A Scalable Algorithm for Network Localization and Synchronization

Florian Meyer, *Member, IEEE*, Bernhard Etzlinger, *Member, IEEE*, Zhenyu Liu, *Student Member, IEEE*, Franz Hlawatsch, *Fellow, IEEE*, and Moe Z. Win, *Fellow, IEEE*

***Abstract*—The Internet of Things (IoT) will seamlessly integrate a large number of densely deployed heterogeneous devices and will enable new location-aware services. However, fine-grained localization of IoT devices is challenging as their computation and communication resources are typically limited and different devices may have different qualities of internal clocks and different mobility patterns. To address these challenges, we propose a cooperative, scalable, and time-recursive algorithm for network localization and synchronization (NLS). Our algorithm is based on time measurements and supports heterogeneous devices with limited computation and communication resources, time-varying clock and location parameters, arbitrary state-evolution models, and time-varying network connectivity. These attributes make the proposed algorithm attractive for IoT-related applications. The algorithm is furthermore able to incorporate measurements from additional sensors for positioning, navigation and timing (PNT) such as receivers for global navigation satellite systems (GNSSs). Based on a factor graph representation of the underlying spatiotemporal Bayesian sequential estimation problem, the algorithm uses belief propagation (BP) for an efficient marginalization of the joint posterior distribution. To account for the nonlinear measurement model and nonlinear state-evolution models while keeping the communication and computation requirements low, we develop an efficient second-order implementation of the BP rules by means of the recently introduced sigma point belief propagation (SPBP) technique. Simulation results demonstrate the high synchronization and localization accuracy as well as the low computational complexity of the proposed algorithm. In particular, in sufficiently dense networks, the proposed algorithm outperforms the state-of-the-art BP-based algorithm for NLS in terms of both estimation accuracy and computational complexity.**

***Index Terms*—Internet of Things, network synchronization, network localization, belief propagation, message passing, factor graph, distributed estimation, cooperative localization, cooperative synchronization.**

Manuscript received September 1, 2017; revised December 6, 2017.

This work was supported, in part, by the Austrian Science Fund (FWF) under grants J3886-N31 and P27370-N30, by the King Abdullah University of Science and Technology through the Sensor Research Initiative Grant OSR-2015-Sensors-2700, by the U.S. Department of Commerce, National Institute of Standards and Technology under Grant 70NANB17H17, by the Czech National Sustainability program under grant LO1401, and by the Linz Center of Mechatronics (LCM) in the framework of the Austrian COMET-K2 program.

F. Meyer, Z. Liu, and Moe Z. Win are with the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA, USA (e-mail: fmeyer@mit.edu, zliu14@mit.edu, moewin@mit.edu). B. Etzlinger is with the Institute for Communications Engineering and RF-Systems, Johannes Kepler University, Linz, Austria (email: bernhard.etzlinger@jku.at). F. Hlawatsch is with the Institute of Telecommunications, TU Wien, Vienna, Austria and with Brno University of Technology, Brno, Czech Republic (e-mail: franz.hlawatsch@tuwien.ac.at).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JIOT.2018.xxxxxxx

I. Introduction

LOCATION AWARENESS plays a critical role for various applications and services in the context of the Internet of Things (IoT), including pedestrian navigation, intelligent transportation, smart home, medical services, and target tracking [1]–[6]. In wireless networks where a global navigation satellite system (GNSS) is unavailable, location information can be obtained by using the device-to-device communication capabilities of the IoT to extract inter-node measurements, such as range measurements and angle-of-arrival measurements [7]–[11]. However, a localization technique with desirable scalability and accuracy for the IoT that can deal with the large number, heterogeneous nature, and limited hardware capabilities of the devices in the network is still lacking.

Time-based cooperative localization is a powerful method for obtaining reliable location information [11]–[19]. Here, nodes in the network transmit wireless signals to other nodes and infer inter-node range information from transmit and receive time measurements. The performance of time-based cooperative localization is impacted by the hardware capabilities of the involved devices. In particular, offsets and skews of the clocks of the IoT devices introduce errors into the range measurements, and thus the localization performance will be degraded if these synchronization issues are not properly addressed [7].

To mitigate the effects of imperfect clocks, the concept of network localization and synchronization (NLS) has been developed recently [19]–[31]. In NLS, the clock parameters are integrated with the location parameters into a state vector, and the values of all parameters are estimated jointly. This joint estimation is performed using either a centralized [20]–[25] or a distributed [19], [28]–[31] mode of processing. However, for IoT applications only a distributed processing mode is suitable, as collecting and processing the measurements of all devices at a central processing unit (fusion center) is prohibitive due to the large number of devices in the network [4], [5]. An attractive approach to distributed estimation in a Bayesian setting is based on the belief propagation (BP) algorithm [32], [33]. This algorithm is able to efficiently compute probability density functions (PDFs) of the individual state components conditioned on all the measurements in a distributed manner, and in a way such that the relevant location and clock information becomes locally available at each device. A major advantage of the BP algorithm, which is important in the IoT context, is its scalability with respect to the network size [12], [18].

Existing distributed NLS algorithms are limited in different

aspects. Most of the existing work considers networks with static nodes [28]–[30]. However, in many IoT scenarios the nodes are mobile, and thus their location estimates need to be updated frequently for good localization accuracy [34], [35]. Moreover, some algorithms are based on simplified clock models that do not consider time-variation of the clock parameters [28]–[30]. These simplified models often are not suitable for the low-cost clocks used in IoT devices, and thus the performance of the resulting algorithms can be degraded significantly. A more practical dynamic system model was considered in [19]; however, the method presented in [19] may be too complex for IoT applications involving low-cost devices.

In this paper, we study NLS in the context of the IoT. We adopt a realistic model with mobile nodes and time-varying clock offsets and skews, which significantly extends the model of [19]. Differently from most existing work, the proposed method can operate with arbitrary time-evolution models and clock models, and is thus suitable for heterogeneous devices. To account for the computational and communication limitations of IoT devices, we use the low-complexity sigma point belief propagation (SPBP) technique [36] to develop a distributed BP algorithm that tracks second-order descriptions of the posterior distributions of the location and clock parameters in a computationally efficient manner.

The main contributions of the paper are as follows:

- We construct a factor graph based on the state-evolution and measurement models, and design a sigma point BP algorithm for tracking the time-varying location and clock parameters of heterogeneous devices.
- We evaluate the performance of the designed algorithm through simulation. Our results show that accurate synchronization and localization are achieved with a reduced computational complexity compared to the state-of-the-art method presented in [19].

The remaining sections are organized as follows. Section II introduces the system model. Section III presents a BP technique for NLS. Section IV reviews the sigma point method [37], [38]. Section V presents the proposed SPBP algorithm for NLS. Section VI reports numerical results, and Section VII 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, and a random vector and its realization by $\mathbf{x}$ and $\boldsymbol{x}$, respectively. Furthermore, $\|\boldsymbol{x}\|$ and $\boldsymbol{x}^{\mathrm{T}}$ denote the Euclidean norm and the transpose of vector $\boldsymbol{x}$, respectively; $\propto$ indicates equality up to a normalization factor; $f(\boldsymbol{x})$ denotes the probability density function (PDF) of random vector $\mathbf{x}$ (this is a short notation for $f_{\mathbf{x}}(\boldsymbol{x})$); $f(\boldsymbol{x}|\boldsymbol{y})$ denotes the conditional PDF of random vector $\mathbf{x}$ conditioned on random vector $\mathbf{y}$; $\mathbf{x} \sim f(\boldsymbol{x})$ indicates that $\mathbf{x}$ is distributed according to PDF $f(\boldsymbol{x})$; $\mathbb{E}\{\mathbf{x}\}$ denotes the expectation of $\mathbf{x}$; $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ represents the Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$; $|\mathcal{S}|$ denotes the cardinality of set $\mathcal{S}$; $\boldsymbol{I}_n$ denotes the $n \times n$ identity matrix; $\mathrm{diag}\{\cdot\}$ denotes the diagonal matrix whose diagonal elements are the listed scalars, and $\mathrm{bdiag}\{\cdot\}$ denotes the block diagonal matrix whose diagonal blocks are the listed matrices.

II. System Model

In this section, we describe the device network, the dynamic model of device locations and clocks, and the measurement model. In our measurement model, time measurements are obtained by means of an asymmetric time-stamped communication technique, which is described in [21] and also used in [19]. However, differently from [19], we consider more general state-evolution models that may be different across the devices and nonlinear in both the location and clock parameters. In addition, our measurement model accommodates measurements from additional sensors for positioning, navigation and timing (PNT) such as GNSS receivers.

A. Network, Clock Model, State-Space Model

We consider a decentralized, time-varying network of I potentially mobile and asynchronous devices $i \in \mathcal{I} = \{1, \ldots, I\}$. We assume that the reference (true) time t is slotted in intervals of length T, indexed by $n \in \{0, 1, \ldots\}$; the nth time step interval is given by $nT \leq t < (n+1)T$. However, the devices are not able to autonomously determine the beginning of a new time interval. The following parameters and attributes are assumed to be static during time step n, but are allowed to change across n. The edge set $\mathcal{C}^{(n)}$ of the network is defined by the fact that two devices $i, j \in \mathcal{I}$, $i \neq j$ are able to communicate at time step n if $(i,j) \in \mathcal{C}^{(n)}$ (and, by symmetry, $(j,i) \in \mathcal{C}^{(n)}$). The *neighborhood* $\mathcal{T}_i^{(n)} \subseteq \mathcal{I} \backslash \{i\}$ of device $i \in \mathcal{I}$ is the set of all devices $j \in \mathcal{I} \backslash \{i\}$ that communicate with device i at time step n, i.e., $\mathcal{T}_i^{(n)} \triangleq \big\{ j \in \mathcal{I} \setminus \{i\} \big| (i,j) \in \mathcal{C}^{(n)} \big\}$. A probabilistic location and clock reference is introduced to the network through statistical prior knowledge of the devices about their location and clock parameters. In particular, a *spatial master device* has accurate prior information of its own location, and a *temporal master device* has accurate prior information of the time.

Each device $i \in \mathcal{I}$ has an internal clock modeled as

$$\mathsf{c}_i\big(t; \boldsymbol{\omega}_i^{(n)}\big) = \boldsymbol{\alpha}_i^{(n)} t + \boldsymbol{\beta}_i^{(n)}, \tag{1}$$

where $\boldsymbol{\alpha}_i^{(n)}$ and $\boldsymbol{\beta}_i^{(n)}$ are, respectively, the *clock skew* and *clock phase* at time step n. These two parameters constitute the *clock state* $\boldsymbol{\omega}_i^{(n)} \triangleq \big[\boldsymbol{\alpha}_i^{(n)} \;\; \boldsymbol{\beta}_i^{(n)}\big]^{\mathrm{T}}$. Furthermore, each device i has a *location-related state* $\mathbf{x}_i^{(n)} \triangleq \big[\mathsf{x}_{1,i}^{(n)} \; \mathsf{x}_{2,i}^{(n)} \; \dot{\mathsf{x}}_{1,i}^{(n)} \; \dot{\mathsf{x}}_{2,i}^{(n)}\big]^{\mathrm{T}}$ that consists of the location $\mathbf{p}_i^{(n)} \triangleq \big[\mathsf{x}_{1,i}^{(n)} \; \mathsf{x}_{2,i}^{(n)}\big]^{\mathrm{T}}$ and the velocity $\big[\dot{\mathsf{x}}_{1,i}^{(n)} \; \dot{\mathsf{x}}_{2,i}^{(n)}\big]^{\mathrm{T}}$. The *state* of device i at time step n is thus given by $\boldsymbol{\theta}_i^{(n)} \triangleq \big[\mathbf{x}_i^{(n)\mathrm{T}} \; \boldsymbol{\omega}_i^{(n)\mathrm{T}}\big]^{\mathrm{T}}$.

We assume a possibly nonlinear state-evolution model with additive noise for both the location-related state and the clock state. The temporal evolution of the location-related state of device i is modeled as

$$\mathbf{x}_i^{(n)} = \boldsymbol{g}_{1,i}\big(\mathbf{x}_i^{(n-1)}\big) + \mathbf{u}_{1,i}^{(n)}, \qquad n = 1, 2, \ldots, \tag{2}$$

where $\boldsymbol{g}_{1,i} : \mathbb{R}^4 \to \mathbb{R}^4$ is an arbitrary function modeling the motion of IoT device i and $\mathbf{u}_{1,i}^{(n)} \in \mathbb{R}^4$ is independent across n

and i and zero-mean Gaussian, i.e., $\mathbf{u}_{1,i}^{(n)} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\boldsymbol{u}_{1,i}})$. The temporal evolution of the clock state of device i is given by

$$\boldsymbol{\omega}_i^{(n)} = \boldsymbol{g}_{2,i}\big(\boldsymbol{\omega}_i^{(n-1)}\big) + \mathbf{u}_{2,i}^{(n)}, \qquad n=1,2,\dots, \tag{3}$$

where $\boldsymbol{g}_{2,i}\colon \mathbb{R}^2 \to \mathbb{R}^2$ is an arbitrary function modeling the clock evolution and $\mathbf{u}_{2,i}^{(n)} \in \mathbb{R}^2$ is independent across n and i and zero-mean Gaussian, i.e., $\mathbf{u}_{2,i}^{(n)} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\boldsymbol{u}_{2,i}})$. We can combine (2) and (3) into the overall state-evolution model

$$\boldsymbol{\theta}_i^{(n)} = \boldsymbol{g}_i\big(\boldsymbol{\theta}_i^{(n-1)}\big) + \mathbf{u}_i^{(n)}, \qquad n=1,2,\dots, \tag{4}$$

where $\boldsymbol{g}_i\big(\boldsymbol{\theta}_i^{(n-1)}\big) \triangleq \big[\boldsymbol{g}_{1,i}^{\mathrm{T}}\big(\mathbf{x}_i^{(n-1)}\big) \;\; \boldsymbol{g}_{2,i}^{\mathrm{T}}\big(\boldsymbol{\omega}_i^{(n-1)}\big)\big]^{\mathrm{T}}$ and $\mathbf{u}_i^{(n)} \triangleq \big[\mathbf{u}_{1,i}^{(n)\mathrm{T}}\, \mathbf{u}_{2,i}^{(n)\mathrm{T}}\big]^{\mathrm{T}}$. We note that $\mathbf{u}_i^{(n)} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\boldsymbol{u}_i})$ with $\boldsymbol{\Sigma}_{\boldsymbol{u}_i} \triangleq \mathrm{bdiag}\{\boldsymbol{\Sigma}_{\boldsymbol{u}_{1,i}}, \boldsymbol{\Sigma}_{\boldsymbol{u}_{2,i}}\}$. The joint state-evolution model in (4) defines the joint state-transition PDF $f\big(\boldsymbol{\theta}_i^{(n)}\big|\boldsymbol{\theta}_i^{(n-1)}\big)$. We assume that device i knows the initial prior PDF of its state at time $n=0$, $f\big(\boldsymbol{\theta}_i^{(0)}\big)$.

B. Measurement Model and Local Likelihood Functions

Following [19], in each time step interval $[nT, (n+1)T]$, there is a *measurement phase* in which the devices acquire the measurements needed for tracking their states $\boldsymbol{\theta}_i^{(n)}$. Each measurement phase consists of an *initialization phase* in which the temporal masters communicate to the other devices the beginning of the measurement phase, and a *packet exchange phase* during which time measurements are obtained by means of the asymmetric time-stamped communication technique described in [21]. We note that the asymmetric time-stamped communication technique can be realized with inexpensive and low-power ultra-wideband impulse radios [39], [40], which have recently been integrated into a single chip [41] and are expected to play a key role in location-aware IoT devices. In addition to these "pairwise" measurements (involving pairs of devices), certain devices may also perform measurements individually, as we will discuss in Section II-B4.

We assume that the duration of the measurement phase is short compared to the time step duration T, so that the clock parameters remain approximately constant during the measurement phase.

1) Initialization Phase: Because of their imprecise clocks, the devices are typically unable to accurately determine when a new time step and, hence, a packet exchange starts. This information is provided by the temporal master devices via the following simple communication protocol:

- After time T has passed since the last measurement phase, the temporal masters initialize a new time step by broadcasting a "start packet exchange" message to their neighbors in the network.
- When a device receives the first "start packet exchange" message from one of its neighbors, it starts a packet exchange with that neighbor and itself broadcasts a "start packet exchange" message to its neighbors.

