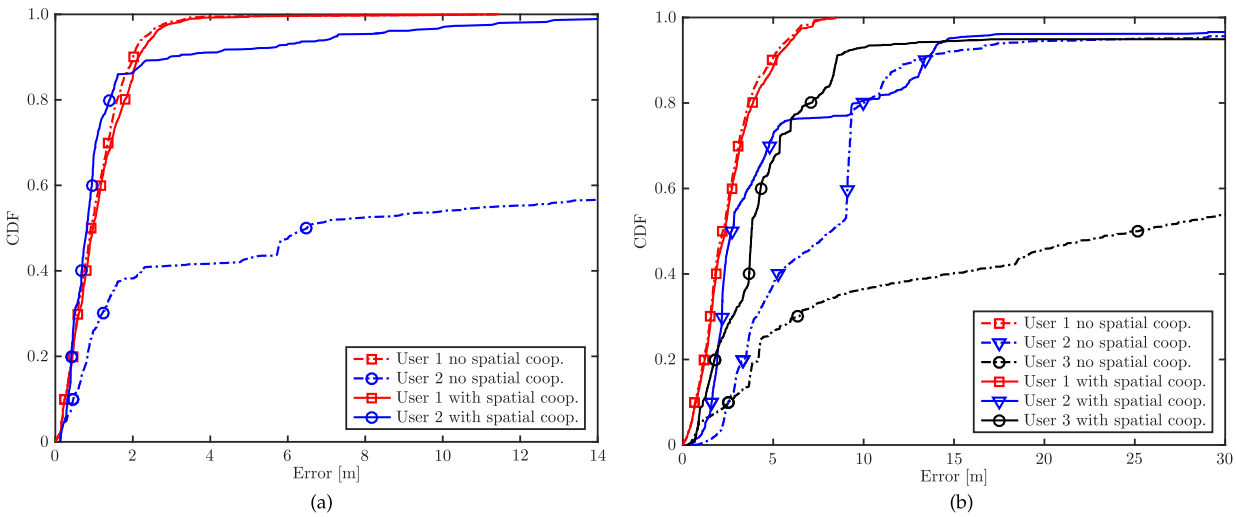


Fig. 11. The CDF of the localization error for Mercury with and without spatial cooperation: (a) on the sixth floor and (b) on the first floor.

Fig. 11 shows the CDF of the localization error for the setting with and without spatial cooperation. First, desirable localization accuracy is achieved in both experiments with spatial cooperation. In the first experiment, the median localization error of user 1 and user 2 are 1.0 and 0.8 m, respectively; in the second experiment, the median localization error of user 1, user 2, and user 3 are 2.3, 2.7, and 3.8 m, respectively. Second, spatial cooperation benefits the localization performance remarkably if the initial positional state is entirely or partly unknown. Specifically, the CDFs of the localization error for user 2 and user 3 without spatial cooperation almost first-order stochastically dominate those with spatial cooperation.[9] Moreover, the median localization errors of user 2 are reduced from 6.5 and 8.2 m to 0.8 and 2.7 m in the first and second experiment, respectively; the median localization errors of user 3 are reduced from 25.2 to 3.8 m in the second experiment.[10]

5.4 Evaluation on the Robustness of Mercury

We next evaluate the robustness of Mercury when the system runs for a long time period on the zeroth floor of the Stata Center. The user trajectory has longer distance, contains more turns, and covers a larger region on the zeroth floor compared with trajectories in experiments conducted on the first and sixth floor. Specifically, the floor plan of the zeroth floor is partitioned into 4,748 squares for Mercury. In the single-user scenario, user 1 walks along the trajectory shown in Fig. 12 for two laps with total length of 832 m. We

9. A CDF $F_1(x)$ first-order stochastically dominates CDF $F_2(x)$ if $F_1(x) \leq F_2(x)$ for all x with strict inequality over some interval.

10. The performance of user 2 and user 3 are worse than user 1 when the initial position and phone heading are unknown and spatial cooperation is unavailable. This is because the information extracted from the intra-user measurements and the map is less sufficient as the trajectories of user 2 and user 3 are shorter than that of user 1.

