Distributed Bernoulli Filtering Using Likelihood Consensus

Giuseppe Papa, Rene Repp, Florian Meyer, *Member, IEEE*, Paolo Braca, *Senior Member, IEEE*, and Franz Hlawatsch, *Fellow, IEEE*

***Abstract*—We consider the detection and tracking of a target in a decentralized sensor network. The presence of the target is uncertain, and the sensor measurements are affected by clutter and missed detections. The state-evolution model and the measurement model may be nonlinear and non-Gaussian. For this practically relevant scenario, we propose a particle-based distributed Bernoulli filter (BF) that provides to each sensor approximations of the Bayes-optimal estimates of the target presence probability and the target state. The proposed method uses all the measurements in the network while requiring only local intersensor communication. This is enabled by an extension of the likelihood consensus method that reaches consensus on the likelihood function under both the target presence and target absence hypotheses. We also propose an adaptive pruning of the likelihood expansion coefficients that yields a significant reduction of intersensor communication. Finally, we present a new variant of the likelihood consensus method that is suited to networks containing star-connected sensor groups. Simulation results show that in challenging scenarios, including a heterogeneous sensor network with significant noise and clutter, the performance of the proposed distributed BF approaches that of the optimal centralized multisensor BF. We also demonstrate that the proposed distributed BF outperforms a state-of-the-art distributed BF at the expense of a higher amount of intersensor communication.**

***Index Terms*—Bernoulli filter, distributed target tracking, distributed particle filtering, likelihood consensus, random finite set, sensor network.**

I. Introduction

A. Background and State of the Art

We CONSIDER the tracking of a single target based on measurements provided by multiple sensors. The target may be present or absent in arbitrary unknown time intervals. Therefore, the tracking method first decides if a target is present, i.e., it detects the target, and in the case of a positive detection result, it estimates the state of the target. An additional complication is the fact that the observed measurements are subject to a measurement origin uncertainty: it is possible that a measurement is not generated by the target (such a measurement is known as clutter or a false alarm) or that an existing target does not generate a measurement at a sensor (this situation is known as a missed detection) [1].

For measurements with a low signal-to-noise ratio, multiple sensors may be required for good performance. In the multiple-sensor setting, a distributed (decentralized), cooperative processing mode has important advantages [2]: it does not require a fusion center, communication between distant points, or complex routing strategies; it is robust to network node and link failures; it is able to adapt to changing network topologies; it scales well with the network size; and in a wireless sensor network scenario, it allows for low transmit powers and facilitates frequency reuse.

The Bernoulli filter (BF)—see [3] and references therein—provides a Bayes-optimal solution to the joint target detection and tracking problem. The target state is modeled as a random finite set (RFS) that is empty when the target is absent and has a single vector-valued element when the target is present. A Bayes-optimal extension of the BF to the case of multiple sensors was introduced in [4]. A particle-based BF that is suitable for arbitrary nonlinear and non-Gaussian state-space models and dynamically adapts to a variation in sensor performance was proposed in [5]. However, these methods are not distributed, i.e., they assume that the measurements of all the sensors are available at a single processing unit.

Two distributed BFs for decentralized networks were proposed recently. The *consensus BF* [6] uses Doppler shift measurements in the presence of measurement origin uncertainty. Single-sensor posterior probability density functions (pdfs) are transmitted to neighboring sensors and fused by means of a generalized covariance intersection technique [7]. The *random exchange BF* [8] uses non-thresholded received-signal-strength measurements. A diffusion technique [9] is employed in which

Manuscript received September 2, 2017; revised March 9, 2018, June 25, 2018, and September 18, 2018; accepted October 4, 2018. Date of publication December 17, 2018; date of current version May 8, 2019. This work was supported in part by the NATO Supreme Allied Command Transformation under projects SAC000601 and SAC000608 and in part by the Austrian Science Fund (FWF) under Grants P27370-N30 and J3886-N31. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Michael Rabbat. This paper was presented in part at FUSION 2018, Cambridge, U.K., July 2018. *(Corresponding author: Rene Repp.)*

G. Papa is with the Kognitiv Corporation, 1070 Vienna, Austria (e-mail: giuseppe.papa@kognitiv.com).

R. Repp and F. Hlawatsch are with the Institute of Telecommunications, TU Wien, 1040 Vienna, Austria, and also with Brno University of Technology, 60190 Brno, Czech Republic (e-mail: rene.repp@tuwien.ac.at; franz.hlawatsch@tuwien.ac.at).