2) Packet Exchange Phase: The packet exchange between a communicating device pair $(i,j) \in \mathcal{C}^{(n)}$ proceeds as follows [21]. Device i transmits $K_{ij} \geq 1$ packets to device j, and device j transmits $K_{ji} \geq 1$ packets to device i. At time n, the

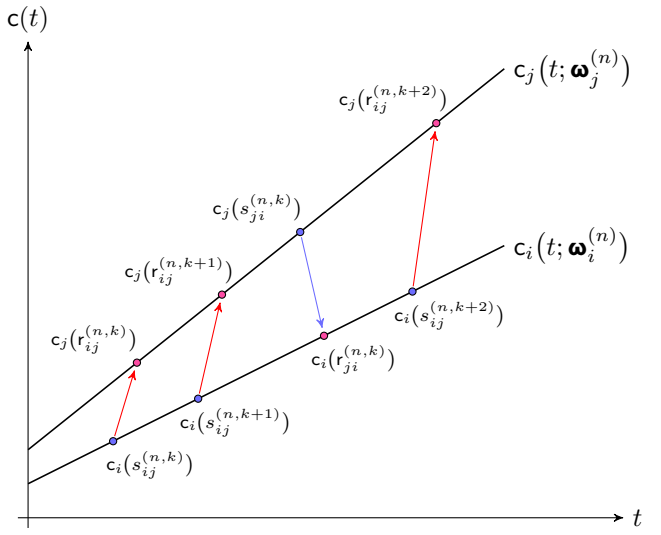


Fig. 1. Local clock functions $\mathsf{c}_i\big(t;\boldsymbol{\omega}_i^{(n)}\big)$ and $\mathsf{c}_j\big(t;\boldsymbol{\omega}_j^{(n)}\big)$, local time measurements (time stamps), and corresponding packet transmissions for devices i and j.

kth "$i \to j$" packet (where $k \in \{1,\dots,K_{ij}\}$) leaves device i at time $s_{ij}^{(n,k)}$ and arrives at device j at measured time $\mathsf{r}_{ij}^{(n,k)} > s_{ij}^{(n,k)}$. The times $s_{ij}^{(n,k)}$ and $\mathsf{r}_{ij}^{(n,k)}$ are related as

$$\mathsf{r}_{ij}^{(n,k)} = s_{ij}^{(n,k)} + \delta_{ij}^{(n,k)}, \quad \text{with } \delta_{ij}^{(n,k)} = \frac{\mathsf{d}_{ij}^{(n)}}{c} + \mathsf{v}_{ij}^{(n,k)},$$

where $\delta_{ij}^{(n,k)}$ is the transmission delay expressed in true time, $\mathsf{d}_{ij}^{(n)} \triangleq \|\mathbf{p}_i^{(n)} - \mathbf{p}_j^{(n)}\|$ is the distance between devices i and j, c is the speed of light, and $\mathsf{v}_{ij}^{(n,k)} \sim f(v_{ij}^{(n,k)}) = \mathcal{N}(0, \sigma_{v_{ij}}^2)$ with $\sigma_{v_{ji}}^2 = \sigma_{v_{ij}}^2$ is measurement noise that is modeled as Gaussian, independent across i, j, k, and n, and independent of all states $\boldsymbol{\theta}_{i'}^{(n')}$ for all i' and n'.

The times $s_{ij}^{(n,k)}$ and $\mathsf{r}_{ij}^{(n,k)}$ are recorded in local time (cf. (1)) at device i and j, respectively, which results in the time stamps

$$\mathsf{c}_i\big(s_{ij}^{(n,k)}\big) = \alpha_i^{(n)} s_{ij}^{(n,k)} + \beta_i^{(n)} \tag{5}$$

and

$$\begin{aligned}\mathsf{c}_j\big(\mathsf{r}_{ij}^{(n,k)}\big) &= \alpha_j^{(n)} \mathsf{r}_{ij}^{(n,k)} + \beta_j^{(n)} \\ &= \alpha_j^{(n)} \left(s_{ij}^{(n,k)} + \frac{\mathsf{d}_{ij}^{(n)}}{c} + \mathsf{v}_{ij}^{(n,k)} \right) + \beta_j^{(n)} \\ &= \psi_{i\to j}^{(n,k)}\big(\boldsymbol{\theta}_i^{(n)}, \boldsymbol{\theta}_j^{(n)}\big) + \mathsf{v}_{ij}^{(n,k)} \alpha_j^{(n)}, \end{aligned} \tag{6}$$

with

$$\psi_{i\to j}^{(n,k)}\big(\boldsymbol{\theta}_i^{(n)}, \boldsymbol{\theta}_j^{(n)}\big) \triangleq \frac{\mathsf{c}_i\big(s_{ij}^{(n,k)}\big) - \beta_i^{(n)}}{\alpha_i^{(n)}} \alpha_j^{(n)} + \beta_j^{(n)} + \frac{\mathsf{d}_{ij}^{(n)}}{c} \alpha_j^{(n)}. \tag{7}$$

A similar discussion applies to the transmission of the kth packet from device j to device i (where $k \in \{1,\dots,K_{ji}\}$); expressions of the resulting time stamps, $\mathsf{c}_j\big(s_{ji}^{(n,k)}\big)$ and $\mathsf{c}_i\big(\mathsf{r}_{ji}^{(n,k)}\big)$, are obtained by exchanging i and j in (5)–(7). Fig. 1 visualizes the clock functions $\mathsf{c}_i\big(t;\boldsymbol{\omega}_i^{(n)}\big)$ and $\mathsf{c}_j\big(t;\boldsymbol{\omega}_j^{(n)}\big)$ and the time stamps. Communication is performed such that all

the time stamps are available at both devices i and j. The aggregated measurement of devices i and j is thus given by $\mathbf{y}_{ij}^{(n)} \triangleq \big[\mathbf{y}_{i\to j}^{(n)\mathrm{T}}\ \mathbf{y}_{j\to i}^{(n)\mathrm{T}}\big]^{\mathrm{T}}$, with $\mathbf{y}_{i\to j}^{(n)} \triangleq \big[\mathsf{c}_j\big(\mathsf{r}_{ij}^{(n,1)}\big)\ \mathsf{c}_j\big(\mathsf{r}_{ij}^{(n,2)}\big) \cdots \mathsf{c}_j\big(\mathsf{r}_{ij}^{(n,K_{ij})}\big)\big]^{\mathrm{T}}$ and $\mathbf{y}_{j\to i}^{(n)} \triangleq \big[\mathsf{c}_i\big(\mathsf{r}_{ji}^{(n,1)}\big)\ \mathsf{c}_i\big(\mathsf{r}_{ji}^{(n,2)}\big) \cdots \mathsf{c}_i\big(\mathsf{r}_{ji}^{(n,K_{ji})}\big)\big]^{\mathrm{T}}$. Similarly, the aggregated measurement noise of devices i and j is given by $\mathbf{v}_{ij}^{(n)} \triangleq \big[\mathbf{v}_{i\to j}^{(n)\mathrm{T}}\ \mathbf{v}_{j\to i}^{(n)\mathrm{T}}\big]^{\mathrm{T}}$, with $\mathbf{v}_{i\to j}^{(n)} \triangleq \big[\mathsf{v}_{ij}^{(n,1)}\ \mathsf{v}_{ij}^{(n,2)} \cdots \mathsf{v}_{ij}^{(n,K_{ij})}\big]^{\mathrm{T}}$ and $\mathbf{v}_{j\to i}^{(n)} \triangleq \big[\mathsf{v}_{ji}^{(n,1)}\ \mathsf{v}_{ji}^{(n,2)} \cdots \mathsf{v}_{ji}^{(n,K_{ji})}\big]^{\mathrm{T}}$. Let us denote by $\boldsymbol{\lambda}_i^{(n)} \triangleq \big[\mathsf{x}_{1,i}^{(n)}\ \mathsf{x}_{2,i}^{(n)}\ \boldsymbol{\alpha}_i^{(n)}\ \boldsymbol{\beta}_i^{(n)}\big]^{\mathrm{T}}$ the "measurement-affecting" part of the state $\boldsymbol{\theta}_i^{(n)}$, i.e., the part of $\boldsymbol{\theta}_i^{(n)}$ that is involved in the measurement model according to (5)–(7). For future reference, we formally introduce the aggregated measurement model

$$\mathbf{y}_{ij}^{(n)} = \boldsymbol{\phi}\big(\boldsymbol{\lambda}_i^{(n)}, \boldsymbol{\lambda}_j^{(n)}, \mathbf{v}_{ij}^{(n)}\big), \tag{8}$$

which can be directly obtained from (6), (7), and the definition of $\mathbf{y}_{ij}^{(n)}$. Note that $\mathbf{v}_{ij}^{(n)}$ is a zero-mean Gaussian random vector with covariance matrix $\boldsymbol{C}_{\boldsymbol{v}ij} = \sigma_{v_{ij}}^2 \boldsymbol{I}_{K_{ij}+K_{ji}}$.

3) Pairwise Likelihood Functions: Because of (6) and the statistical properties of the $\mathsf{v}_{ij}^{(n,k)}$, the "pairwise" local likelihood function of devices i and j is obtained as

$$\begin{aligned}
&f\big(\boldsymbol{y}_{ij}^{(n)}\big|\boldsymbol{\theta}_i^{(n)}, \boldsymbol{\theta}_j^{(n)}\big)\\
&= f\big(\boldsymbol{y}_{i\to j}^{(n)}\big|\boldsymbol{\theta}_i^{(n)}, \boldsymbol{\theta}_j^{(n)}\big)\, f\big(\boldsymbol{y}_{j\to i}^{(n)}\big|\boldsymbol{\theta}_i^{(n)}, \boldsymbol{\theta}_j^{(n)}\big)\\
&= \left(\prod_{k=1}^{K_{ij}} f\big(c_j\big(r_{ij}^{(n,k)}\big)\big|\boldsymbol{\theta}_i^{(n)}, \boldsymbol{\theta}_j^{(n)}\big)\right) \prod_{k'=1}^{K_{ji}} f\big(c_i\big(r_{ji}^{(n,k')}\big)\big|\boldsymbol{\theta}_i^{(n)}, \boldsymbol{\theta}_j^{(n)}\big)\\
&= M_{ij}^{(n)} \exp\left(-\frac{\|\boldsymbol{y}_{i\to j}^{(n)} - \boldsymbol{\psi}_{i\to j}^{(n)}\|^2}{2\alpha_j^{(n)2}\sigma_{v_{ij}}^2} - \frac{\|\boldsymbol{y}_{j\to i}^{(n)} - \boldsymbol{\psi}_{j\to i}^{(n)}\|^2}{2\alpha_i^{(n)2}\sigma_{v_{ji}}^2}\right).
\end{aligned}$$

Here, $f\big(c_j\big(r_{ij}^{(n,k)}\big)\big|\boldsymbol{\theta}_i^{(n)}, \boldsymbol{\theta}_j^{(n)}\big)$ and $f\big(c_i\big(r_{ji}^{(n,k)}\big)\big|\boldsymbol{\theta}_i^{(n)}, \boldsymbol{\theta}_j^{(n)}\big)$ are the single-packet likelihood functions corresponding to, respectively, (6) and the analogous expression with i, j interchanged, and furthermore

$$\begin{aligned}
M_{ij}^{(n)} &\triangleq \big(2\pi\alpha_j^{(n)2}\sigma_{v_{ij}}^2\big)^{-K_{ij}/2} \big(2\pi\alpha_i^{(n)2}\sigma_{v_{ji}}^2\big)^{-K_{ji}/2},\\
\boldsymbol{\psi}_{i\to j}^{(n)} &\triangleq \big[\psi_{i\to j}^{(n,1)}(\boldsymbol{\theta}_i^{(n)}, \boldsymbol{\theta}_j^{(n)}) \cdots \psi_{i\to j}^{(n,K_{ij})}(\boldsymbol{\theta}_i^{(n)}, \boldsymbol{\theta}_j^{(n)})\big]^{\mathrm{T}},\\
\boldsymbol{\psi}_{j\to i}^{(n)} &\triangleq \big[\psi_{j\to i}^{(n,1)}(\boldsymbol{\theta}_j^{(n)}, \boldsymbol{\theta}_i^{(n)}) \cdots \psi_{j\to i}^{(n,K_{ji})}(\boldsymbol{\theta}_j^{(n)}, \boldsymbol{\theta}_i^{(n)})\big]^{\mathrm{T}}.
\end{aligned}$$

We note that $f\big(\boldsymbol{y}_{ij}^{(n)}\big|\boldsymbol{\theta}_i^{(n)}, \boldsymbol{\theta}_j^{(n)}\big) = f\big(\boldsymbol{y}_{ij}^{(n)}\big|\boldsymbol{\lambda}_i^{(n)}, \boldsymbol{\lambda}_j^{(n)}\big)$.

4) Self-Measurements and Individual Likelihood Functions: In addition to the above-described pairwise measurements involving other devices, some devices may also incorporate measurements from additional sensors for PNT such as GNSS receivers. These measurements will be briefly termed "self-measurements" in what follows. Let $\mathcal{S}^{(n)} \subseteq \mathcal{I}$ denote the set of devices obtaining self-measurements, and let $\mathbf{y}_i^{(n)} \in \mathbb{R}^{m_i}$ denote the self-measurement of device $i \in \mathcal{S}^{(n)}$ at time n. For example, if $\mathbf{y}_i^{(n)}$ comprises measurements of the global time and of the two-dimensional (2-D) location of device i, then $m_i = 3$. We model $\mathbf{y}_i^{(n)}$ according to

$$\mathbf{y}_i^{(n)} = \boldsymbol{h}_i\big(\boldsymbol{\theta}_i^{(n)}\big) + \mathbf{v}_i^{(n)}, \tag{9}$$

where $\boldsymbol{h}_i(\cdot)$ is a vector function with output dimension m_i and $\mathbf{v}_i^{(n)} \in \mathbb{R}^{m_i}$ is zero-mean Gaussian measurement noise with covariance matrix $\boldsymbol{C}_{\boldsymbol{v}_i}$, i.e., $\mathbf{v}_i^{(n)} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{C}_{\boldsymbol{v}_i})$, which is independent across i and n and independent of all states $\boldsymbol{\theta}_{i'}^{(n')}$ and of all measurement noise vectors $\mathbf{v}_{i'j}^{(n')}$ for all i', j, and n'. From (9), the "individual" likelihood function of device $i \in \mathcal{S}^{(n)}$ is obtained as

$$\begin{aligned}
f\big(\boldsymbol{y}_i^{(n)}\big|\boldsymbol{\theta}_i^{(n)}\big) = M_i \exp\Big(&-\frac{1}{2}\big(\boldsymbol{y}_i^{(n)} - \boldsymbol{h}_i\big(\boldsymbol{\theta}_i^{(n)}\big)\big)^{\mathrm{T}}\\
&\times \boldsymbol{C}_{\boldsymbol{v}_i}^{-1}\big(\boldsymbol{y}_i^{(n)} - \boldsymbol{h}_i\big(\boldsymbol{\theta}_i^{(n)}\big)\big)\Big),
\end{aligned}$$

with $M_i \triangleq \big((2\pi)^{m_i} \det(\boldsymbol{C}_{\boldsymbol{v}_i})\big)^{-1/2}$.