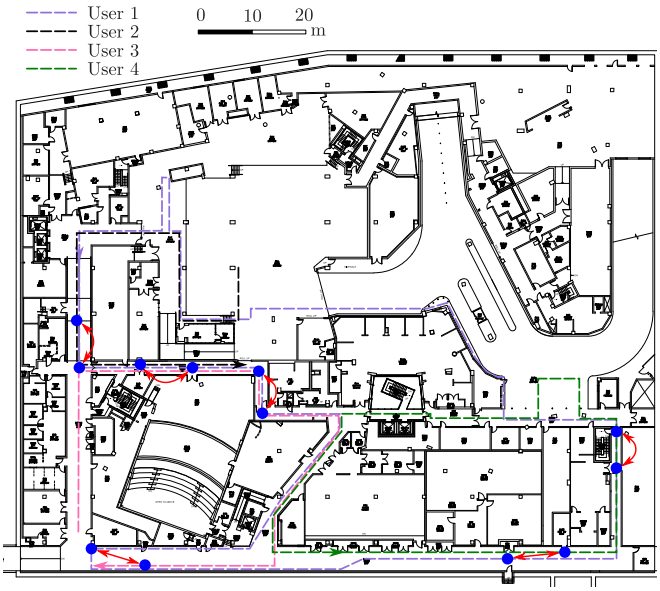


Fig. 12. The trajectories of user 1 – 4 on the zeroth floor. Solid red arrows represent spatial cooperation.

evaluate the localization performance under two different settings: the information of the initial position and phone heading is perfectly known and unknown. In the multi-user scenario, three additional users, user $2-4$, are included in the experiments. They walk along the trajectories for one lap shown in Fig. 12 and perform spatial cooperation with user 1. The initial position and phone heading of user 1 are unknown, whereas those of user $2-4$ are known. For both the single-user and multi-user scenarios, we evaluate the localization performance when the compass measurements are and are not incorporated.

Fig. 13 shows the CDF of the localization error in the single-user scenario. When the initial position and phone heading are known, the 80th percentile of localization error are 3.6 and 3.8 m for the setting with and without compass measurements, respectively. When the initial position and phone heading are unknown, the 80th percentile of localization error are 4.1 and 4.2 m for the setting with and without compass measurements, respectively. These results validate the robustness of Mercury when the trajectory of the user is complicated and the area of interest has a large size. Moreover, it can be observed that the compass measurements can improve the localization accuracy.

Fig. 14 shows the CDF of the localization error of user 1 in the multi-user scenario. Spatial cooperation reduces the 80th and 90th percentile of localization errors from 4.2 and 6.5 m to 4.1 and 5.1 m, respectively, without compass measurements, and spatial cooperation reduces these errors from 4.1 and 5.9 m to 3.8 and 4.7 m, respectively, with compass measurements. Compared with the results obtained in the sixth and first floor shown in Fig. 11, the performance gain of spatial cooperation is less remarkable on the zeroth floor with a longer trajectory of the user. This is because better localization performance is achieved without spatial cooperation when the trajectory is longer. In particular, with a longer trajectory, more information about the positional states of the user can be extracted from the intra-user measurements as well the map constraints. With such information, Mercury can track the position of the user after the first several steps even when the initial position and phone heading are unknown.

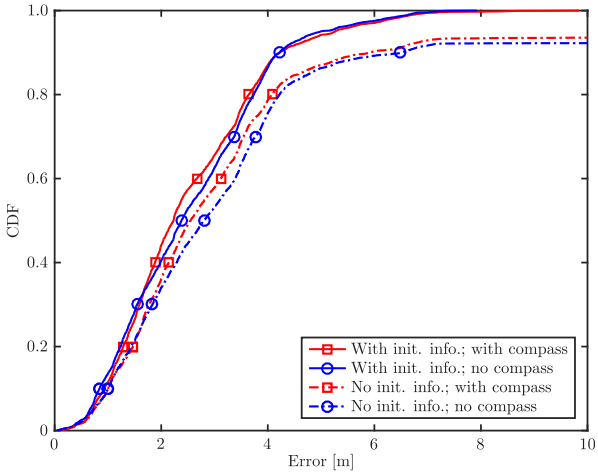


Fig. 13. The CDF of the localization error for Mercury in the single-user scenario on the zeroth floor.