F. Meyer is with the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: fmeyer@mit.edu).

P. Braca is with the NATO STO Centre for Maritime Research and Experimentation, 19126 La Spezia, Italy (e-mail: paolo.braca@cmre.nato.int).

Digital Object Identifier 10.1109/TSIPN.2018.2881718

each sensor exchanges (i.e., swaps) its local posterior pdf with a randomly chosen neighbor and broadcasts its local measurements to all neighbors. Then, both kinds of received information are fused with information related to the local measurements. Furthermore, a distributed multi-Bernoulli filter for the tracking of multiple targets was proposed in [10]. We note that all the mentioned distributed methods are based on the fusion of local posterior pdfs, which is not Bayes-optimal.

B. Contributions and Paper Organization

In this paper, we propose a distributed, particle-based, approximate implementation of the Bayes-optimal multisensor BF introduced in [4]. Our method provides each sensor with estimates of the time-varying target presence probability and of the time-varying target state. The method requires only local intersensor communication, i.e., each sensor communicates only with neighboring sensors. Because of its particle-based formulation, the method is suitable for arbitrary nonlinear and non-Gaussian measurement and state-evolution models. To the best of our knowledge, the proposed method is the first distributed multisensor BF that computes an approximation to the Bayes-optimal solution in a distributed way and for arbitrary measurement models.

The centralized multisensor BF in [4] is based on the joint likelihood function (JLF), which is not locally available at the individual sensors. The distributed BF algorithm proposed here uses the likelihood consensus (LC) method [11]–[13] for a distributed calculation of the JLF. The idea of LC is to expand the local log-likelihood function of each sensor into a dictionary of functions and then apply a distributed averaging algorithm [14], [15] for each expansion coefficient. In this way, an approximation of the JLF is obtained at each sensor through local communication. The approximate JLF is then used at the individual sensors in local BFs, whose performance approximates that of the centralized multisensor BF.

The LC method proposed in this paper extends the original LC of [11]–[13] in two different directions. First, as required by the BF algorithm, it reaches consensus on the JLF under both the target presence and target absence hypotheses. Second, it performs an adaptive pruning of the LC expansion coefficients via a thresholding in each consensus iteration. We demonstrate experimentally that this pruning results in a significant reduction of intersensor communication without compromising the detection and tracking performance. In addition, we present an LC variant with reduced communication cost in the case of networks containing star-connected sensor groups.

The remainder of this paper is organized as follows. The system model is described in Section II. Section III reviews the centralized BF and its particle-based implementation. Section IV develops an LC-based distributed computation of the JLF under both the target presence and target absence hypotheses. Section V extends this computation by including an adaptive pruning in each consensus iteration. The proposed distributed BF is presented in Section VI. A new variant of the LC suited to networks containing star-connected sensor groups is presented in Section VII. Finally, Section VIII reports simulation results that demonstrate the performance and communication requirements of our method and compare them with those of the state-of-the-art distributed BF proposed in [8]. This paper advances beyond our conference publication [16] by presenting the following additional contributions: a new variant of the LC for networks with star-connected sensor groups (Section VII), simulation results for a homogeneous sensor network with identical range-bearing sensors (Section VIII-B), and an experimental analysis of the variation of the communication cost in the course of the consensus iterations and due to pruning (Fig. 5).