III. Synchronization and Localization Using BP

At each time step n, each device i estimates its current state $\boldsymbol{\theta}_i^{(n)}$ from all past and present observed measurements, $\boldsymbol{y}^{(1:n)} \triangleq \big[\boldsymbol{y}_{ij}^{(n')}\big]_{(i,j)\in\mathcal{C}^{(n')},\, n'\in\{1,\ldots,n\}}$. In particular, the minimum mean-square error (MMSE) estimate [42] of $\boldsymbol{\theta}_i^{(n)}$ is given by the mean of the *marginal posterior PDF* $f\big(\boldsymbol{\theta}_i^{(n)}\big|\boldsymbol{y}^{(1:n)}\big)$, i.e.,

$$\hat{\boldsymbol{\theta}}_{i,\mathrm{MMSE}}^{(n)} \triangleq \int \boldsymbol{\theta}_i^{(n)} f\big(\boldsymbol{\theta}_i^{(n)}\big|\boldsymbol{y}^{(1:n)}\big)\, \mathrm{d}\boldsymbol{\theta}_i^{(n)}, \quad i \in \mathcal{I}. \tag{10}$$

Here, $f\big(\boldsymbol{\theta}_i^{(n)}\big|\boldsymbol{y}^{(1:n)}\big)$ can be obtained from the *joint posterior PDF* $f(\boldsymbol{\theta}^{(0:n)}|\boldsymbol{y}^{(1:n)})$, where $\boldsymbol{\theta}^{(0:n)} \triangleq \big[\boldsymbol{\theta}_i^{(n')}\big]_{i\in\mathcal{I},\, n'\in\{0,1,\ldots,n\}}$. This is done via the marginalization operation

$$f\big(\boldsymbol{\theta}_i^{(n)}\big|\boldsymbol{y}^{(1:n)}\big) = \int f(\boldsymbol{\theta}^{(0:n)}|\boldsymbol{y}^{(1:n)})\, \mathrm{d}{\sim}\boldsymbol{\theta}_i^{(n)}, \tag{11}$$

where ${\sim}\boldsymbol{\theta}_i^{(n)}$ denotes the vector stacking all $\boldsymbol{\theta}_{i'}^{(n')}$ except $\boldsymbol{\theta}_i^{(n)}$. However, a straightforward implementation of this marginalization operation is computationally infeasible due to the typically high dimensionality of ${\sim}\boldsymbol{\theta}_i^{(n)}$. This problem can be addressed by using the BP algorithm, which takes advantage of the temporal and spatial independence structure of the joint posterior PDF $f(\boldsymbol{\theta}^{(0:n)}|\boldsymbol{y}^{(1:n)})$—to be explored next—and avoids explicit integration.

A. Factorization and Factor Graph

Under commonly used assumptions [12], [28], the joint posterior PDF can be factored as

$$\begin{aligned}
&f(\boldsymbol{\theta}^{(0:n)}|\boldsymbol{y}^{(1:n)})\\
&\propto \left(\prod_{i'\in\mathcal{I}} f\big(\boldsymbol{\theta}_{i'}^{(0)}\big)\right) \prod_{n'=1}^{n} \left(\prod_{j'\in\mathcal{S}^{(n')}} f\big(\boldsymbol{y}_{j'}^{(n')}\big|\boldsymbol{\theta}_{j'}^{(n')}\big)\right)\\
&\quad\times \prod_{i\in\mathcal{I}} f\big(\boldsymbol{\theta}_i^{(n')}\big|\boldsymbol{\theta}_i^{(n'-1)}\big) \prod_{\substack{j\in\mathcal{T}_i^{(n')}\\ j>i}} f\big(\boldsymbol{y}_{ij}^{(n')}\big|\boldsymbol{\theta}_j^{(n')}, \boldsymbol{\theta}_i^{(n')}\big).
\end{aligned} \tag{12}$$

A graphical representation of this factorization is provided by the factor graph shown in Fig. 2. Each factor function in (12) is represented by a square factor node, and each variable in (12) by a circular variable node. A variable node is connected to a factor node by an edge if the respective variable is an argument of the respective factor function. The factor graph is the basis for applying the BP algorithm, as explained next.

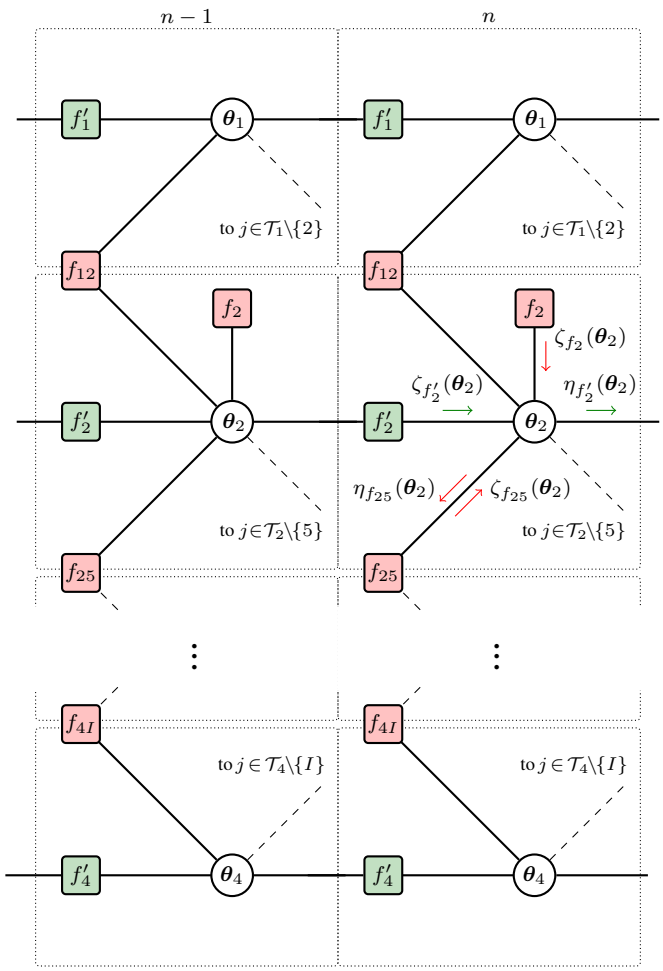


Fig. 2. Factor graph representing the factorization (12) for a network of devices $i \in \{1,2,\ldots,I\}$, where $(1,2)$, $(2,5)$, and $(4,I)$ belong to both $\mathcal{C}^{(n)}$ and $\mathcal{C}^{(n-1)}$ and 2 belongs to both $\mathcal{S}^{(n-1)}$ and $\mathcal{S}^{(n)}$. The time steps $n-1$ and n are shown; time indices are omitted for simplicity. Each dotted box corresponds to a device i; calculations within the box are performed locally by that device. Connections between dotted boxes imply communication between devices. The following short notations are used: $f_i' \triangleq f\big(\boldsymbol{\theta}_i^{(n)}\big|\boldsymbol{\theta}_i^{(n-1)}\big)$, $f_{ij} \triangleq f\big(\boldsymbol{y}_{ij}^{(n)}\big|\boldsymbol{\theta}_i^{(n)},\boldsymbol{\theta}_j^{(n)}\big)$, and $f_i \triangleq f\big(\boldsymbol{y}_i^{(n)}\big|\boldsymbol{\theta}_i^{(n)}\big)$.

B. Belief Propagation

Because a straightforward computation of the marginalization integral in (11) is infeasible, we resort to approximate inference by means of BP [32], [33] performed on the factor graph in Fig. 2. BP provides approximations of the marginal posterior PDFs, $b\big(\boldsymbol{\theta}_i^{(n)}\big) \approx f\big(\boldsymbol{\theta}_i^{(n)}\big|\boldsymbol{y}^{(1:n)}\big)$, which are referred to as *beliefs*. These beliefs are calculated in a distributed and sequential (i.e., time-recursive) manner, such that at each time n, $b\big(\boldsymbol{\theta}_i^{(n)}\big)$ becomes available at device i. An approximation of the MMSE estimate $\hat{\boldsymbol{\theta}}_{i,\mathrm{MMSE}}^{(n)}$ in (10) is then obtained by calculating (10) with $f\big(\boldsymbol{\theta}_i^{(n)}\big|\boldsymbol{y}^{(1:n)}\big)$ replaced by $b\big(\boldsymbol{\theta}_i^{(n)}\big)$.

In our implementation of BP, we apply the following three-step message passing schedule: (i) In the *temporal cooperation step*, at each device i, a message for state prediction is passed forward in time (from time step $n-1$ to time step n). (ii) In the *self-update step*, at each time step n and at each device $i \in \mathcal{S}^{(n)}$, the message for state prediction is multiplied by a "self-update message." (iii) In the *spatial cooperation step*, at each time step n, messages are updated across the devices i. After these three message passing steps, the beliefs are computed. In accordance with the time-recursive nature of our algorithm, no messages are passed backward in time. Hereafter, we drop the time index n for simplicity, and we use the following short notation (cf. Fig. 2): $f_i' \triangleq f\big(\boldsymbol{\theta}_i^{(n)}\big|\boldsymbol{\theta}_i^{(n-1)}\big)$, $f_{ij} \triangleq f\big(\boldsymbol{y}_{ij}^{(n)}\big|\boldsymbol{\theta}_i^{(n)},\boldsymbol{\theta}_j^{(n)}\big)$, and $f_i \triangleq f\big(\boldsymbol{y}_i^{(n)}\big|\boldsymbol{\theta}_i^{(n)}\big)$. Furthermore, $\boldsymbol{\theta}_i^-$ denotes the variable (state) $\boldsymbol{\theta}_i$ at the preceding time $n-1$, and $\eta_{f_i'}(\boldsymbol{\theta}_i^-)$ the message passed from variable node $\boldsymbol{\theta}_i^-$ at time step $n-1$ to factor node f_i' at time n. Finally, $\zeta_{f_i'}(\boldsymbol{\theta}_i)$ denotes the message passed from factor node f_i' to variable node $\boldsymbol{\theta}_i$; $\zeta_{f_i}(\boldsymbol{\theta}_i)$ the message passed from factor node f_i to variable node $\boldsymbol{\theta}_i$; $\zeta_{f_{ij}}(\boldsymbol{\theta}_i)$ the message passed from factor node f_{ij} to variable node $\boldsymbol{\theta}_i$; $\eta_{f_{ij}}(\boldsymbol{\theta}_i)$ the message passed from variable node $\boldsymbol{\theta}_i$ to factor node f_{ij}; and $\eta_{f_i'}(\boldsymbol{\theta}_i)$ the message passed from variable node $\boldsymbol{\theta}_i$ to factor node f_i' (all at time n).

The following message calculation and message passing rules are obtained by applying BP [32], [33] to the factor graph in Fig. 2, using the three-step message passing schedule described above and performing in the spatial cooperation step only a single message passing iteration. In the temporal cooperation step, at each device i, the message $\eta_{f_i'}(\boldsymbol{\theta}_i^-)$ is set equal to the belief calculated at the previous time $n-1$, i.e.,

$$\eta_{f_i'}(\boldsymbol{\theta}_i^-) = b(\boldsymbol{\theta}_i^-)\,,$$

and the message $\zeta_{f_i'}(\boldsymbol{\theta}_i)$ is calculated as

$$\zeta_{f_i'}(\boldsymbol{\theta}_i) = \int f(\boldsymbol{\theta}_i|\boldsymbol{\theta}_i^-)\,\eta_{f_i'}(\boldsymbol{\theta}_i^-)\,\mathrm{d}\boldsymbol{\theta}_i^- = \int f(\boldsymbol{\theta}_i|\boldsymbol{\theta}_i^-)\,b(\boldsymbol{\theta}_i^-)\,\mathrm{d}\boldsymbol{\theta}_i^-. \tag{13}$$

Indeed, due to the specific message schedule used (i.e., no messages are passed backward in time), the message from variable node $\boldsymbol{\theta}_i^-$ at time $n-1$ to factor node f_i' at time n, $\eta_{f_i'}(\boldsymbol{\theta}_i^-)$, is directly given by the preceding belief $b(\boldsymbol{\theta}_i^-)$. Thus, $b(\boldsymbol{\theta}_i^-)$ is used in the temporal update step (13). Furthermore, for devices $i \in \mathcal{S}$, the messages $\zeta_{f_i}(\boldsymbol{\theta}_i)$ are set equal to $f(\boldsymbol{y}_i|\boldsymbol{\theta}_i)$. It will be convenient to formally define messages $\zeta_{f_i}(\boldsymbol{\theta}_i)$ also for devices $i \notin \mathcal{S}$, namely, as the trivial message $\zeta_{f_i}(\boldsymbol{\theta}_i) = 1$.

In the self-update step, at each device i, the message $\eta_{f_{ij}}(\boldsymbol{\theta}_i)$ is calculated as

$$\eta_{f_{ij}}(\boldsymbol{\theta}_i) = \zeta_{f_i'}(\boldsymbol{\theta}_i)\,\zeta_{f_i}(\boldsymbol{\theta}_i)\,, \tag{14}$$

and then broadcast to the neighbors $j \in \mathcal{T}_i$. (Note that for devices $i \notin \mathcal{S}$, (14) reduces to $\eta_{f_{ij}}(\boldsymbol{\theta}_i) = \zeta_{f_i'}(\boldsymbol{\theta}_i)$.) In the spatial cooperation step, at each device i, the message $\zeta_{f_{ij}}(\boldsymbol{\theta}_i)$ is calculated as

$$\zeta_{f_{ij}}(\boldsymbol{\theta}_i) = \int f(\boldsymbol{y}_{ij}|\boldsymbol{\theta}_i,\boldsymbol{\theta}_j)\,\eta_{f_{ji}}(\boldsymbol{\theta}_j)\,\mathrm{d}\boldsymbol{\theta}_j\,, \tag{15}$$

for each neighbor $j \in \mathcal{T}_i$. Finally, the belief at device i is obtained as

$$\begin{aligned} b(\boldsymbol{\theta}_i) &\propto \zeta_{f_i'}(\boldsymbol{\theta}_i)\,\zeta_{f_i}(\boldsymbol{\theta}_i) \prod_{j\in\mathcal{T}_i} \zeta_{f_{ij}}(\boldsymbol{\theta}_i) \\ &= \eta_{f_{ij}}(\boldsymbol{\theta}_i) \prod_{j\in\mathcal{T}_i} \zeta_{f_{ij}}(\boldsymbol{\theta}_i)\,. \end{aligned} \tag{16}$$

We note that the messages $\zeta_{f_i'}(\boldsymbol{\theta}_i)$ in (13) and the belief

$b(\boldsymbol{\theta}_i)$ in (16) are PDFs, i.e., they integrate to one. Furthermore, the message $\zeta_{f_{ij}}(\boldsymbol{\theta}_i)$ in (15) can also be expressed as

$$\zeta_{f_{ij}}(\boldsymbol{\theta}_i) = \int f(\boldsymbol{y}_{ij}|\boldsymbol{\lambda}_i, \boldsymbol{\lambda}_j)\, \tilde{\eta}_{f_{ji}}(\boldsymbol{\lambda}_j)\, \mathrm{d}\boldsymbol{\lambda}_j\,, \tag{17}$$

where

$$\tilde{\eta}_{f_{ji}}(\boldsymbol{\lambda}_j) \triangleq \iint \eta_{f_{ji}}(\boldsymbol{\theta}_j)\, \mathrm{d}\dot{x}_{1,j}\, \mathrm{d}\dot{x}_{2,j}\,. \tag{18}$$

An exact evaluation of the expressions (13)–(17) is computationally infeasible in general. For a static NLS scenario, an approximate particle implementation was presented in [28]. A similar implementation could be developed for the present dynamic scenario but would be too complex for most low-cost IoT devices. In Section V, we will propose an approximate implementation based on the sigma point BP technique recently introduced in [36]. This implementation has a significantly lower complexity than a particle implementation, and is thus better suited to IoT applications.