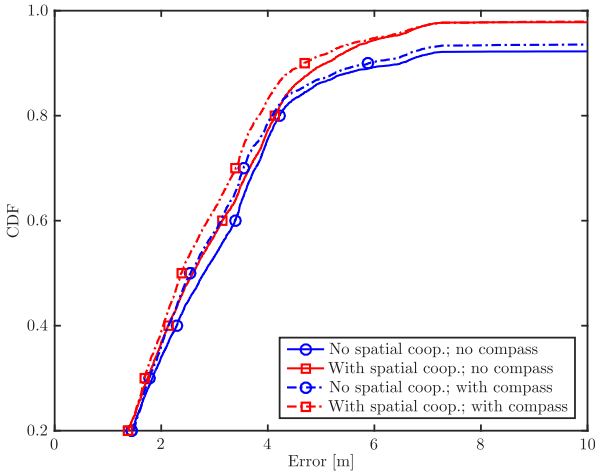


Fig. 14. The CDF of the localization error for Mercury in the multi-user scenario on the zeroth floor.

TABLE 2
Computational Overhead of Mercury

N_b	Model	CPU (%)	Mem. [MB]	Exe. Time [ms]
150	Galaxy S4	4.8 – 10.6	18.6	82
	Nexus 5	6.8 – 15.3	28.3	43
	Moto X	6.0 – 11.1	25.8	43
500	Galaxy S4	6.0 – 15.6	18.7	218
	Nexus 5	7.3 – 18.9	29.8	101
	Moto X	9.5 – 13.5	30.0	172

5.5 Computational Overhead and Execution Time

We next evaluate the computational overhead and execution time of Mercury. We measure the central processing unit (CPU) usage and memory usage using the Android Studio integrated development environment. We also measure the execution time it takes for updating the position estimate when a new step is detected. The results for $N_b = 150$ and $N_b = 500$ are summarized in Table 2. It can be seen that both the CPU usage and the execution time increase with N_b. On the other hand, the memory usage does not change much with N_b. Instead, it depends on the smartphone model and the operating system version. For both values of N_b, the

amount of used memory is below 30 MB. This shows that the memory consumption of Mercury is moderate, as modern smartphones have memory of no less than 1 GB. The maximum CPU usage is below 20 percent for all the three models. This shows that the computational overhead of Mercury is acceptable. The execution time increases with the parameter N_b, and it is below 220 ms for all the three models. This verifies that Mercury is responsive and is able to provide position estimates in real time.

6 Related Work

In this section, we present related work on navigation with inertial measurements and spatial cooperation with range measurements.

6.1 Navigation with Inertial Measurements

Inertial measurements provide information about the movement of a user [29]. They contain errors caused by gyroscope bias and noise, and these errors can degrade the localization performance remarkably if they are not handled appropriately. Different methods have been proposed to obtain position estimates using inertial measurements and mitigate the effects of the errors.

Most existing methods formulate the navigation problem with inertial measurements within the Bayesian estimation framework. In such a framework, the joint distribution of the positional states and the measurements are described with a dynamic model, a measurement model, and a prior distribution model [47]. Based on these models, the Bayesian filtering techniques are applied to compute the posterior distribution of the current positional state given the available measurements. The Kalman filtering method is an efficient technique for determining such a distribution [38], [42], [48]. In this method, the posterior distribution of the positional state is approximated by a Gaussian distribution. The mean and covariance matrix characterizing the Gaussian distribution are updated when new measurements are available. Moreover, the zero-velocity update (ZUPT) method can be applied to mitigate the impact of the errors in the inertial measurements [42], [48]. This method calibrates the posterior distribution of the positional state when a user's foot is detected to strike the ground. However, this method requires that a user mounts an IMU onto its shoes and thus its practicality is limited. In addition, since the posterior distribution of the user's position is approximated by a Gaussian distribution, it is non-trivial to incorporate the map information appropriately.