II. System Model

We consider a network of S sensors that monitor a given surveillance region with the aim of detecting the presence of a single target and tracking the time-varying target state.

A. Target State Model

At a given time scan $k \in \{1, 2, \ldots\}$, the target may be absent or present. This fact is accounted for by modeling the target state at time k by a *Bernoulli RFS* [3], [17]

$$X_k \triangleq \begin{cases} \emptyset, & \text{if the target is absent,} \\ \{\boldsymbol{x}_k\}, & \text{if the target is present.} \end{cases}$$

Here, $\boldsymbol{x}_k \in \mathbb{R}^d$ is a random state vector that exists when the target is present. The associated RFS pdf [3] is

$$f(X_k) = \begin{cases} 1 - q_k, & X_k = \emptyset, \\ q_k f(\boldsymbol{x}_k), & X_k = \{\boldsymbol{x}_k\}, \end{cases} \tag{1}$$

where $q_k \triangleq \Pr(X_k \neq \emptyset)$ is the probability of target presence and $f(\boldsymbol{x}_k)$ is the pdf of the state vector. The state X_k evolves in time according to the state transition pdf [3]

$$f(X_k|X_{k-1}) = \begin{cases} 1 - P_{\mathrm{b}}, & X_k = \emptyset, X_{k-1} = \emptyset, \\ P_{\mathrm{b}} f_{\mathrm{b}}(\boldsymbol{x}_k), & X_k = \{\boldsymbol{x}_k\}, X_{k-1} = \emptyset, \\ 1 - P_{\mathrm{s}}, & X_k = \emptyset, X_{k-1} = \{\boldsymbol{x}_{k-1}\}, \\ P_{\mathrm{s}} f(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}), & X_k = \{\boldsymbol{x}_k\}, X_{k-1} = \{\boldsymbol{x}_{k-1}\}, \end{cases}$$

where P_{b} is the target birth probability, P_{s} is the target survival probability, $f_{\mathrm{b}}(\boldsymbol{x}_k)$ is the target birth pdf, and $f(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})$ is the state vector transition pdf. We note that $f(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})$ is usually derived from a state vector evolution model of the form [1]

$$\boldsymbol{x}_k = \boldsymbol{g}(\boldsymbol{x}_{k-1}, \boldsymbol{v}_k), \tag{2}$$

where $\boldsymbol{g}$ is a possibly nonlinear state transition function and $\boldsymbol{v}_k$ is a process noise vector that is independent across time k.

B. Measurement Model

1) Measurement Set: We use the measurement origin uncertainty model from [1], [17]. At time k, sensor $s \in \{1, \ldots, S\}$ observes the set of measurements

$$Z_k^{(s)} \triangleq \{\boldsymbol{z}_{k,i}^{(s)}\}_{i=1}^{|Z_k^{(s)}|},$$

where $\boldsymbol{z}_{k,i}^{(s)}$ is a measurement vector (briefly termed "measurement") and $|Z_k^{(s)}|$ denotes the cardinality of the set $Z_k^{(s)}$. We assume that there occurs one of the following two events: (i) one of the measurements in $Z_k^{(s)}$ is generated by the target (hence, this measurement is termed *target-originated measurement*) and all the other measurements are false alarms (termed *clutter-originated measurements* or simply *clutter*); (ii) all the measurements are false alarms. In the first event, the clutter-originated measurements are statistically independent—both unconditionally and conditioned on the target states—of the target-originated measurement. When the target is absent, only the second event is possible. When the target is present, the first event occurs with probability $P_{\mathrm{d}}^{(s)}(\boldsymbol{x}_k)$ and the second event occurs with probability $1 - P_{\mathrm{d}}^{(s)}(\boldsymbol{x}_k)$. The probability $P_{\mathrm{d}}^{(s)}(\boldsymbol{x}_k)$ is a sensor parameter and is referred to as *probability of detection*. Note that $P_{\mathrm{d}}^{(s)}(\boldsymbol{x}_k)$ can depend on $\boldsymbol{x}_k$. The clutter-originated measurements are modeled as independent and identically distributed (i.i.d.). The measurement set $Z_k^{(s)}$ may also be empty, which means that there is no clutter and the target, if present, was not detected by the sensor.