IV. Review of Sigma Point Based Estimation

We first review the basic principles of sigma points (SPs) [37], [38]. Consider the nonlinear measurement model

$$\mathsf{y} = \boldsymbol{\gamma}(\mathsf{x}, \mathsf{n})\,,$$

where $\mathsf{x} \in \mathbb{R}^{L_{\boldsymbol{x}}}$ is a random vector with known mean $\boldsymbol{\mu}_{\boldsymbol{x}} \triangleq \mathbb{E}\{\mathsf{x}\}$ and covariance matrix $\boldsymbol{C}_{\boldsymbol{x}} \triangleq \mathbb{E}\{(\mathsf{x}-\boldsymbol{\mu}_{\boldsymbol{x}})(\mathsf{x}-\boldsymbol{\mu}_{\boldsymbol{x}})^{\mathrm{T}}\}$; the noise $\mathsf{n} \in \mathbb{R}^{L_{\boldsymbol{n}}}$ is statistically independent of x and has mean $\boldsymbol{\mu}_{\boldsymbol{n}} = \mathbf{0}$ and a known covariance matrix $\boldsymbol{C}_{\boldsymbol{n}}$; and $\boldsymbol{\gamma}(\cdot,\cdot)$ is a known, generally nonlinear function. We wish to estimate x from an observed realization of y. In the Bayesian framework, we need to calculate the posterior PDF

$$f(\boldsymbol{x}|\boldsymbol{y}) \propto f(\boldsymbol{y}|\boldsymbol{x}) f(\boldsymbol{x})\,, \tag{19}$$

where $f(\boldsymbol{y}|\boldsymbol{x})$ is the likelihood function and $f(\boldsymbol{x})$ is the prior PDF. A direct exact calculation of (19) would be possible if x and n were Gaussian random vectors and $\boldsymbol{\gamma}(\boldsymbol{x}, \boldsymbol{n}) = \boldsymbol{\Gamma}\boldsymbol{x} + \boldsymbol{n}$ with some known matrix $\boldsymbol{\Gamma}$. In that case, $f(\boldsymbol{x}|\boldsymbol{y})$ is Gaussian, i.e., $f(\boldsymbol{x}|\boldsymbol{y}) = \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}}, \boldsymbol{C}_{\boldsymbol{x}|\boldsymbol{y}})$, and the posterior mean $\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}}$ and posterior covariance matrix $\boldsymbol{C}_{\boldsymbol{x}|\boldsymbol{y}}$ are given by [43]

$$\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}} = \boldsymbol{\mu}_{\boldsymbol{x}} + \boldsymbol{K}(\boldsymbol{y} - \boldsymbol{\mu}_{\boldsymbol{y}})\,, \tag{20}$$

$$\boldsymbol{C}_{\boldsymbol{x}|\boldsymbol{y}} = \boldsymbol{C}_{\boldsymbol{x}} - \boldsymbol{K}\boldsymbol{C}_{\boldsymbol{y}}\boldsymbol{K}^{\mathrm{T}}, \tag{21}$$

where $\boldsymbol{\mu}_{\boldsymbol{y}} = \boldsymbol{\Gamma}\boldsymbol{\mu}_{\boldsymbol{x}}$, $\boldsymbol{C}_{\boldsymbol{y}} = \boldsymbol{\Gamma}\boldsymbol{C}_{\boldsymbol{x}}\boldsymbol{\Gamma}^{\mathrm{T}} + \boldsymbol{C}_{\boldsymbol{n}}$, and

$$\boldsymbol{K} = \boldsymbol{C}_{\boldsymbol{x}\boldsymbol{y}}\boldsymbol{C}_{\boldsymbol{y}}^{-1}, \tag{22}$$

with $\boldsymbol{C}_{\boldsymbol{x}\boldsymbol{y}} = \boldsymbol{C}_{\boldsymbol{x}}\boldsymbol{\Gamma}^{\mathrm{T}}$. We note that $\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}}$ is also the MMSE estimate of x, and $\boldsymbol{C}_{\boldsymbol{x}|\boldsymbol{y}}$ characterizes the uncertainty of the estimate.

In the general case of a nonlinear function $\boldsymbol{\gamma}(\cdot,\cdot)$ and non-Gaussian x and n considered here, the expressions (20) and (21) cannot be applied directly, but they can be used to develop an approximation based on so-called SPs [37], [38]. More specifically, let us define the "noise-augmented" state vector $\mathsf{x}_* \triangleq [\mathsf{x}^{\mathrm{T}}\, \mathsf{n}^{\mathrm{T}}]^{\mathrm{T}}$ with dimension $L \triangleq L_{\boldsymbol{x}} + L_{\boldsymbol{n}}$. Following [37], [38], we deterministically choose SPs $\{\boldsymbol{x}_*^{(l)}\}_{l=0}^{2L}$ with $\boldsymbol{x}_*^{(l)} \triangleq \left[\boldsymbol{x}^{(l)\mathrm{T}}\, \boldsymbol{n}^{(l)\mathrm{T}}\right]^{\mathrm{T}}$ and corresponding weights $\{w^{(l)}\}_{l=0}^{2L}$ such that the resulting weighted sample mean and weighted sample covariance matrix are exactly equal to $\boldsymbol{\mu}_{\boldsymbol{x}_*} = [\boldsymbol{\mu}_{\boldsymbol{x}}^{\mathrm{T}}\, \boldsymbol{\mu}_{\boldsymbol{n}}^{\mathrm{T}}]^{\mathrm{T}}$ and $\boldsymbol{C}_{\boldsymbol{x}_*} = \mathrm{bdiag}\{\boldsymbol{C}_{\boldsymbol{x}}, \boldsymbol{C}_{\boldsymbol{n}}\}$, respectively, i.e.,

$$\tilde{\boldsymbol{\mu}}_{\boldsymbol{x}_*} \triangleq \sum_{l=0}^{2L} w^{(l)} \boldsymbol{x}_*^{(l)} = \boldsymbol{\mu}_{\boldsymbol{x}_*}\,,$$

$$\tilde{\boldsymbol{C}}_{\boldsymbol{x}_*} \triangleq \sum_{l=0}^{2L} w^{(l)} \left(\boldsymbol{x}_*^{(l)} - \tilde{\boldsymbol{\mu}}_{\boldsymbol{x}_*}\right)\left(\boldsymbol{x}_*^{(l)} - \tilde{\boldsymbol{\mu}}_{\boldsymbol{x}_*}\right)^{\mathrm{T}} = \boldsymbol{C}_{\boldsymbol{x}_*}\,.$$

According to [37], [38], these SPs and weights can be calculated as

$$\boldsymbol{x}_*^{(l)} = \begin{cases} \boldsymbol{\mu}_{\boldsymbol{x}_*}\,, & l = 0 \\ \boldsymbol{\mu}_{\boldsymbol{x}_*} + \sqrt{L+\kappa}\left(\boldsymbol{C}_{\boldsymbol{x}_*}^{1/2}\right)_l\,, & l = 1, \ldots, L \\ \boldsymbol{\mu}_{\boldsymbol{x}_*} - \sqrt{L+\kappa}\left(\boldsymbol{C}_{\boldsymbol{x}_*}^{1/2}\right)_{l-L}\,, & l = L+1, \ldots, 2L, \end{cases} \tag{23}$$

$$w^{(l)} = \begin{cases} \dfrac{\kappa}{L+\kappa}\,, & l = 0 \\ \dfrac{1}{2(L+\kappa)}\,, & l = 1, \ldots, 2L, \end{cases} \tag{24}$$

where $\left(\boldsymbol{C}_{\boldsymbol{x}_*}^{1/2}\right)_l$ denotes the lth column of the matrix square root of $\boldsymbol{C}_{\boldsymbol{x}_*}$ and κ is a tuning parameter that determines the spread of the SPs around $\boldsymbol{\mu}_{\boldsymbol{x}_*}$.

Let $\boldsymbol{y}^{(l)} \triangleq \boldsymbol{\gamma}(\boldsymbol{x}^{(l)}, \boldsymbol{n}^{(l)})$, $l = 0, 1, \ldots, 2L$ denote the SPs $\boldsymbol{x}_*^{(l)} = [\boldsymbol{x}^{(l)\mathrm{T}}\, \boldsymbol{n}^{(l)\mathrm{T}}]^{\mathrm{T}}$ transformed by $\boldsymbol{\gamma}(\cdot,\cdot)$. An approximate representation of the second-order statistics of the two random vectors x and $\mathsf{y} = \boldsymbol{\gamma}(\mathsf{x}, \mathsf{n})$ is then provided by the set $\{(\boldsymbol{x}^{(l)}, \boldsymbol{y}^{(l)}, w^{(l)})\}_{l=0}^{2L}$. In particular, $\boldsymbol{\mu}_{\boldsymbol{y}}$, $\boldsymbol{C}_{\boldsymbol{y}}$, and $\boldsymbol{C}_{\boldsymbol{x}\boldsymbol{y}} \triangleq \mathbb{E}\{(\mathsf{x}-\boldsymbol{\mu}_{\boldsymbol{x}})(\mathsf{y}-\boldsymbol{\mu}_{\boldsymbol{y}})^{\mathrm{T}}\}$ are approximated by, respectively,

$$\tilde{\boldsymbol{\mu}}_{\boldsymbol{y}} \triangleq \sum_{l=0}^{2L} w^{(l)} \boldsymbol{y}^{(l)}, \tag{25}$$

$$\tilde{\boldsymbol{C}}_{\boldsymbol{y}} \triangleq \sum_{l=0}^{2L} w^{(l)} (\boldsymbol{y}^{(l)} - \tilde{\boldsymbol{\mu}}_{\boldsymbol{y}})(\boldsymbol{y}^{(l)} - \tilde{\boldsymbol{\mu}}_{\boldsymbol{y}})^{\mathrm{T}}, \tag{26}$$

$$\tilde{\boldsymbol{C}}_{\boldsymbol{x}\boldsymbol{y}} \triangleq \sum_{l=0}^{2L} w^{(l)} (\boldsymbol{x}^{(l)} - \tilde{\boldsymbol{\mu}}_{\boldsymbol{x}})(\boldsymbol{y}^{(l)} - \tilde{\boldsymbol{\mu}}_{\boldsymbol{y}})^{\mathrm{T}}. \tag{27}$$

These approximations have been shown in [37], [38] to be at least as accurate as those resulting from a linearization of $\boldsymbol{\gamma}(\cdot,\cdot)$. Furthermore, the number of SPs, $2L+1$, grows only linearly with the dimension of x_* (which is L). Thus, the number of SPs is typically much smaller than the number of random samples required for a particle-based approximation of the second-order statistics of x and y.

To obtain an approximate second-order description of the posterior PDF $f(\boldsymbol{x}|\boldsymbol{y})$ in (19), we now use (20)–(22) with $\boldsymbol{\mu}_{\boldsymbol{y}}$, $\boldsymbol{C}_{\boldsymbol{y}}$, and $\boldsymbol{C}_{\boldsymbol{x}\boldsymbol{y}}$ replaced by the SP approximations $\tilde{\boldsymbol{\mu}}_{\boldsymbol{y}}$, $\tilde{\boldsymbol{C}}_{\boldsymbol{y}}$, and $\tilde{\boldsymbol{C}}_{\boldsymbol{x}\boldsymbol{y}}$ in (25)–(27). This yields the following approximations of $\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}}$ and $\boldsymbol{C}_{\boldsymbol{x}|\boldsymbol{y}}$:

$$\tilde{\boldsymbol{\mu}}_{\boldsymbol{x}|\boldsymbol{y}} = \boldsymbol{\mu}_{\boldsymbol{x}} + \tilde{\boldsymbol{K}}(\boldsymbol{y} - \tilde{\boldsymbol{\mu}}_{\boldsymbol{y}})\,, \tag{28}$$

$$\tilde{\boldsymbol{C}}_{\boldsymbol{x}|\boldsymbol{y}} = \boldsymbol{C}_{\boldsymbol{x}} - \tilde{\boldsymbol{K}}\tilde{\boldsymbol{C}}_{\boldsymbol{y}}\tilde{\boldsymbol{K}}^{\mathrm{T}}, \tag{29}$$

with $\tilde{\boldsymbol{K}} = \tilde{\boldsymbol{C}}_{\boldsymbol{x}\boldsymbol{y}}\tilde{\boldsymbol{C}}_{\boldsymbol{y}}^{-1}$. To summarize, we have obtained the

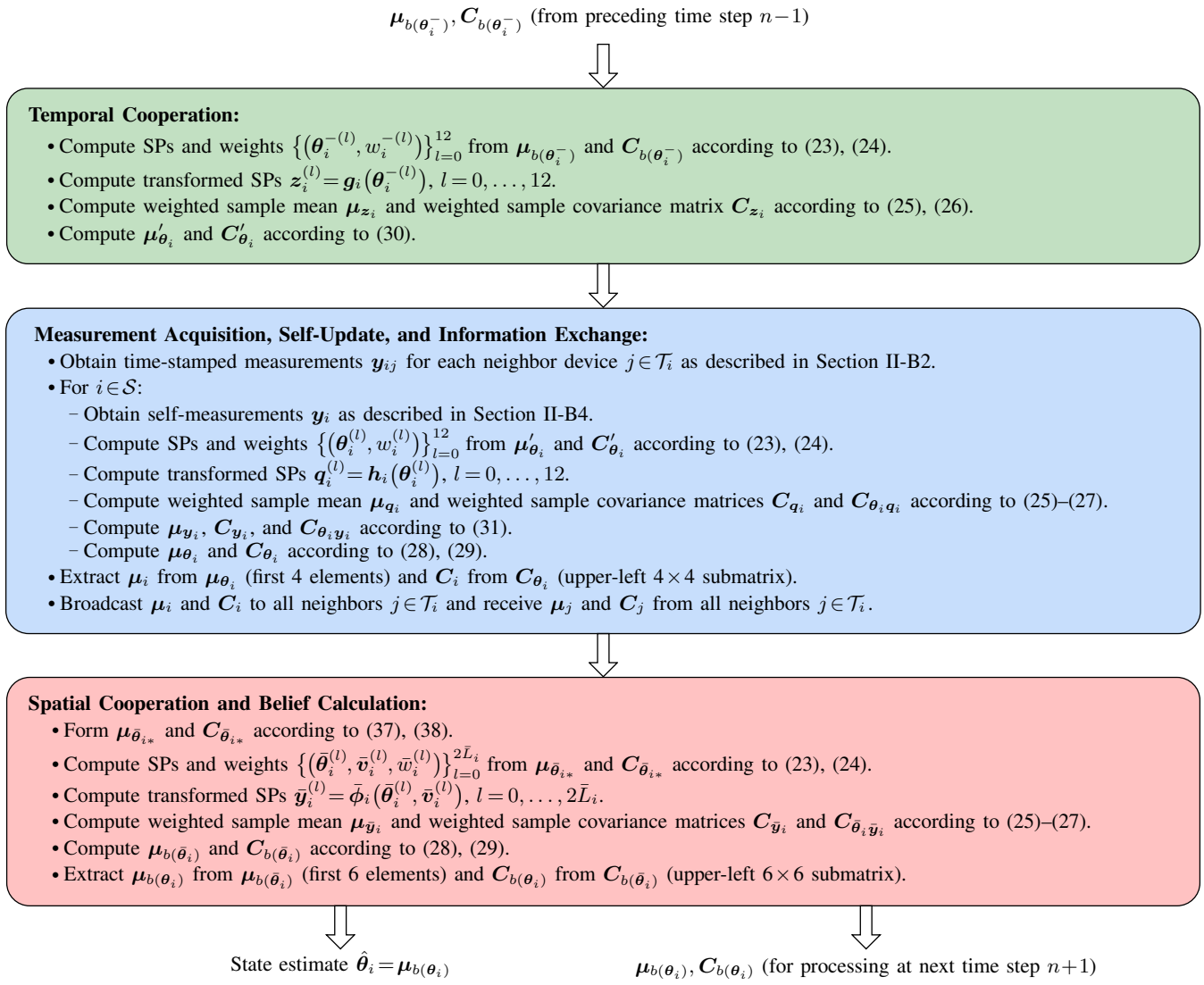


Fig. 3. Flowchart of the proposed distributed SPBP algorithm (operations performed at device i).

following approximate SP implementation of (19), resulting in an approximate second-order description of the posterior PDF $f(\boldsymbol{x}|\boldsymbol{y})$:

Step 1: SPs and weights $\{(\boldsymbol{x}^{(l)}, \boldsymbol{n}^{(l)}, w^{(l)})\}_{l=0}^{2L}$ are calculated from $\boldsymbol{\mu}_{\boldsymbol{x}_*}$ and $\boldsymbol{C}_{\boldsymbol{x}_*}$ according to (23) and (24).

Step 2: The transformed SPs $\boldsymbol{y}^{(l)} = \boldsymbol{\gamma}(\boldsymbol{x}^{(l)}, \boldsymbol{n}^{(l)})$, $l = 0, 1, \ldots, 2L$ are calculated.

Step 3: Approximations of the posterior mean $\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}}$ and posterior covariance $\boldsymbol{C}_{\boldsymbol{x}|\boldsymbol{y}}$ are calculated from $\{(\boldsymbol{x}^{(l)}, \boldsymbol{y}^{(l)}, w^{(l)})\}_{l=0}^{2L}$ by evaluating $\tilde{\boldsymbol{\mu}}_{\boldsymbol{y}}$, $\tilde{\boldsymbol{C}}_{\boldsymbol{y}}$, and $\tilde{\boldsymbol{C}}_{\boldsymbol{x}\boldsymbol{y}}$ in (25)–(27) and, in turn, $\tilde{\boldsymbol{\mu}}_{\boldsymbol{x}|\boldsymbol{y}}$ in (28) and $\tilde{\boldsymbol{C}}_{\boldsymbol{x}|\boldsymbol{y}}$ in (29).