Another method for inferring the posterior distribution of a user's position is the PF technique [29], [34], [35], [37]. In this method, the state vector contains the user's position, and its posterior distribution is approximated by the weighted sum of Dirac delta functions. These Dirac delta functions are also referred to as particles. The map information can be incorporated into this method to combat the inertial measurement errors. In particular, the weights of particles that do not satisfy the map constraints are set to zero [29], [35]. The computational complexity of the PF technique depends on the number of particles used to approximate the actual distribution of the state. This number grows fast with the dimension of the state if one needs a tight approximation [29]. Therefore, it is non-trivial to incorporate extra variables, such as the phone heading error and gyroscope bias, into the state vector with moderate computational overhead.

Recently it has been shown that the navigation problem can be formulated using the graphical model of conditional random field (CRF) [30]. In this model, the posterior distribution of the positional states is represented by a set of functions called feature functions describing the relation between the measurements and the states. To mitigate the effects of the inertial measurement errors, the position estimates in a previous time window is used to calibrate the step direction measurements, as described in Section 5.1. Little is known on how to incorporate spatial cooperation into the framework of CRF.

6.2 Spatial Cooperation

It has been shown that spatial cooperation among different users can benefit the localization performance remarkably [2], [3], [26]. A widely adopted technique for realizing spatial cooperation is UWB radio [3], [13], [48], [49], [50]. This technique requires customized hardware to provide accurate range measurements between users. However, due to the limited hardware capability of smartphones, it is difficult to obtain range measurements using RF signals. Another technique to realize spatial cooperation is employing acoustic signals for making inter-user measurements [39], [40], [41], [51], [52]. Since acoustic signals propagate more slowly than RF signals, desirable inter-user measurements can be obtained through two-way ranging with only a pair of smartphones.

The inter-user measurements can be fused with other types of measurements in different manners [38], [39], [40], [48], [51], [53]. In [39], a pair of users send the range measurements between them as well as their own displacement measurements obtained via IMU to a server periodically. The server combines the measurements sent by each pair of users, computes their geometric relationship, and determine their relative positions. In [40], the inter-user range measurements are combined with WiFi RSS measurements for improved localization accuracy. In particular, a user first estimates its position using the fingerprinting method according to the WiFi RSS measurements. The user then makes range measurements with neighboring users via acoustic signals, and calibrates its position estimate accordingly. In [51] and [53], the BP method is adopted for processing inter-user measurements and obtaining user's positions in scenarios where users are static. Specifically, in [51], a server collects all the range measurements made by neighboring users and runs a centralized BP method to infer users' positions. In [53], a user exchanges positional beliefs with neighboring users, and adopts the distributed BP method to estimate its position. Even though the BP method is applied in these two papers, the movement of users is not taken into account. The Kalman filtering technique has also been applied for incorporating the spatial cooperation. In [38], a user performs spatial cooperation by broadcasting sinusoid acoustic signals periodically and detecting the signals transmitted by its neighboring users. The frequency of the acoustic signal sent by a user is associated with the estimate of the user's position. Therefore, a user can determine the position estimates of its neighboring users according to the signals it receives. The average of the neighboring users' position estimates is then used as a measurement of the user's own position in the Kalman filtering technique. In [48], a user makes range measurements opportunistically using UWB radios. The range measurements and the inertial measurements

obtained from a shoe-mounted IMU are fused using the Kalman filtering technique to estimate the user's position. The map information is not taken into account in [38] and [48].

7 Conclusion

In this paper, we presented Mercury, a network localization and navigation system consisting of only smartphones. This system exploits positional information from spatiotemporal cooperation and environmental knowledge in a principled manner. In particular, we designed a graphical model and developed a BP algorithm accordingly. In the graphical model, each state vertex represents a user's position, the phone heading error, and the gyroscope bias. The proposed BP algorithm fuses intra-user measurements and inter-user measurements, where the former includes acceleration and angular velocity samples via IMU, and the latter includes range measurements via acoustic signals. We fused the map information into the BP algorithm by imposing positional constraints. Extensive experiments were carried out in both single-user and multi-user scenarios and the performance of Mercury was compared with that of existing systems. In the singer-user scenario, the 80th percentile of the localization error in the sixth and first floor of the Stata Center is 1.6 and 3.5 m, respectively. Moreover, the performance of Mercury is more robust than that of existing systems in the presence of imperfect initial positional knowledge. In multi-user scenarios, the localization performance is remarkably improved by spatial cooperation, especially for users whose initial positional knowledge is entirely or partly unknown. Mercury is a prototype of NLN implemented with only smartphones, and the results demonstrate its advantages thanks to the exploitation of spatiotemporal cooperation as well as environmental knowledge.