2) Local Likelihood Function: The measurement characteristic of sensor s is described by the *local likelihood function* $f(Z_k^{(s)}|X_k)$. Regarding the dependence on X_k, we need to distinguish the cases $X_k = \emptyset$ and $X_k = \{\boldsymbol{x}_k\}$, resulting in the functions $f(Z_k^{(s)}|\emptyset)$ and $f(Z_k^{(s)}|\{\boldsymbol{x}_k\})$, respectively.

When $X_k = \emptyset$, all the measurements are clutter and $Z_k^{(s)}$ is an *i.i.d. cluster RFS* [3], whose pdf is given by

$$f\big(Z_k^{(s)}|\emptyset\big) = \tilde{f}_{\mathrm{c}}^{(s)}\big(Z_k^{(s)}\big) \triangleq |Z_k^{(s)}|!\, p_{\mathrm{c}}^{(s)}\big(|Z_k^{(s)}|\big) \prod_{\boldsymbol{z}\in Z_k^{(s)}} f_{\mathrm{c}}^{(s)}(\boldsymbol{z}). \tag{3}$$

Here, $p_{\mathrm{c}}^{(s)}(n)$ is the probability mass function (pmf) of the number of clutter measurements and $f_{\mathrm{c}}^{(s)}(\boldsymbol{z})$ is the pdf of the clutter measurements. Note that $p_{\mathrm{c}}^{(s)}(n)$ and $f_{\mathrm{c}}^{(s)}(\boldsymbol{z})$ can be sensor-dependent. For $Z_k^{(s)} = \emptyset$, Eq. (3) reduces to $\tilde{f}_{\mathrm{c}}^{(s)}(\emptyset) = p_{\mathrm{c}}^{(s)}(0)$.

When $X_k = \{\boldsymbol{x}_k\}$, the local likelihood function $f(Z_k^{(s)}|X_k) = f(Z_k^{(s)}|\{\boldsymbol{x}_k\})$ is given by [3, Sec. V-A]

$$f(Z_k^{(s)}|\{\boldsymbol{x}_k\}) = \big(1 - P_{\mathrm{d}}^{(s)}(\boldsymbol{x}_k)\big)\tilde{f}_{\mathrm{c}}^{(s)}\big(Z_k^{(s)}\big) + P_{\mathrm{d}}^{(s)}(\boldsymbol{x}_k) \times \sum_{\boldsymbol{z}\in Z_k^{(s)}} f^{(s)}(\boldsymbol{z}|\boldsymbol{x}_k)\tilde{f}_{\mathrm{c}}^{(s)}\big(Z_k^{(s)}\backslash\{\boldsymbol{z}\}\big). \tag{4}$$

Here, the "vector likelihood function" $f^{(s)}(\boldsymbol{z}|\boldsymbol{x}_k)$ can be derived from a target-originated measurement model of the form

$$\boldsymbol{z}_k^{(s)} = \boldsymbol{h}^{(s)}\big(\boldsymbol{x}_k, \boldsymbol{w}_k^{(s)}\big), \tag{5}$$

where $\boldsymbol{h}^{(s)}$ is the (generally nonlinear) measurement function of sensor s and $\boldsymbol{w}_k^{(s)}$ is a measurement noise vector that is independent across k and s. For $Z_k^{(s)} = \emptyset$, Eq. (4) reduces to $f(\emptyset|\{\boldsymbol{x}_k\}) = (1 - P_{\mathrm{d}}^{(s)}(\boldsymbol{x}_k))\tilde{f}_{\mathrm{c}}^{(s)}(\emptyset) = (1 - P_{\mathrm{d}}^{(s)}(\boldsymbol{x}_k))p_{\mathrm{c}}^{(s)}(0)$.