Note that the SP method obtains only a second-order description of the posterior PDF $f(\boldsymbol{x}|\boldsymbol{y})$. This implies a limitation of the class of posterior PDFs that can be represented with good accuracy.

V. The SPBP Algorithm for NLS

We will now present the proposed SPBP algorithm for NLS. This algorithm is a low-complexity "second-order" implementation of the BP method described in Section III-B, and is based on an extension of the SPBP technique introduced in [36] to nonadditive measurement noise. A flowchart of the presented method is shown in Fig. 3.

A. Temporal Cooperation Step

In the temporal cooperation step of the BP method described in Section III-B, each device i calculates the message $\zeta_{f'_i}(\boldsymbol{\theta}_i)$ from the preceding belief $b(\boldsymbol{\theta}_i^-)$ according to (13). Using the SP method, we can obtain an approximate second-order description of $\zeta_{f'_i}(\boldsymbol{\theta}_i)$ (i.e., mean $\boldsymbol{\mu}'_{\boldsymbol{\theta}_i}$ and covariance matrix $\boldsymbol{C}'_{\boldsymbol{\theta}_i}$) from an approximate second-order description of $b(\boldsymbol{\theta}_i^-)$ (i.e., mean $\boldsymbol{\mu}_{b(\boldsymbol{\theta}_i^-)}$ and covariance matrix $\boldsymbol{C}_{b(\boldsymbol{\theta}_i^-)}$, which were calculated at time step $n-1$). This is done by the following

steps, the first three of which are analogous[1] to Steps 1–3 in Section IV:

Step 1A: SPs and weights $\{(\boldsymbol{\theta}_i^{-(l)}, w_i^{-(l)})\}_{l=0}^{12}$ corresponding to $b(\boldsymbol{\theta}_i^-)$ are calculated from $\boldsymbol{\mu}_{b(\boldsymbol{\theta}_i^-)}$ and $\boldsymbol{C}_{b(\boldsymbol{\theta}_i^-)}$ according to (23) and (24) with obvious substitutions. (Note that the dimension of $\boldsymbol{\theta}_i$ is $L = 6$, and thus the number of sigma points is $2L + 1 = 13$.)

Step 2A: The transformed SPs $\boldsymbol{z}_i^{(l)} = \boldsymbol{g}_i(\boldsymbol{\theta}_i^{-(l)})$, $l = 0, 1, \ldots, 12$ are calculated. Here, $\boldsymbol{g}_i(\cdot)$ is the state-evolution function from (4).

Step 3A: From $\{(\boldsymbol{z}_i^{(l)}, w_i^{-(l)})\}_{l=0}^{12}$, the weighted sample mean $\boldsymbol{\mu}_{\boldsymbol{z}_i}$ and weighted sample covariance matrix $\boldsymbol{C}_{\boldsymbol{z}_i}$ are calculated as in (25) and (26), respectively.

Step 4A: The approximate mean $\boldsymbol{\mu}'_{\boldsymbol{\theta}_i}$ and covariance matrix $\boldsymbol{C}'_{\boldsymbol{\theta}_i}$ corresponding to $\zeta_{f'_i}(\boldsymbol{\theta}_i)$ are obtained as

$$\boldsymbol{\mu}'_{\boldsymbol{\theta}_i} = \boldsymbol{\mu}_{\boldsymbol{z}_i}, \quad \boldsymbol{C}'_{\boldsymbol{\theta}_i} = \boldsymbol{C}_{\boldsymbol{z}_i} + \boldsymbol{\Sigma}_{\boldsymbol{u}_i}. \tag{30}$$

(We note that these expressions are based on the relation $\boldsymbol{\theta}_i = \mathbf{z}_i + \mathbf{u}_i$, which is obtained from the state-evolution model (4) using the relation $\mathbf{z}_i = \boldsymbol{g}_i(\boldsymbol{\theta}_i^-)$.)

In the special case where the state-evolution function is linear, i.e., where (4) simplifies to $\boldsymbol{\theta}_i = \boldsymbol{G}_i\boldsymbol{\theta}_i^- + \mathbf{u}_i$ with some matrix $\boldsymbol{G}_i$, Steps 1A–4A are replaced by the direct relations

$$\boldsymbol{\mu}'_{\boldsymbol{\theta}_i} = \boldsymbol{G}_i\boldsymbol{\mu}_{b(\boldsymbol{\theta}_i^-)}, \quad \boldsymbol{C}'_{\boldsymbol{\theta}_i} = \boldsymbol{G}_i\boldsymbol{C}_{b(\boldsymbol{\theta}_i^-)}\boldsymbol{G}_i^{\mathrm{T}} + \boldsymbol{\Sigma}_{\boldsymbol{u}_i}.$$

No SPs are needed in that case.

B. Self-Update Step

In the self-update step of the BP method described in Section III-B, each device i calculates the message $\eta_{f_{ij}}(\boldsymbol{\theta}_i)$ that is broadcast to all neighbors $j \in \mathcal{T}_i$ from the messages $\zeta_{f'_i}(\boldsymbol{\theta}_i)$ and $\zeta_{f_i}(\boldsymbol{\theta}_i)$ via (14). If $i \notin \mathcal{S}$, then because (14) reduces to $\eta_{f_{ij}}(\boldsymbol{\theta}_i) = \zeta_{f'_i}(\boldsymbol{\theta}_i)$, an approximate second-order description $(\boldsymbol{\mu}_{\boldsymbol{\theta}_i}, \boldsymbol{C}_{\boldsymbol{\theta}_i})$ of $\eta_{f_{ij}}(\boldsymbol{\theta}_i)$ is directly given by the approximate second-order description $(\boldsymbol{\mu}'_{\boldsymbol{\theta}_i}, \boldsymbol{C}'_{\boldsymbol{\theta}_i})$ of $\zeta_{f'_i}(\boldsymbol{\theta}_i)$ calculated in (30), i.e., $\boldsymbol{\mu}_{\boldsymbol{\theta}_i} = \boldsymbol{\mu}'_{\boldsymbol{\theta}_i}$ and $\boldsymbol{C}_{\boldsymbol{\theta}_i} = \boldsymbol{C}'_{\boldsymbol{\theta}_i}$.

If $i \in \mathcal{S}$, then $(\boldsymbol{\mu}_{\boldsymbol{\theta}_i}, \boldsymbol{C}_{\boldsymbol{\theta}_i})$ can be obtained from $(\boldsymbol{\mu}'_{\boldsymbol{\theta}_i}, \boldsymbol{C}'_{\boldsymbol{\theta}_i})$ using the SP method. First, we recall that $\zeta_{f_i}(\boldsymbol{\theta}_i) = f(\boldsymbol{y}_i|\boldsymbol{\theta}_i)$, and thus the calculation of $\eta_{f_{ij}}(\boldsymbol{\theta}_i) = \zeta_{f'_i}(\boldsymbol{\theta}_i)\zeta_{f_i}(\boldsymbol{\theta}_i) = \zeta_{f'_i}(\boldsymbol{\theta}_i)\, f(\boldsymbol{y}_i|\boldsymbol{\theta}_i)$ (see (14)) is equivalent to the Bayesian update step in (19). Thus, $\boldsymbol{\mu}_{\boldsymbol{\theta}_i}$ and $\boldsymbol{C}_{\boldsymbol{\theta}_i}$ are now calculated by the following steps, the first three of which are analogous to Steps 1–3 in Section IV:

Step 1B: SPs and weights $\{(\boldsymbol{\theta}_i^{(l)}, w_i^{(l)})\}_{l=0}^{12}$ corresponding to $\zeta_{f'_i}(\boldsymbol{\theta}_i)$ are calculated from $\boldsymbol{\mu}'_{\boldsymbol{\theta}_i}$ and $\boldsymbol{C}'_{\boldsymbol{\theta}_i}$ according to (23) and (24).

Step 2B: The transformed SPs $\boldsymbol{q}_i^{(l)} = \boldsymbol{h}_i(\boldsymbol{\theta}_i^{(l)})$, $l = 0, \ldots, 12$ are calculated. Here, $\boldsymbol{h}_i(\cdot)$ is the measurement function from (9).

Step 3B: From $\{(\boldsymbol{\theta}_i^{(l)}, \boldsymbol{q}_i^{(l)}, w_i^{(l)})\}_{l=0}^{12}$, the weighted sample mean $\boldsymbol{\mu}_{\boldsymbol{q}_i}$, weighted sample covariance matrix $\boldsymbol{C}_{\boldsymbol{q}_i}$, and weighted sample cross-covariance matrix $\boldsymbol{C}_{\boldsymbol{\theta}_i\boldsymbol{q}_i}$ are calculated as in (25), (26), and (27), respectively.

Step 4B: An approximate mean $\boldsymbol{\mu}_{\boldsymbol{y}_i}$, covariance matrix $\boldsymbol{C}_{\boldsymbol{y}_i}$, and cross-covariance matrix $\boldsymbol{C}_{\boldsymbol{\theta}_i\boldsymbol{y}_i}$ are obtained as

$$\boldsymbol{\mu}_{\boldsymbol{y}_i} = \boldsymbol{\mu}_{\boldsymbol{q}_i}, \quad \boldsymbol{C}_{\boldsymbol{y}_i} = \boldsymbol{C}_{\boldsymbol{q}_i} + \boldsymbol{C}_{\boldsymbol{v}_i}, \quad \boldsymbol{C}_{\boldsymbol{\theta}_i\boldsymbol{y}_i} = \boldsymbol{C}_{\boldsymbol{\theta}_i\boldsymbol{q}_i}. \tag{31}$$

(These expressions are based on the relation $\mathbf{y}_i = \mathbf{q}_i + \mathbf{v}_i$, which is obtained from the self-measurement model (9) using the relation $\mathbf{q}_i = \boldsymbol{h}_i(\boldsymbol{\theta}_i)$.)

Step 5B: The desired approximate second-order description $(\boldsymbol{\mu}_{\boldsymbol{\theta}_i}, \boldsymbol{C}_{\boldsymbol{\theta}_i})$ of $\eta_{f_{ij}}(\boldsymbol{\theta}_i)$ is calculated as in (20)–(22) with $\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}}$, $\boldsymbol{C}_{\boldsymbol{x}|\boldsymbol{y}}$, $\boldsymbol{\mu}_{\boldsymbol{x}}$, $\boldsymbol{C}_{\boldsymbol{x}}$, $\boldsymbol{\mu}_{\boldsymbol{y}}$, $\boldsymbol{C}_{\boldsymbol{y}}$, and $\boldsymbol{C}_{\boldsymbol{x}\boldsymbol{y}}$ replaced by $\boldsymbol{\mu}_{\boldsymbol{\theta}_i}$, $\boldsymbol{C}_{\boldsymbol{\theta}_i}$, $\boldsymbol{\mu}'_{\boldsymbol{\theta}_i}$, $\boldsymbol{C}'_{\boldsymbol{\theta}_i}$, $\boldsymbol{\mu}_{\boldsymbol{y}_i}$, $\boldsymbol{C}_{\boldsymbol{y}_i}$, and $\boldsymbol{C}_{\boldsymbol{\theta}_i\boldsymbol{y}_i}$, respectively.

If the self-measurement model is linear, i.e., if (9) takes the form $\mathbf{y}_i = \boldsymbol{H}_i\boldsymbol{\theta}_i + \mathbf{v}_i$ with some matrix $\boldsymbol{H}_i$, then no SPs are needed because $(\boldsymbol{\mu}_{\boldsymbol{\theta}_i}, \boldsymbol{C}_{\boldsymbol{\theta}_i})$ can be directly obtained as

$$\boldsymbol{\mu}_{\boldsymbol{\theta}_i} = \boldsymbol{H}_i\boldsymbol{\mu}'_{\boldsymbol{\theta}_i}, \quad \boldsymbol{C}_{\boldsymbol{\theta}_i} = \boldsymbol{H}_i\boldsymbol{C}'_{\boldsymbol{\theta}_i}\boldsymbol{H}_i^{\mathrm{T}} + \boldsymbol{C}_{\boldsymbol{v}_i}.$$

C. Dimension-Augmented Reformulation of the Belief

The remaining steps in the BP method of Section III-B are the spatial cooperation operation (15) and the belief calculation (16). These steps are performed jointly. For a presentation of this joint calculation, we first introduce a "dimension-augmented" reformulation of the belief $b(\boldsymbol{\theta}_i)$ in (16). Let the neighbor nodes $j \in \mathcal{T}_i$ of node i be denoted by $j_1, j_2, \ldots, j_{|\mathcal{T}_i|}$, i.e., $\mathcal{T}_i = \{j_1, j_2, \ldots, j_{|\mathcal{T}_i|}\}$, and define the stacked (dimension-augmented) vectors

$$\bar{\boldsymbol{\theta}}_i \triangleq \big[\boldsymbol{\theta}_i^{\mathrm{T}}\ \boldsymbol{\lambda}_{j_1}^{\mathrm{T}}\ \boldsymbol{\lambda}_{j_2}^{\mathrm{T}} \cdots \boldsymbol{\lambda}_{j_{|\mathcal{T}_i|}}^{\mathrm{T}}\big]^{\mathrm{T}},$$
$$\bar{\mathbf{y}}_i \triangleq \big[\mathbf{y}_{ij_1}^{\mathrm{T}}\ \mathbf{y}_{ij_2}^{\mathrm{T}} \cdots \mathbf{y}_{ij_{|\mathcal{T}_i|}}^{\mathrm{T}}\big]^{\mathrm{T}}.$$

Note that $\bar{\boldsymbol{\theta}}_i$ comprises the state of device i, i.e., $\boldsymbol{\theta}_i$, and the measurement-affecting parts of the states of all the neighbors of device i, i.e., $\boldsymbol{\lambda}_j$ for $j \in \mathcal{T}_i$. The dimension of $\bar{\boldsymbol{\theta}}_i$ is $\bar{L}_i = 6 + 4|\mathcal{T}_i|$. Furthermore, $\bar{\mathbf{y}}_i$ comprises all the measurements involving $\boldsymbol{\theta}_i$. Now, using (17) in (16), one obtains

$$b(\boldsymbol{\theta}_i) \propto \eta_{f_{ij}}(\boldsymbol{\theta}_i) \prod_{j\in\mathcal{T}_i} \int f(\boldsymbol{y}_{ij}|\boldsymbol{\lambda}_i, \boldsymbol{\lambda}_j)\, \tilde{\eta}_{f_{ji}}(\boldsymbol{\lambda}_j)\, \mathrm{d}\boldsymbol{\lambda}_j$$
$$= \int\cdots\int \eta_{f_{ij}}(\boldsymbol{\theta}_i) \prod_{j\in\mathcal{T}_i} f(\boldsymbol{y}_{ij}|\boldsymbol{\lambda}_i, \boldsymbol{\lambda}_j)\, \tilde{\eta}_{f_{ji}}(\boldsymbol{\lambda}_j)\, \mathrm{d}\boldsymbol{\lambda}_j.$$

This can be rewritten in terms of the dimension-augmented vector $\bar{\boldsymbol{\theta}}_i$ as

$$b(\boldsymbol{\theta}_i) = \int b(\bar{\boldsymbol{\theta}}_i)\, \mathrm{d}\bar{\boldsymbol{\theta}}_{\sim i}, \tag{32}$$

where $b(\bar{\boldsymbol{\theta}}_i)$ is defined, up to a normalization factor, as

$$b(\bar{\boldsymbol{\theta}}_i) \propto f(\bar{\boldsymbol{y}}_i|\bar{\boldsymbol{\theta}}_i)\, f(\bar{\boldsymbol{\theta}}_i), \tag{33}$$

with

$$f(\bar{\boldsymbol{\theta}}_i) \triangleq \eta_{f_{ij}}(\boldsymbol{\theta}_i) \prod_{j\in\mathcal{T}_i} \tilde{\eta}_{f_{ji}}(\boldsymbol{\lambda}_j), \tag{34}$$

$$f(\bar{\boldsymbol{y}}_i|\bar{\boldsymbol{\theta}}_i) \triangleq \prod_{j\in\mathcal{T}_i} f(\boldsymbol{y}_{ij}|\boldsymbol{\lambda}_i, \boldsymbol{\lambda}_j), \tag{35}$$

and $\mathrm{d}\bar{\boldsymbol{\theta}}_{\sim i} \triangleq \prod_{j\in\mathcal{T}_i} \mathrm{d}\boldsymbol{\lambda}_j$.