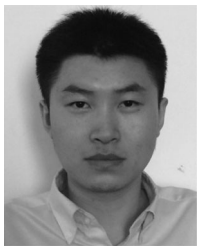
Acknowledgments

This research was supported, in part, by the Office of Naval Research under Grant N00014-16-1-2141 and the MIT Institute for Soldier Nanotechnologies. The authors wish to thank S. Bartoletti, R. Cohen, A. Conti, G. C. Ferrante, S. Mazuelas, F. Meyer, F. Wang, and T. Wang for their fruitful discussions and careful reading of the manuscript.

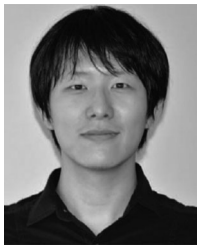
References

[1] M. Z. Win, et al., "Network localization and navigation via cooperation," *IEEE Commun. Mag.*, vol. 49, no. 5, pp. 56–62, May 2011.
[2] Y. Shen, S. Mazuelas, and M. Z. Win, "Network navigation: Theory and interpretation," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 9, pp. 1823–1834, Oct. 2012.
[3] A. Conti, M. Guerra, D. Dardari, N. Decarli, and M. Z. Win, "Network experimentation for cooperative localization," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 2, pp. 467–475, Feb. 2012.
[4] Y. Qi, H. Kobayashi, and H. Suda, "Analysis of wireless geolocation in a non-line-of-sight environment," *IEEE Trans. Wireless Commun.*, vol. 5, no. 3, pp. 672–681, Mar. 2006.
[5] S. Gezici, et al., "Localization via ultra-wideband radios: A look at positioning aspects for future sensor networks," *IEEE Signal Process. Mag.*, vol. 22, no. 4, pp. 70–84, Jul. 2005.
[6] Y. Shen and M. Z. Win, "Fundamental limits of wideband localization – Part I: A general framework," *IEEE Trans. Inf. Theory*, vol. 56, no. 10, pp. 4956–4980, Oct. 2010.
[7] F. Zabini and A. Conti, "Inhomogeneous Poisson sampling of finite-energy signals with uncertainties in $\mathbb{R}^d$," *IEEE Trans. Signal Process.*, vol. 64, no. 18, pp. 4679–4694, Sep. 2016.
[8] J. Shen and A. F. Molisch, "Estimating multiple target locations in multi-path environments," *IEEE Trans. Wireless Commun.*, vol. 13, no. 8, pp. 4547–4559, Aug. 2014.
[9] A. Conti, D. Dardari, M. Guerra, L. Mucchi, and M. Z. Win, "Experimental characterization of diversity navigation," *IEEE Syst. J.*, vol. 8, no. 1, pp. 115–124, Mar. 2014.
[10] S. Bartoletti, A. Giorgetti, M. Z. Win, and A. Conti, "Blind selection of representative observations for sensor radar networks," *IEEE Trans. Veh. Technol.*, vol. 64, no. 4, pp. 1388–1400, Apr. 2015.
[11] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: Part I," *IEEE Robot. Autom. Mag.*, vol. 13, no. 2, pp. 99–110, Jun. 2006.
[12] M. Leigsnering, M. Amin, F. Ahmad, and A. M. Zoubir, "Multipath exploitation and suppression for SAR imaging of building interiors: An overview of recent advances," *IEEE Signal Process. Mag.*, vol. 31, no. 4, pp. 110–119, Jul. 2014.
[13] N. Patwari, J. N. Ash, S. Kyperountas, A. O. Hero, R. L. Moses, and N. S. Correal, "Locating the nodes: Cooperative localization in wireless sensor networks," *IEEE Signal Process. Mag.*, vol. 22, no. 4, pp. 54–69, Jul. 2005.
[14] S. Bartoletti, A. Conti, A. Giorgetti, and M. Z. Win, "Sensor radar networks for indoor tracking," *IEEE Wireless Commun. Lett.*, vol. 3, no. 2, pp. 157–160, Apr. 2014.
[15] P. Lazik, N. Rajagopal, O. Shih, B. Sinopoli, and A. Rowe, "ALPS: A bluetooth and ultrasound platform for mapping and localization," in *Proc. 13th ACM Conf. Embedded Netw. Sensor Syst.*, Nov. 2015, pp. 73–84.
[16] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, "The cricket location-support system," in *Proc. 6th Annu. Int. Conf. Mobile Comput. Netw.*, 2000, pp. 32–43.
[17] C. Wu, Z. Yang, and Y. Liu, "Smartphones based crowdsourcing for indoor localization," *IEEE Trans. Mobile Comput.*, vol. 14, no. 2, pp. 444–457, Feb. 2015.
[18] S. Sen, B. Radunović, R. R. Choudhury, and T. Minka, "You are facing the Mona Lisa: Spot localization using PHY layer information," in *Proc. ACM Int. Conf. Mobile Syst. Appl. Services*, Jun. 2012, pp. 183–196.
[19] Y. Chen, D. Lymberopoulos, J. Liu, and B. Priyantha, "FM-based indoor localization," in *Proc. ACM Int. Conf. Mobile Syst. Appl. Services*, Jun. 2012, pp. 169–182.
[20] V. Otsason, A. Varshavsky, A. LaMarca, and E. de Lara, "Accurate GSM indoor localization," in *Proc. ACM Int. Conf. Ubiquitous Comput.*, Sep. 2005, pp. 141–158.
[21] S. Sen, D. Kim, S. Laroche, K.-H. Kim, and J. Lee, "Bringing CUPID indoor positioning system to practice," in *Proc. Int. World Wide Web Conf.*, May 2015, pp. 938–948.
[22] T. Bourchas, M. Bednarek, D. Giustiniano, and V. Lenders, "Poster abstract: Practical limits of WiFi time-of-flight echo techniques," in *Proc. IEEE Inf. Process. Sensor Netw.*, Apr. 2014, pp. 273–274.
[23] D. Giustiniano and S. Mangold, "CAESAR: Carrier sense-based ranging in off-the-shelf 802.11 wireless LAN," presented at the ACM Conf. Emerging Netw. Experiments Technol., Tokyo, Japan, Dec. 2011.
[24] W. W.-L. Li, R. A. Iltis, and M. Z. Win, "A smartphone localization algorithm using RSSI and inertial sensor measurement fusion," in *Proc. IEEE Global Telecommun. Conf.*, Dec. 2013, pp. 3357–3362.
[25] K. Wu, J. Xiao, Y. Yi, M. Gao, and L. M. Ni, "FILA: Fine-grained indoor localization," in *Proc. IEEE Conf. Comput. Commun.*, Mar. 2012, pp. 2210–2218.
[26] Y. Shen, H. Wymeersch, and M. Z. Win, "Fundamental limits of wideband localization – Part II: Cooperative networks," *IEEE Trans. Inf. Theory*, vol. 56, no. 10, pp. 4981–5000, Oct. 2010.
[27] H. Wymeersch, J. Lien, and M. Z. Win, "Cooperative localization in wireless networks," *Proc. IEEE*, vol. 97, no. 2, pp. 427–450, Feb. 2009.
[28] F. Montorsi, S. Mazuelas, G. M. Vitetta, and M. Z. Win, "On the performance limits of map-aware localization," *IEEE Trans. Inf. Theory*, vol. 59, no. 8, pp. 5023–5038, Aug. 2013.
[29] O. Woodman and R. Harle, "Pedestrian localisation for indoor environments," in *Proc. ACM Int. Conf. Ubiquitous Comput.*, Sep. 2008, pp. 114–123.
[30] Z. Xiao, H. Wen, A. Markham, and N. Trigoni, "Indoor tracking using undirected graphical models," *IEEE Trans. Mobile Comput.*, vol. 14, no. 11, pp. 2286–2301, Feb. 2015.
[31] R. Harle, "A survey of indoor inertial positioning systems for pedestrians," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 3, pp. 1281–1293, Jul.–Sep. 2013.