[1]Note, however, that contrary to Steps 1–3 in Section IV, we do not need to augment the statistics of the state vector, $\boldsymbol{\mu}'_{\boldsymbol{\theta}_i}$ and $\boldsymbol{C}'_{\boldsymbol{\theta}_i}$, by the statistics of the noise vector, $\boldsymbol{\mu}_{\boldsymbol{u}_i} = \mathbf{0}$ and $\boldsymbol{\Sigma}_{\boldsymbol{u}_i}$, since noise in the state-evolution model (4) is modeled as additive Gaussian and can thus be incorporated by adding the noise covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{u}_i}$ to the transformed covariance matrix (see (4)).

The "dimension-augmented" reformulation of $b(\boldsymbol{\theta}_i)$ in (33) is seen to be formally analogous to (19), and the factors on the right-hand side of (33), $f(\bar{\boldsymbol{y}}_i|\bar{\boldsymbol{\theta}}_i)$ and $f(\bar{\boldsymbol{\theta}}_i)$, can be given the following interpretations. The expression (34) of $f(\bar{\boldsymbol{\theta}}_i)$ has the form of a prior PDF. Furthermore, according to (35), $f(\bar{\boldsymbol{y}}_i|\bar{\boldsymbol{\theta}}_i)$ is the likelihood function related to the "stacked pairwise observation model"

$$\bar{\mathbf{y}}_i = \bar{\phi}_i(\bar{\boldsymbol{\theta}}_i, \bar{\mathbf{v}}_i), \tag{36}$$

where $\bar{\phi}_i(\bar{\boldsymbol{\theta}}_i, \bar{\mathbf{v}}_i) \triangleq \big[\boldsymbol{\phi}(\boldsymbol{\lambda}_i, \boldsymbol{\lambda}_j, \mathbf{v}_{ij})\big]_{j\in\mathcal{T}_i}$ (cf. (8)) and $\bar{\mathbf{v}}_i \triangleq [\mathbf{v}_{ij}]_{j\in\mathcal{T}_i}$. We note that $\bar{\mathbf{v}}_i$ is a zero-mean Gaussian random vector of dimension $L_{\bar{\boldsymbol{v}}_i} = \sum_{j\in\mathcal{T}_i}(K_{ij}+K_{ji})$; its covariance matrix is $\boldsymbol{C}_{\bar{\boldsymbol{v}}_i} = \mathrm{bdiag}\big\{\boldsymbol{C}_{\boldsymbol{v}_{ij_1}}, \boldsymbol{C}_{\boldsymbol{v}_{ij_2}}, \ldots, \boldsymbol{C}_{\boldsymbol{v}_{ij_{|\mathcal{T}_i|}}}\big\}$.

D. Spatial Cooperation Step and Belief Calculation

We will now describe how the dimension-augmented reformulation of the belief $b(\boldsymbol{\theta}_i)$ introduced above can be used for an SP-based joint second-order implementation of spatial cooperation (15) and belief calculation (16). Indeed, because of the formal analogy of (33), i.e., $b(\bar{\boldsymbol{\theta}}_i) \propto f(\bar{\boldsymbol{y}}_i|\bar{\boldsymbol{\theta}}_i) f(\bar{\boldsymbol{\theta}}_i)$, to (19), i.e., $f(\boldsymbol{x}|\boldsymbol{y}) \propto f(\boldsymbol{y}|\boldsymbol{x})f(\boldsymbol{x})$, we can obtain a second-order description of $b(\boldsymbol{\theta}_i)$—i.e., mean $\boldsymbol{\mu}_{b(\boldsymbol{\theta}_i)}$ and covariance matrix $\boldsymbol{C}_{b(\boldsymbol{\theta}_i)}$—by means of SPs in a similar manner as we obtained a second-order description of $f(\boldsymbol{x}|\boldsymbol{y})$ in Section IV. To this end, we define the noise-augmented dimension-augmented state vector $\bar{\boldsymbol{\theta}}_{i*} \triangleq [\bar{\boldsymbol{\theta}}_i^{\mathrm{T}}\ \bar{\mathbf{v}}_i^{\mathrm{T}}]^{\mathrm{T}}$ with PDF (cf. (34))

$$f(\bar{\boldsymbol{\theta}}_{i*}) = f(\bar{\boldsymbol{\theta}}_i) f(\bar{\boldsymbol{v}}_i) = \eta_{f_{ij}}(\boldsymbol{\theta}_i) \left(\prod_{j\in\mathcal{T}_i} \tilde{\eta}_{f_{ji}}(\boldsymbol{\lambda}_j) \right) f(\bar{\boldsymbol{v}}_i).$$

(Note that $f(\bar{\boldsymbol{\theta}}_{i*}) = f(\bar{\boldsymbol{\theta}}_i) f(\bar{\boldsymbol{v}}_i)$ because, according to Section II-B, $\bar{\boldsymbol{\theta}}_i$ and $\bar{\mathbf{v}}_i$ are independent.) The PDF $f(\bar{\boldsymbol{\theta}}_{i*})$ is the basis for performing spatial cooperation and belief calculation jointly. The second-order description of $f(\bar{\boldsymbol{\theta}}_{i*})$ is provided by the stacked mean vector and corresponding covariance matrix defined as

$$\boldsymbol{\mu}_{\bar{\boldsymbol{\theta}}_{i*}} \triangleq \big[\boldsymbol{\mu}_{\boldsymbol{\theta}_i}^{\mathrm{T}}\ \boldsymbol{\mu}_{j_1}^{\mathrm{T}}\ \boldsymbol{\mu}_{j_2}^{\mathrm{T}} \cdots \boldsymbol{\mu}_{j_{|\mathcal{T}_i|}}^{\mathrm{T}}\ \mathbf{0}_{L_{\bar{\boldsymbol{v}}_i}}^{\mathrm{T}}\big]^{\mathrm{T}}, \tag{37}$$

$$\boldsymbol{C}_{\bar{\boldsymbol{\theta}}_{i*}} \triangleq \mathrm{bdiag}\big\{\boldsymbol{C}_{\boldsymbol{\theta}_i}, \boldsymbol{C}_{j_1}, \boldsymbol{C}_{j_2}, \ldots, \boldsymbol{C}_{j_{|\mathcal{T}_i|}}, \boldsymbol{C}_{\bar{\boldsymbol{v}}_i}\big\}. \tag{38}$$

Here, $\boldsymbol{\mu}_{\boldsymbol{\theta}_i}$ and $\boldsymbol{C}_{\boldsymbol{\theta}_i}$ are the mean and covariance matrix of $\eta_{f_{ij}}(\boldsymbol{\theta}_i)$, which were calculated in the self-update step (see Step 5B); furthermore, the 4-dimensional vectors $\boldsymbol{\mu}_j$ and the 4×4 matrices $\boldsymbol{C}_j$ denote the means and covariance matrices of the messages $\tilde{\eta}_{f_{ji}}(\boldsymbol{\lambda}_j)$, $j \in \mathcal{T}_i$. (Note that by (14) and (18), $\tilde{\eta}_{f_{ji}}(\boldsymbol{\lambda}_j) = \iint \zeta_{f'_j}(\boldsymbol{\theta}_j)\zeta_{f_j}(\boldsymbol{\theta}_j)\,\mathrm{d}\dot{x}_{1,j}\,\mathrm{d}\dot{x}_{2,j}$, which shows that $\tilde{\eta}_{f_{ji}}(\boldsymbol{\lambda}_j)$ does not depend on i.) In accordance with the marginalization (18), $\boldsymbol{\mu}_j$ comprises the first 4 elements of the 6-dimensional vector $\boldsymbol{\mu}_{\boldsymbol{\theta}_j}$, and similarly, $\boldsymbol{C}_j$ equals the upper-left 4×4 submatrix of the 6×6 matrix $\boldsymbol{C}_{\boldsymbol{\theta}_j}$. Thus, $\boldsymbol{\mu}_j$ and $\boldsymbol{C}_j$ can be directly extracted from, respectively, $\boldsymbol{\mu}_{\boldsymbol{\theta}_j}$ and $\boldsymbol{C}_{\boldsymbol{\theta}_j}$, which were calculated by device j in the self-update step. After the self-update step, device i broadcasts its own $\boldsymbol{\mu}_i$ and $\boldsymbol{C}_i$ to all neighbors $j\in\mathcal{T}_i$ and receives their $\boldsymbol{\mu}_j$ and $\boldsymbol{C}_j$; thus, $\boldsymbol{\mu}_j$ and $\boldsymbol{C}_j$ for $j\in\mathcal{T}_i$ are available at device i. We note that $\boldsymbol{\mu}_{\bar{\boldsymbol{\theta}}_{i*}}$ and $\boldsymbol{C}_{\bar{\boldsymbol{\theta}}_{i*}}$ have dimension $\bar{L}_i = 6 + 4|\mathcal{T}_i| + L_{\bar{\boldsymbol{v}}_i}$ and $\bar{L}_i \times \bar{L}_i$, respectively.

The following steps are now performed for joint spatial cooperation and belief calculation (note that the first three steps are analogous to Steps 1–3 in Section IV):

Step 1C: SPs and weights $\big\{\big(\bar{\boldsymbol{\theta}}_i^{(l)}, \bar{\boldsymbol{v}}_i^{(l)}, \bar{w}_i^{(l)}\big)\big\}_{l=0}^{2\bar{L}_i}$ corresponding to $f(\bar{\boldsymbol{\theta}}_{i*}) = f(\bar{\boldsymbol{\theta}}_i) f(\bar{\boldsymbol{v}}_i)$ are calculated from $\boldsymbol{\mu}_{\bar{\boldsymbol{\theta}}_{i*}}$ and $\boldsymbol{C}_{\bar{\boldsymbol{\theta}}_{i*}}$ according to (23) and (24).

Step 2C: The transformed SPs $\bar{\boldsymbol{y}}_i^{(l)} = \bar{\boldsymbol{\phi}}_i\big(\bar{\boldsymbol{\theta}}_i^{(l)}, \bar{\boldsymbol{v}}_i^{(l)}\big)$, $l = 0, 1, \ldots, 2\bar{L}_i$ (cf. (36)) are calculated.

Step 3C: From $\big\{\big(\bar{\boldsymbol{\theta}}_i^{(l)}, \bar{\boldsymbol{y}}_i^{(l)}, \bar{w}_i^{(l)}\big)\big\}_{l=0}^{2\bar{L}_i}$, the weighted sample means and covariances $\boldsymbol{\mu}_{\bar{\boldsymbol{y}}_i}$, $\boldsymbol{C}_{\bar{\boldsymbol{y}}_i}$, and $\boldsymbol{C}_{\bar{\boldsymbol{\theta}}_i\bar{\boldsymbol{y}}_i}$ are calculated as in (25)–(27). Subsequently, an approximate second-order description of $b(\bar{\boldsymbol{\theta}}_i)$ in (33), constituted by mean $\boldsymbol{\mu}_{b(\bar{\boldsymbol{\theta}}_i)}$ and covariance matrix $\boldsymbol{C}_{b(\bar{\boldsymbol{\theta}}_i)}$, is calculated as in (20) and (21) with $\boldsymbol{\mu}_{\boldsymbol{x}|\boldsymbol{y}}$, $\boldsymbol{C}_{\boldsymbol{x}|\boldsymbol{y}}$, $\boldsymbol{\mu}_{\boldsymbol{x}}$, $\boldsymbol{C}_{\boldsymbol{x}}$, $\boldsymbol{\mu}_{\boldsymbol{y}}$, $\boldsymbol{C}_{\boldsymbol{y}}$, and $\boldsymbol{C}_{\boldsymbol{xy}}$ replaced by $\boldsymbol{\mu}_{b(\bar{\boldsymbol{\theta}}_i)}$, $\boldsymbol{C}_{b(\bar{\boldsymbol{\theta}}_i)}$, $\boldsymbol{\mu}_{\bar{\boldsymbol{\theta}}_{i*}}$, $\boldsymbol{C}_{\bar{\boldsymbol{\theta}}_{i*}}$, $\boldsymbol{\mu}_{\bar{\boldsymbol{y}}_i}$, $\boldsymbol{C}_{\bar{\boldsymbol{y}}_i}$, and $\boldsymbol{C}_{\bar{\boldsymbol{\theta}}_i\bar{\boldsymbol{y}}_i}$, respectively.

Step 4C: From $\boldsymbol{\mu}_{b(\bar{\boldsymbol{\theta}}_i)}$ and $\boldsymbol{C}_{b(\bar{\boldsymbol{\theta}}_i)}$, the parts $\boldsymbol{\mu}_{b(\boldsymbol{\theta}_i)}$ and $\boldsymbol{C}_{b(\boldsymbol{\theta}_i)}$ related to $\boldsymbol{\theta}_i$ are extracted; note that this corresponds to the marginalization (32). More specifically, $\boldsymbol{\mu}_{b(\boldsymbol{\theta}_i)}$ is given by the first 6 elements of $\boldsymbol{\mu}_{b(\bar{\boldsymbol{\theta}}_i)}$, and $\boldsymbol{C}_{b(\boldsymbol{\theta}_i)}$ is given by the upper-left 6×6 submatrix of $\boldsymbol{C}_{b(\bar{\boldsymbol{\theta}}_i)}$ (cf. the stacked structure of $\boldsymbol{\mu}_{\bar{\boldsymbol{\theta}}_{i*}}$ in (37) and the block structure of $\boldsymbol{C}_{\bar{\boldsymbol{\theta}}_{i*}}$ in (38)). The mean $\boldsymbol{\mu}_{b(\boldsymbol{\theta}_i)}$ and covariance matrix $\boldsymbol{C}_{b(\boldsymbol{\theta}_i)}$ constitute the desired second-order description of the belief $b(\boldsymbol{\theta}_i)$; furthermore, $\boldsymbol{\mu}_{b(\boldsymbol{\theta}_i)}$ provides an approximation of the MMSE estimate of $\boldsymbol{\theta}_i$, and $\boldsymbol{C}_{b(\boldsymbol{\theta}_i)}$ provides an approximate characterization of the uncertainty of the estimate.

We note that according to (23), Step 1C requires the computation of the square root of the $\bar{L}_i \times \bar{L}_i$ matrix $\boldsymbol{C}_{\bar{\boldsymbol{\theta}}_{i*}}$. Because of (38), this matrix square root is given by

$$\boldsymbol{C}_{\bar{\boldsymbol{\theta}}_{i*}}^{1/2} = \mathrm{bdiag}\big\{\boldsymbol{C}_{\boldsymbol{\theta}_i}^{1/2}, \boldsymbol{C}_{j_1}^{1/2}, \boldsymbol{C}_{j_2}^{1/2}, \ldots, \boldsymbol{C}_{j_{|\mathcal{T}_i|}}^{1/2}, \boldsymbol{C}_{\bar{\boldsymbol{v}}_i}^{1/2}\big\}.$$

Hence, the computation of $\boldsymbol{C}_{\bar{\boldsymbol{\theta}}_{i*}}^{1/2}$ reduces to computing the smaller matrix square roots $\boldsymbol{C}_{\boldsymbol{\theta}_i}^{1/2}, \boldsymbol{C}_{j_1}^{1/2}, \boldsymbol{C}_{j_2}^{1/2}, \ldots, \boldsymbol{C}_{j_{|\mathcal{T}_i|}}^{1/2}$, $\boldsymbol{C}_{\bar{\boldsymbol{v}}_i}^{1/2}$. Here, in particular, $\boldsymbol{C}_{\bar{\boldsymbol{v}}_i}$ is a diagonal matrix, and thus $\boldsymbol{C}_{\bar{\boldsymbol{v}}_i}^{1/2}$ is the diagonal matrix whose diagonal elements are the square roots of the diagonal elements of $\boldsymbol{C}_{\bar{\boldsymbol{v}}_i}$. The remaining matrix square roots $\boldsymbol{C}_{\boldsymbol{\theta}_i}^{1/2}, \boldsymbol{C}_{j_1}^{1/2}, \boldsymbol{C}_{j_2}^{1/2}, \ldots, \boldsymbol{C}_{j_{|\mathcal{T}_i|}}^{1/2}$ can be computed efficiently by means of the Cholesky decomposition [44], whose complexity is cubic in the matrix dimension.