[32] H. Nurminen, A. Ristimäki, S. Ali-Löytty, and R. Piché, "Particle filter and smoother for indoor localization," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat.*, Oct. 2013, pp. 1–10.
[33] F. Evennou, F. Marx, and E. Novakov, "Map-aided indoor mobile positioning system using particle filter," in *Proc. IEEE Wireless Commun. Netw. Conf.*, Mar. 2005, pp. 2490–2494.
[34] S. Hilsenbeck, D. Bobkov, G. Schroth, R. Huitl, and E. Steinbach, "Graph-based data fusion of pedometer and WiFi measurements for mobile indoor positioning," in *Proc. ACM Int. Conf. Ubiquitous Comput.*, Sep. 2014, pp. 147–158.
[35] F. Li, C. Zhao, G. Ding, J. Gong, C. Liu, and F. Zhao, "A reliable and accurate indoor localization method using phone inertial sensors," in *Proc. ACM Int. Conf. Ubiquitous Comput.*, Sep. 2012, pp. 421–430.
[36] H. Wang, S. Sen, A. Elgohary, M. Farid, M. Youssef, and R. R. Choudhury, "No need to war-drive: Unsupervised indoor localization," in *Proc. ACM Int. Conf. Mobile Syst. Appl. Services*, Jun. 2012, pp. 197–210.
[37] M. Angermann and P. Robertson, "FootSLAM: Pedestrian simultaneous localization and mapping without exteroceptive sensors – hitchhiking on human perception and cognition," *Proc. IEEE*, vol. 100, pp. 1840–1848, May 2012.
[38] Z. Yang, X. Feng, and Q. Zhang, "Adometer: Push the limit of pedestrian indoor localization through cooperation," *IEEE Trans. Mobile Comput.*, vol. 13, no. 11, pp. 2473–2483, Nov. 2014.
[39] L. Zhang, et al., "Montage: Combine frames with movement continuity for realtime multi-user tracking," *IEEE Trans. Mobile Comput.*, vol. 16, no. 4, pp. 1019–1031, Apr. 2017.
[40] H. Liu, J. Yang, S. Sidhom, Y. Wang, Y. Chen, and F. Ye, "Accurate WiFi based localization for smartphones using peer assistance," *IEEE Trans. Mobile Comput.*, vol. 13, no. 10, pp. 2199–2214, Oct. 2014.
[41] C. Peng, G. Shen, Y. Zhang, Y. Li, and K. Tan, "BeepBeep: A high accuracy acoustic ranging system using cots mobile devices," in *Proc. ACM Conf. Embedded Netw. Sensor Syst.*, Nov. 2007, pp. 1–14.
[42] E. Foxlin, "Pedestrian tracking with shoe-mounted inertial sensors," *IEEE Comput. Graph. Appl.*, vol. 25, no. 6, pp. 38–46, Nov./Dec. 2005.
[43] J. Prieto, S. Mazuelas, and M. Z. Win, "Context-aided inertial navigation via belief condensation," *IEEE Trans. Signal Process.*, vol. 64, no. 12, pp. 3250–3261, Jun. 2016.
[44] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo, CA, USA: Morgan Kaufmann, 1988.
[45] S. O. Madgwick, A. J. Harrison, and R. Vaidyanathan, "Estimation of IMU and MARG orientation using a gradient descent algorithm," in *Proc. IEEE Int. Conf. Rehabil. Robot.*, Jun. 2011, pp. 1–7.
[46] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen, "Zee: Zero-effort crowdsourcing for indoor localization," in *Proc. ACM Int. Conf. Mobile Comput. Netw.*, Aug. 2012, pp. 293–304.
[47] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174–188, Feb. 2002.
[48] J. Rantakokko, et al., "Accurate and reliable soldier and first responder indoor positioning: Multisensor systems and cooperative localization," *IEEE Wireless Commun. Mag.*, vol. 18, no. 2, pp. 10–18, Apr. 2011.
[49] S. Bartoletti, W. Dai, A. Conti, and M. Z. Win, "A mathematical model for wideband ranging," *IEEE J. Sel. Topics Signal Process.*, vol. 9, no. 2, pp. 216–228, Mar. 2015.
[50] D. Dardari, A. Conti, U. J. Ferner, A. Giorgetti, and M. Z. Win, "Ranging with ultrawide bandwidth signals in multipath environments," *Proc. IEEE*, vol. 97, no. 2, pp. 404–426, Feb. 2009.
[51] R. Nandakumar, K. K. Chintalapudi, and V. N. Padmanabhan, "Centaur: Locating devices in an office environment," in *Proc. ACM Int. Conf. Mobile Comput. Netw.*, Aug. 2012, pp. 281–292.
[52] K. Liu, X. Liu, and X. Li, "Guoguo: Enabling fine-grained smartphone localization via acoustic anchors," *IEEE Trans. Mobile Comput.*, vol. 15, no. 5, pp. 1144–1156, May 2016.
[53] A. T. Ihler, J. W. Fisher III, R. L. Moses, and A. S. Willsky, "Nonparametric belief propagation for self-localization of sensor networks," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 4, pp. 809–819, Apr. 2005.



Zhenyu Liu (S'15) received the BS (with honor) and MS degrees in electronic engineering from Tsinghua University, Beijing, China, in 2011 and 2014, respectively. He is working toward the PhD degree in the Wireless Information and Network Sciences Laboratory, Massachusetts Institute of Technology (MIT), Cambridge, Massachusetts. His research interests include wireless communication, network localization, and stochastic optimization. He received the first prize of the IEEE Communications Society Student Competition in 2016. He received academic excellence scholarships from 2008 to 2010 from Tsinghua University. He served as a reviewer of the *Proceedings of the IEEE*, the *IEEE Transactions on Vehicular Technology*, the *IEEE Communications Letters*, and the *IEEE Wireless Communications Letters*. He is a student member of the IEEE.



Wenhan Dai (S'12) received the BS degrees in electronic engineering and mathematics from Tsinghua University in 2011, and the SM degree in aeronautics and astronautics at the Massachusetts Institute of Technology (MIT) in 2014. He is pursuing the PhD degree with the Wireless Information and Network Sciences Laboratory at MIT. His research interests include communication theory and stochastic optimization with application to wireless communication and network localization. His current research focuses on efficient network localization, cooperative network operation, and ultra-wideband communication. Mr. Dai was honored by the Marconi Society with the Paul Baran Young Scholar Award (2017). He received the Marconi-BISITE Best Paper Award from the IEEE ICUWB (2017), the Chinese Government Award for Outstanding Student Abroad (2016), the first prize of the IEEE Communications Society Student Competition (2016), and the Student Paper Award (first place) from the IEEE CWIT (2015). He was recognized as the exemplary reviewer of the *IEEE Communications Letters* (2014).

Moe Z. Win (S'85-M'87-SM'97-F'04) is a professor at the Massachusetts Institute of Technology (MIT) and the founding director of the Wireless Information and Network Sciences Laboratory. Prior to joining MIT, he was with AT&T Research Laboratories and the NASA Jet Propulsion Laboratory. His research encompasses fundamental theories, algorithm design, and network experimentation for a broad range of real-world problems. His current research topics include network localization and navigation, network interference exploitation, and quantum information science. He has served the IEEE Communications Society as an elected Member-at-Large on the Board of Governors, as elected Chair of the Radio Communications Committee, and as an IEEE Distinguished Lecturer. Over the last two decades, he held various editorial posts for IEEE journals and organized numerous international conferences. Currently, he is serving on the SIAM Diversity Advisory Committee. He is an elected Fellow of the AAAS, the IEEE, and the IET. He was honored with two IEEE Technical Field Awards: the IEEE Kiyo Tomiyasu Award (2011) and the IEEE Eric E. Sumner Award (2006, jointly with R. A. Scholtz). Together with students and colleagues, his papers have received numerous awards. Other recognitions include the IEEE Communications Society Edwin H. Armstrong Achievement Award (2016), the International Prize for Communications Cristoforo Colombo (2013), the Copernicus Fellowship (2011) and the *Laurea Honoris Causa* (2008) from the University of Ferrara, and the U.S. Presidential Early Career Award for Scientists and Engineers (2004). He is an ISI Highly Cited Researcher.