E. Computation and Communication Requirements

The computational complexity of the proposed SPBP algorithm for NLS is dominated by the calculation of the covariance matrices $\boldsymbol{C}_{\bar{\boldsymbol{y}}_i}$ and $\boldsymbol{C}_{\bar{\boldsymbol{\theta}}_i\bar{\boldsymbol{y}}_i}$ in Step 3C, which is analogous to the calculation of expressions (26) and (27). As a consequence, the complexity of the SPBP algorithm at one device i is cubic in $\big|\mathcal{T}_i\big|$ since the lengths of the vectors $\bar{\boldsymbol{\theta}}_i$ and $\bar{\boldsymbol{y}}_i$ as well as the number of SPs grow linearly with $\big|\mathcal{T}_i\big|$. Furthermore, the computational complexity is linear in the number of SPs (which is $2\bar{L}_i+1$, where $\bar{L}_i = 6 + 4|\mathcal{T}_i| + L_{\bar{\boldsymbol{v}}_i}$ with $L_{\bar{\boldsymbol{v}}_i} = \sum_{j\in\mathcal{T}_i}(K_{ij}+K_{ji})$). We note that the complexity of the hybrid particle-based/parametric BP method for NLS presented in [19] is linear in $\big|\mathcal{T}_i\big|$ and linear in the number of particles. However, the number of particles is much larger than the number of SPs in the SPBP algorithm. The results of our numerical study reported in Section VI demonstrate that

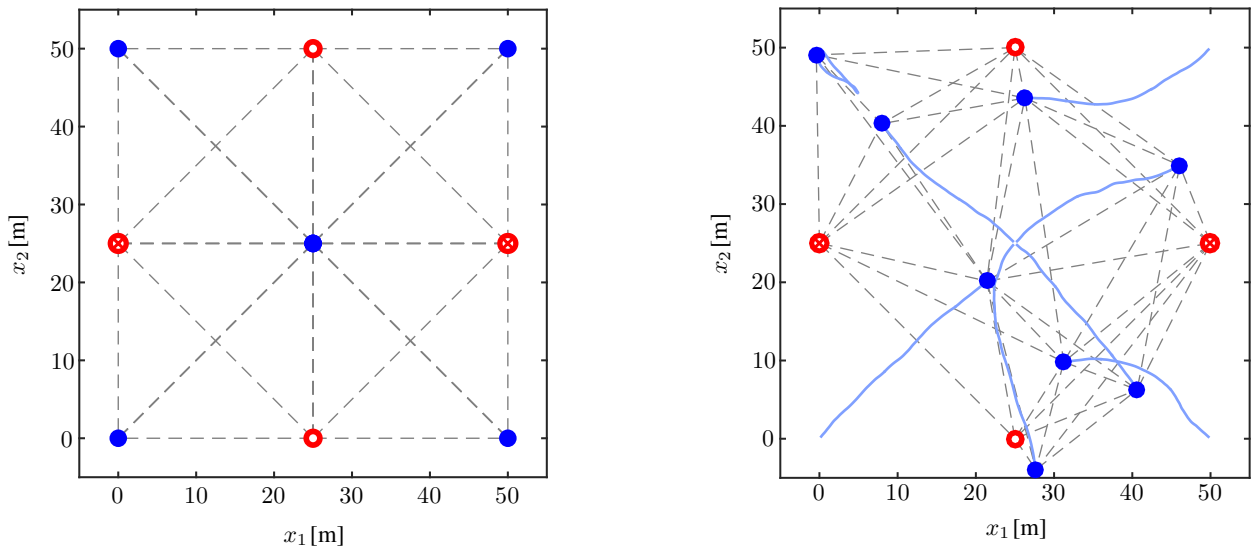


Fig. 4. Network topology (a) at time step $n=0$ and (b) at time step $n=120$. Blue bullets indicate the initial locations of mobile devices with unknown clock parameters and unknown locations, red circles indicate static devices with unknown clock parameters and known locations, red crossed circles indicate static devices with known clock parameters and known locations, grey dashed lines represent measurement/communication links, and blue solid lines represent a realization of the trajectories of the mobile devices. Note that in (a), the single blue bullet at the center of the scene represents the (identical) initial locations of four mobile devices.

in the scenario considered, the SPBP algorithm is significantly less complex than the method presented in [19].

The communication requirements of the SPBP algorithm are as follows. As a prerequisite for performing Step 1C, device i receives $\boldsymbol{\mu}_j$ and $\boldsymbol{C}_j$ from each neighbor $j \in \mathcal{T}_i$ (this is needed to calculate $\boldsymbol{\mu}_{\bar{\boldsymbol{\theta}}_{i*}}$ and $\boldsymbol{C}_{\bar{\boldsymbol{\theta}}_{i*}}$, see (37) and (38)), and it broadcasts $\boldsymbol{\mu}_i$ and $\boldsymbol{C}_i$ to all neighbors $j \in \mathcal{T}_i$. Thus, device i broadcasts a 4-dimensional mean vector and a 4×4 covariance matrix, which amounts to a total of 14 real values. These communication requirements are similar to those of the method in [19].

VI. Numerical Evaluation

In this section, we compare the performance and computational complexity of the proposed SPBP algorithm for NLS with those of the BP method presented in [19].

A. Simulation Setting

We simulated a network consisting of $|\mathcal{I}| = 12$ devices. Devices 1–4 are static, i.e., $\mathbf{x}_i^{(n)} = \mathbf{x}_i^{(0)}$ for $i = 1,2,3,4$ and all $n \geq 0$; these devices serve as spatial master devices. Devices 5–12 are mobile devices that follow the constant velocity motion model [45, Section 6.3.2]. That is, their state-evolution function in (2) is given by $\boldsymbol{g}_{1,i}(\boldsymbol{x}) = \boldsymbol{G}_1\boldsymbol{x}$ and the covariance matrix of the driving process $\mathbf{u}_{1,i}^{(n)}$ in (2) is given by $\boldsymbol{\Sigma}_{\boldsymbol{u}_{1,i}} = \sigma_1^2\boldsymbol{G}_2\boldsymbol{G}_2^{\mathrm{T}}$, where

$$\boldsymbol{G}_1 = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \boldsymbol{G}_2 = \begin{bmatrix} T^2/2 & 0 \\ 0 & T^2/2 \\ T & 0 \\ 0 & T \end{bmatrix},$$

and $\sigma_1 = 0.3\,\mathrm{m/s^2}$. The duration of one time step is $T = 0.2\,\mathrm{s}$. Devices 1 and 2 serve as temporal master devices with clock skew equal to 1 and clock phase equal to 0s, i.e., $\boldsymbol{\omega}_i^{(n)} = [1\ 0\mathrm{s}]^{\mathrm{T}}$ for $i = 1,2$ and all $n \geq 0$. The clock states of devices 3–12 evolve according to (3) with $\boldsymbol{g}_{2,i}(\boldsymbol{\omega}) = \boldsymbol{\omega}$ and $\boldsymbol{\Sigma}_{\boldsymbol{u}_{2,i}} = \mathrm{diag}\{T\sigma_2^2, T\sigma_3^2\}$, where $\sigma_2 = 10\,\mathrm{ppm}\cdot\mathrm{s}^{-1/2}$ and $\sigma_3 = 10^{-6}\,\mathrm{s}^{1/2}$. There are no self-measurements, i.e., $\mathcal{S}^{(n)} = \emptyset$ for all n.

We present the results of 500 simulation runs, each comprising 120 time steps. In each simulation run, new realizations of the sequences of clock parameters of devices 3–12 and of the trajectories of devices 5–12 are generated according to the state-evolution models described above. The initial location $\mathbf{p}_i^{(0)}$ of each device is the same for each simulation run, whereas the initial clock states $\boldsymbol{\omega}_i^{(0)}$ for $i \geq 3$ are generated independently from a Gaussian distribution with mean vector $[1\ 0\mathrm{s}]^{\mathrm{T}}$ and covariance matrix $\mathrm{diag}\{(150\,\mathrm{ppm})^2, (1\mathrm{s})^2\}$.

Each device exchanges $K_{ij} = K_{ji} = 10$ packets with each of its neighbor devices and obtains time measurements as described in Section II-B. The neighbors of device i at time n are all the devices $j \neq i$ within a radius of 37.5m, i.e., $\|\boldsymbol{p}_j^{(n)} - \boldsymbol{p}_i^{(n)}\| \leq 37.5\mathrm{m}$. We consider three different values of the standard deviation of the measurement noise, namely, $\sigma_{v_{i,j}} = \sigma_v \in \{1\mathrm{ns}, 5\mathrm{ns}, 10\mathrm{ns}\}$. Fig. 4 shows the network topology (i.e., the 12 devices and the measurement/communication links connecting them) at the initial time step $n = 0$ and at the final time step $n = 120$, as well as a realization of the trajectories of the mobile devices.

In the simulated algorithms, the SP parameter κ is set to 0. The initial prior distributions $f(\boldsymbol{\theta}_i^{(0)})$ (cf. (12)) are assumed to be Gaussian for all devices i that do not serve as master devices. More specifically, for devices 5–12, $\mathbf{x}_i^{(0)} \sim$

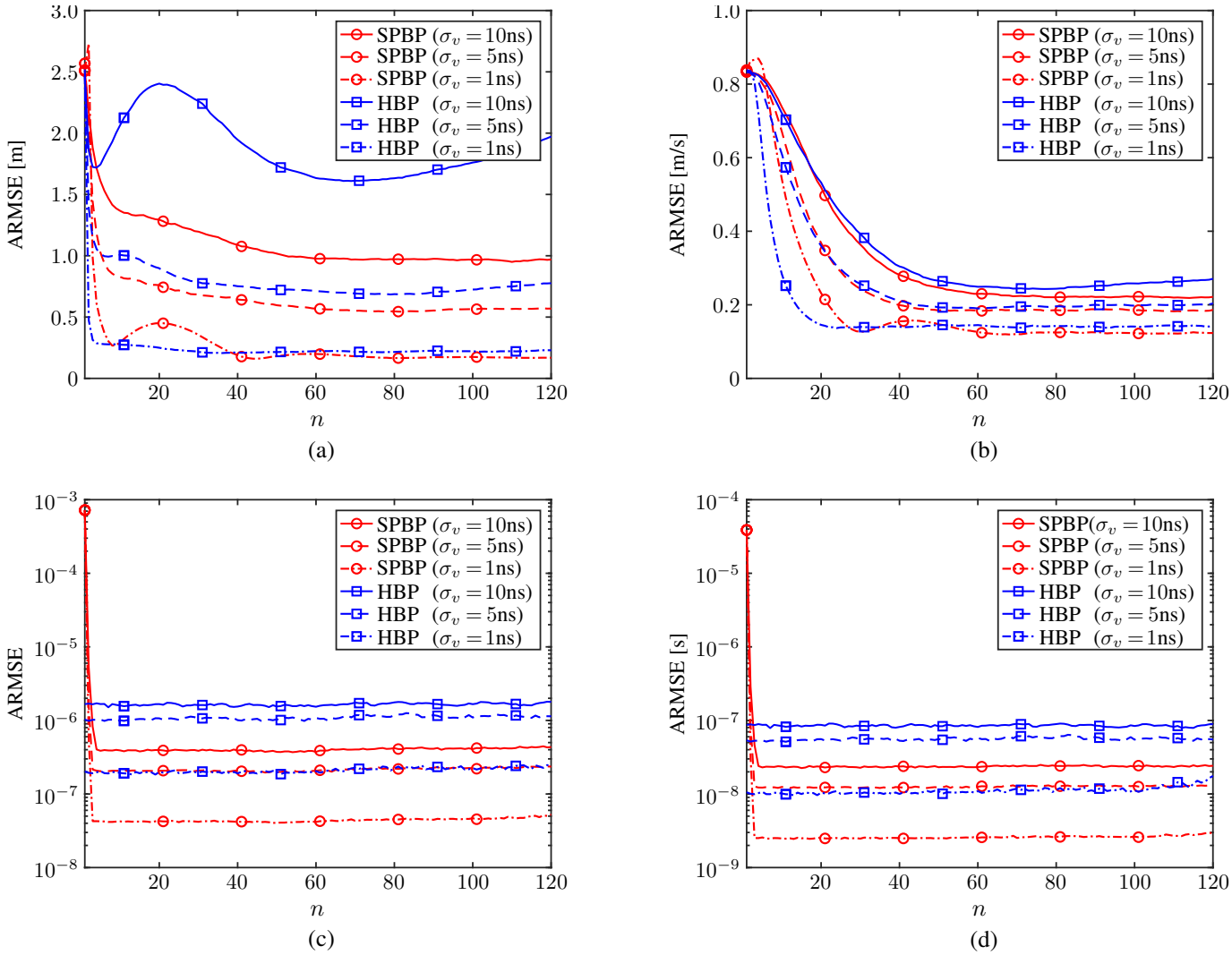


Fig. 5. Simulated ARMSEs of the proposed SPBP algorithm and the HBP algorithm versus times step n for three different measurement noise standard deviations σ_v: (a) location, (b) velocity, (c) clock skew, and (d) clock phase.

$\mathcal{N}\big(\boldsymbol{\mu}_{\boldsymbol{x},i}^{(0)}, \boldsymbol{\Sigma}_{\boldsymbol{x},i}^{(0)}\big)$ with $\boldsymbol{\mu}_{\boldsymbol{x},i}^{(0)} = \boldsymbol{x}_i^{(0)} + \boldsymbol{\epsilon}_i$ and $\boldsymbol{\Sigma}_{\boldsymbol{x},i}^{(0)} = \mathrm{diag}\big\{(3\mathrm{m})^2, (3\mathrm{m})^2, (1\mathrm{m/s})^2, (1\mathrm{m/s})^2\big\}$. Here, $\boldsymbol{x}_i^{(0)}$ is the true initial location-related state and $\boldsymbol{\epsilon}_i \sim \mathcal{N}\big(\mathbf{0}, \boldsymbol{\Sigma}_{\boldsymbol{x},i}^{(0)}\big)$ is drawn independently for all i and all simulation runs. Furthermore, for devices 3–12, $\boldsymbol{\omega}_i^{(0)} \sim \mathcal{N}\big(\boldsymbol{\mu}_{\boldsymbol{\omega},i}^{(0)}, \boldsymbol{\Sigma}_{\boldsymbol{\omega},i}^{(0)}\big)$ with $\boldsymbol{\mu}_{\boldsymbol{\omega},i}^{(0)} = [1\ 0\mathrm{s}]^{\mathrm{T}}$ and $\boldsymbol{\Sigma}_{\boldsymbol{\omega},i}^{(0)} = \mathrm{diag}\big\{(150\,\mathrm{ppm})^2, (1\mathrm{s})^2\big\}$.

B. Simulation Results

We simulated the proposed SPBP algorithm for NLS (briefly referred to as SPBP) and compared its results with those of the hybrid particle-based/parametric cooperative BP algorithm presented in [19] (briefly referred to as HBP). Our implementation of HBP used 1000 particles to represent the belief of each device and performed two message passing iterations in each time step. We did not consider any further state-of-the-art algorithms for NLS because, to the best of our knowledge, there are no other distributed algorithms that can handle time-variation of both the location and clock parameters. Our measure of performance is the average root mean square error (ARMSE), which is the root mean square error averaged over 500 simulation runs and over all devices.

Fig. 5 shows the ARMSEs of location, velocity, clock skew, and clock phase versus time n for three different values of the measurement noise standard deviation, $\sigma_{v_{i,j}} = \sigma_v \in \{1\mathrm{ns}, 5\mathrm{ns}, 10\mathrm{ns}\}$. It can be seen that at almost all times, the localization and synchronization accuracy of SPBP is better than that of HBP. This can be partly explained by the fact that the factor graph underlying HBP is "more loopy" and the number of particles used by HBP is rather small considering the dimensionality of the problem. (We note that a larger number of particles would result in a smaller ARMSE at the cost of a larger runtime.) In addition, the superior performance of SPBP relative to HBP in the simulated setting is also due to the dense network, i.e., the relatively high number of neighbor devices and, thus, available measurements. In sparse networks, the messages and beliefs often have complicated, multimodal shapes, in which case the second-order description provided by SPs may not be sufficient and a particle-based representation as used in [19] may be required. Finally, further

experiments demonstrate that in the absence of informative prior information at time $n = 0$, HBP tends to outperform SPBP during the initial time steps.

Using MATLAB implementations on a single core of an Intel Xeon E5-2640 v3 CPU, we obtained the following average runtimes per time step n: 0.91 s for SPBP and 1.93 s for HBP. Thus, in the chosen setting, SPBP outperforms HBP in terms of both estimation accuracy and runtime.

VII. Conclusion

Devices for the Internet of Things (IoT) often have limited computation and communication resources. They are typically heterogeneous in terms of qualities of internal clocks and mobility patterns. In this paper, we presented a distributed, cooperative, time-recursive algorithm for network localization and synchronization (NLS). The proposed algorithm is suited to heterogeneous devices with limited hardware capabilities, time-varying clock and location parameters, arbitrary state-evolution models, and time-varying network connectivity. Further advantages include low computational complexity and low communication requirements, and the fact that the computational complexity per device is independent of the size of the network. The algorithm uses time measurements obtained through asymmetric time-stamped communication. This measurement technique can be realized with emerging inexpensive and low-power ultra-wideband impulse radio modules, which are expected to play a key role in location-aware IoT devices. Therefore, the proposed algorithm is attractive for the decentralized wireless networks arising in IoT applications such as logistics, social networking, medical services, search and rescue operations, and automotive safety.

The main contributions of the paper can be summarized as follows:

- We formulated the joint synchronization and localization task as a network-wide spatiotemporal Bayesian estimation problem and represented the statistical structure of that problem by a factor graph. Based on the factor graph, we derived a belief propagation (BP) method for efficient marginalization of the joint posterior distribution of all parameters. Subsequently, we applied the recently introduced sigma point BP technique to develop an efficient second-order implementation. This resulted in a low-complexity, scalable inference algorithm for tracking the time-varying location and clock parameters of possibly heterogeneous IoT devices.
- We evaluated the performance of the developed algorithm through simulation. Our results show that in sufficiently dense networks, accurate synchronization and localization are achieved with a low computational complexity. In particular, the algorithm was observed to outperform the state-of-the-art BP-based algorithm for NLS with respect to both estimation accuracy and computational complexity.

Promising directions for future research include extensions of the proposed algorithm that are better suited to represent multimodal messages and beliefs. We conjecture that such extensions can be obtained by generalizing the sigma point BP technique [36] to multiple means and covariance matrices describing the modes, in the spirit of the Gaussian sum filter [46]. Similar extensions can possibly be based on the belief condensation filter [47]. We expect that the resulting extended algorithms will exhibit improved performance for sparse networks and in cases where no initial prior information on the device locations is available. Further promising research directions are the derivation of fundamental limits of synchronization and localization accuracy for the considered statistical model and the evaluation of our algorithm on a hardware testbed [48].

References

[1] S. G. Nagarajan, P. Zhang, and I. Nevat, "Geo-spatial location estimation for Internet of Things (IoT) networks with one-way time-of-arrival via stochastic censoring," *IEEE Internet of Things J.*, vol. 4, no. 1, pp. 205–214, Feb. 2017.

[2] C. Chen, Y. Chen, Y. Han, H.-Q. Lai, and K. J. R. Liu, "Achieving centimeter-accuracy indoor localization on WiFi platforms: A frequency hopping approach," *IEEE Internet of Things J.*, vol. 4, no. 1, pp. 111–121, Feb. 2017.

[3] J. Cho, J. Yu, S. Oh, J. Ryoo, J. Song, and H. Kim, "Wrong siren! A location spoofing attack on indoor positioning systems: The Starbucks case study," *IEEE Commun. Mag.*, vol. 55, no. 3, pp. 132–137, Mar. 2017.

[4] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, Oct. 2010.

[5] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A survey on enabling technologies, protocols, and applications," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 2347–2376, 2015.

[6] F. Meyer, T. Kropfreiter, J. L. Williams, R. A. Lau, F. Hlawatsch, P. Braca, and M. Z. Win, "Message passing algorithms for scalable multitarget tracking," *Proc. IEEE*, vol. 106, no. 2, pp. 221–259, Feb. 2018.

[7] 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.

[8] S. Gezici, Z. Tian, G. B. Giannakis, H. Kobayashi, A. F. Molisch, H. V. Poor, and Z. Sahinoglu, "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.

[9] Z. Liu, W. Dai, and M. Z. Win, "Mercury: An infrastructure-free system for network localization and navigation," *IEEE Trans. Mobile Comput.*, vol. IEEE Xplore Early Access, 2018.

[10] O. Bello and S. Zeadally, "Intelligent device-to-device communication in the Internet of Things," *IEEE Syst. J.*, vol. 10, no. 3, pp. 1172–1182, Sep. 2016.

[11] M. Z. Win, A. Conti, S. Mazuelas, Y. Shen, W. M. Gifford, D. Dardari, and M. Chiani, "Network localization and navigation via cooperation," *IEEE Commun. Mag.*, vol. 49, no. 5, pp. 56–62, May 2011.

[12] H. Wymeersch, J. Lien, and M. Z. Win, "Cooperative localization in wireless networks," *Proc. IEEE*, vol. 97, no. 2, pp. 427–450, Feb. 2009, special issue on *Ultra -Wide Bandwidth (UWB) Technology & Emerging Applications*.

[13] 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.

[14] S. Li, M. Hedley, and I. B. Collings, "New efficient indoor cooperative localization algorithm with empirical ranging error model," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 7, pp. 1407–1417, Jul. 2015.

[15] 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.

[16] D. Vasisht, S. Kumar, and D. Katabi, "Decimeter-level localization with a single WiFi access point," in *USENIX NSDI '16*, Santa Clara, CA, Mar. 2016, pp. 165–178.

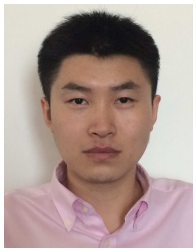
[17] Y. Shen, S. Mazuelas, and M. Z. Win, "Network navigation: Theory and interpretation," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 9, pp. 1823–1834, Oct. 2012.

[18] F. Meyer, O. Hlinka, H. Wymeersch, E. Riegler, and F. Hlawatsch, "Distributed localization and tracking of mobile networks including noncooperative objects," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 2, no. 1, pp. 57–71, Mar. 2016.
[19] B. Etzlinger, F. Meyer, F. Hlawatsch, A. Springer, and H. Wymeersch, "Cooperative simultaneous localization and synchronization in mobile agent networks," *IEEE Trans. Signal Process.*, vol. 65, no. 14, pp. 3587–3602, Jul. 2017.
[20] S. Zhu and Z. Ding, "Joint synchronization and localization using TOAs: A linearization based WLS solution," *IEEE J. Sel. Areas Commun.*, vol. 28, no. 7, pp. 1017–1025, Aug. 2010.
[21] S. P. Chepuri, G. Leus, and A.-J. van der Veen, "Joint localization and clock synchronization for wireless sensor networks," in *Proc. Asilomar Conf. Sig., Syst., Comput.*, Pacific Grove, CA, Nov. 2012, pp. 1432–1436.
[22] Y. Wang, X. Ma, and G. Leus, "Robust time-based localization for asynchronous networks," *IEEE Trans. Signal Process.*, vol. 59, no. 9, pp. 4397–4410, Sep. 2011.
[23] J. Zheng and Y.-C. Wu, "Joint time synchronization and localization of an unknown node in wireless sensor networks," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1309–1320, Mar. 2010.
[24] D. Zachariah, A. De Angelis, S. Dwivedi, and P. Händel, "Schedule-based sequential localization in asynchronous wireless networks," *EURASIP J. Adv. Sig. Process.*, vol. 16, 2014.
[25] R. M. Vaghefi and R. M. Buehrer, "Cooperative joint synchronization and localization in wireless sensor networks," *IEEE Trans. Signal Process.*, vol. 63, no. 14, pp. 3615–3627, Jul. 2015.
[26] A. Yeredor, "Decentralized TOA-based localization in non-synchronized wireless networks with partial, asymmetric connectivity," in *Proc. IEEE SPAWC-14*, Toronto, Canada, Jun. 2014, pp. 165–169.
[27] D. Benoît, J.-B. Pierrot, and C. Abou-Rjeily, "Joint distributed synchronization and positioning in UWB ad hoc networks using TOA," *IEEE Trans. Microw. Theory Tech.*, vol. 54, no. 4, pp. 1896–1911, Apr. 2006.
[28] F. Meyer, B. Etzlinger, F. Hlawatsch, and A. Springer, "A distributed particle-based belief propagation algorithm for cooperative simultaneous localization and synchronization," in *Proc. Asilomar Conf. Sig., Syst., Comput.*, Pacific Grove, CA, Nov. 2013, pp. 527–531.
[29] B. Etzlinger, F. Meyer, A. Springer, F. Hlawatsch, and H. Wymeersch, "Cooperative simultaneous localization and synchronization: A distributed hybrid message passing algorithm," in *Proc. Asilomar Conf. Sig., Syst., Comput.*, Pacific Grove, CA, Nov. 2013, pp. 1978–1982.
[30] B. Etzlinger, F. Meyer, H. Wymeersch, F. Hlawatsch, A. Springer, and G. Müller, "Cooperative simultaneous localization and synchronization: Toward a low-cost hardware implementation," in *Proc. IEEE SAM-14*, A Coruña, Spain, Jun. 2014, pp. 33–36.
[31] W. Yuan, N. Wu, B. Etzlinger, H. Wang, and J. Kuang, "Cooperative joint localization and clock synchronization based on Gaussian message passing in asynchronous wireless networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 9, pp. 7258–7273, Sep. 2016.
[32] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.
[33] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques*. Cambridge, MA: MIT Press, 2009.
[34] A. Orsino, A. Ometov, G. Fodor, D. Moltchanov, L. Militano, S. Andreev, O. N. C. Yilmaz, T. Tirronen, J. Torsner, G. Araniti, A. Iera, M. Dohler, and Y. Koucheryavy, "Effects of heterogeneous mobility on D2D- and drone-assisted mission-critical MTC in 5G," *IEEE Commun. Mag.*, vol. 55, no. 2, pp. 79–87, Feb. 2017.
[35] J. A. Stankovic, "Research directions for the Internet of Things," *IEEE Internet of Things J.*, vol. 1, no. 1, pp. 3–9, Feb. 2014.
[36] F. Meyer, O. Hlinka, and F. Hlawatsch, "Sigma point belief propagation," *IEEE Signal Process. Lett.*, vol. 21, no. 2, pp. 145–149, Feb. 2014.
[37] S. J. Julier and J. K. Uhlmann, "A new extension of the Kalman filter to nonlinear systems," in *Proc. AeroSense-97*, Orlando, FL, Apr. 1997, pp. 182–193.
[38] E. A. Wan and R. van der Merwe, "The unscented Kalman filter," in *Kalman Filtering and Neural Networks*, S. Haykin, Ed. New York, NY: Wiley, 2001, ch. 7, pp. 221–280.
[39] M. Z. Win and R. A. Scholtz, "Impulse radio: How it works," *IEEE Commun. Lett.*, vol. 2, no. 2, pp. 36–38, Feb. 1998.
[40] ——, "Ultra-wide bandwidth time-hopping spread-spectrum impulse radio for wireless multiple-access communications," *IEEE Trans. Commun.*, vol. 48, no. 4, pp. 679–691, Apr. 2000.
[41] D. Schneider, "You are here," *IEEE Spectrum*, vol. 50, no. 12, pp. 34–39, Dec. 2013.
[42] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Upper Saddle River, NJ: Prentice-Hall, 1993.
[43] S. Haykin, "Kalman filters," in *Kalman Filtering and Neural Networks*, S. Haykin, Ed. New York, NY: Wiley, 2001, ch. 1, pp. 1–21.
[44] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C: The Art of Scientific Computing*. New York, NY: Cambridge University Press, 1992.
[45] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. New York, NY: Wiley, 2001.
[46] T. Vercauteren and X. Wang, "Decentralized sigma-point information filters for target tracking in collaborative sensor networks," *IEEE Trans. Signal Process.*, vol. 53, no. 8, pp. 2997–3009, Aug. 2005.
[47] S. Mazuelas, Y. Shen, and M. Z. Win, "Belief condensation filtering," *IEEE Trans. Signal Process.*, vol. 61, no. 18, pp. 4403–4415, Sep. 2013.
[48] B. Teague, Z. Liu, F. Meyer, and M. Z. Win, "Peregrine: 3-D network localization and navigation," in *Proc. IEEE Latin-American Conf. Commun.*, Guatemala City, Guatemala, Nov. 2017, pp. 1–6.

Florian Meyer (S'12–M'15) received the Dipl.-Ing. (M.Sc.) and Ph.D. degrees in electrical engineering from TU Wien, Vienna, Austria in 2011 and 2015, respectively.

He was a visiting researcher with the Department of Signals and Systems, Chalmers University of Technology, Gothenburg, Sweden in 2013 and with the NATO Centre for Maritime Research and Experimentation (CMRE), La Spezia, Italy in 2014 and 2015. In 2016, he was with CMRE as a research scientist. Currently, Dr. Meyer is an Erwin Schrödinger Fellow at the Massachusetts Institute of Technology (MIT), Cambridge, MA, USA. He served as a TPC member of several IEEE conferences and is a co-chair of the IEEE ANLN Workshop at IEEE ICC 2018, Kansas City, MO, USA. His research interests include signal processing for wireless sensor networks, localization and tracking, information-seeking control, message passing algorithms, and finite set statistics.

Bernhard Etzlinger (S'11–M'16) received the Dipl.-Ing. (M.Sc.) degree in mechatronics in 2010 and the Dr. techn. (Ph.D.) degree in technical sciences in 2016, both from Johannes Kepler University Linz, Linz, Austria. In 2010, he worked at Fraunhofer FKIE, Wachtberg, Germany. During his Ph.D. studies, he was a visiting student with the Department of Signals and Systems, Chalmers University of Technology, Gothenburg, Sweden, with the Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX, USA, and with the Ming Hsieh Department of Electrical Engineering, University of Southern California, CA, USA. His research activities are focused on statistical signal processing for receiver design and cooperative network clock synchronization and localization.



Zhenyu Liu (S'15) received the B.Sc. (with honor) and the M.Sc. in electronic engineering from Tsinghua University, Beijing, China, in 2011 and 2014, respectively.

Mr. Liu is pursuing the Ph.D. degree with the Wireless Information and Network Sciences Laboratory at Massachusetts Institute of Technology (MIT), Cambridge, MA, USA. His research interests include statistical inference and stochastic optimization, with applications to communications and localization.

Mr. Liu received the Best Paper Award at IEEE LATINCOM in 2017, and the first prize of the IEEE Communications Society Student Competition in 2016. He received academic excellence scholarships from Tsinghua University from 2008 to 2010. He served as a reviewer for the Proceedings of the IEEE, the IEEE Transactions on Vehicular Technology, the IEEE Communications Letters, and the IEEE Wireless Communications Letters.

Franz Hlawatsch (S'85–M'88–SM'00–F'12) received the Diplom-Ingenieur, Dr. techn., and Univ.-Dozent (habilitation) degrees in electrical engineering/signal processing from TU Wien, Vienna, Austria in 1983, 1988, and 1996, respectively.

Since 1983, he has been with the Institute of Telecommunications, TU Wien, where he is currently an Associate Professor. During 1991–1992, as a recipient of an Erwin Schrödinger Fellowship, he spent a sabbatical year with the Department of Electrical Engineering, University of Rhode Island, Kingston, RI, USA. In 1999, 2000, and 2001, he held one-month Visiting Professor positions with INP/ENSEEIHT, Toulouse, France and IRCCyN, Nantes, France. He (co)authored a book, three review papers that appeared in the IEEE SIGNAL PROCESSING MAGAZINE, about 200 refereed scientific papers and book chapters, and three patents. He coedited three books.

Dr. Hlawatsch was a member of the IEEE SPCOM Technical Committee from 2004 to 2009. He was a Technical Program Co-Chair of EUSIPCO 2004 and served on the technical committees of numerous IEEE conferences. He was an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING from 2003 to 2007, for the IEEE TRANSACTIONS ON INFORMATION THEORY from 2008 to 2011, and for the IEEE TRANSACTIONS ON SIGNAL AND INFORMATION PROCESSING OVER NETWORKS from 2014 to 2017. He coauthored papers that won an IEEE Signal Processing Society Young Author Best Paper Award and a Best Student Paper Award at IEEE ICASSP 2011. His research interests include statistical and compressive signal processing methods and their application to sensor networks and wireless communications.

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 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.

Dr. Win 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.