3) Joint Likelihood Function (JLF): The JLF is defined as $f(Z_k|X_k)$, where $Z_k \triangleq (Z_k^{(s)})_{s=1}^S$ is an ordered sequence of the measurement sets of all the sensors. Since the measurements from different sensors are conditionally independent given the target state X_k, the JLF is equal to the product of the local likelihood functions of all the sensors, i.e.,

$$f(Z_k|X_k) = \prod_{s=1}^S f\big(Z_k^{(s)}|X_k\big). \tag{6}$$

Here, $f(Z_k^{(s)}|X_k)$ is given by (3) if $X_k = \emptyset$ and by (4) if $X_k = \{\boldsymbol{x}_k\}$.

C. Intersensor Communication

Each sensor $s \in \{1, \ldots, S\}$ is assumed to be able to communicate with a subset of the other sensors, $\mathcal{S}_s \subseteq \{1, \ldots, S\}\backslash\{s\}$, which will be referred to as the set of "neighbors" of sensor s. Typically, $\mathcal{S}_s$ depends on the coverage region of the communication system with which sensor s is equipped [18], [19]. Communication is assumed symmetric, i.e., $s' \in \mathcal{S}_s$ implies $s \in \mathcal{S}_{s'}$. The communication graph of the sensor network, which is defined by all neighbor sets $\mathcal{S}_s$, is assumed to be connected.

III. Review of Centralized Bernoulli Filtering

As a basis for developing the proposed distributed BF, we first review the centralized BF and its particle-based implementation [3]. For the centralized BF, we allow for multiple sensors but assume that $Z_k = (Z_k^{(s)})_{s=1}^S$ is available at a central processing unit.

A. The BF Recursion

The centralized BF is a sequential Bayesian estimator that computes the posterior pdf of the target state X_k given all the measurements up to the current time k, $Z_{1:k} \triangleq (Z_i)_{i=1}^k$ [3]. This posterior pdf is of the Bernoulli form (1), i.e.,

$$f(X_k|Z_{1:k}) \triangleq \begin{cases} 1 - q_{k|k}, & X_k = \emptyset, \\ q_{k|k} f(\boldsymbol{x}_k|Z_{1:k}), & X_k = \{\boldsymbol{x}_k\}, \end{cases} \tag{7}$$

where $q_{k|k} \triangleq \Pr(X_k \neq \emptyset|Z_{1:k})$ is the posterior probability of target presence and $f(\boldsymbol{x}_k|Z_{1:k})$ is the posterior pdf of the target state vector. The posterior pdf $f(X_k|Z_{1:k})$ is calculated recursively via the *prediction step*

$$f(X_k|Z_{1:k-1}) = \int f(X_k|X_{k-1})f(X_{k-1}|Z_{1:k-1})\,\delta X_{k-1} \tag{8}$$

and the subsequent *update step*

$$f(X_k|Z_{1:k}) = \frac{f(Z_k|X_k)f(X_k|Z_{1:k-1})}{f(Z_k|Z_{1:k-1})}, \tag{9}$$

where

$$f(Z_k|Z_{1:k-1}) = \int f(Z_k|X_k)f(X_k|Z_{1:k-1})\,\delta X_k\,. \tag{10}$$

Here, the integrals in (8) and (10) are set integrals [3]. We note that $f(X_k|Z_{1:k-1})$ in (8) is the *predicted* posterior pdf, which represents the knowledge about the state X_k given all the measurements up to the previous time $k - 1$. Furthermore, Eq. (9) incorporates the measurements Z_k observed at the current time

k. Eqs. (8)–(10) provide a recursive calculation of $f(X_k|Z_{1:k})$ from $f(X_{k-1}|Z_{1:k-1})$. This recursion is initialized at time $k=1$ by setting $f(X_0|Z_{1:0})$ on the right-hand side of (8) equal to an initial prior Bernoulli pdf $f(X_0)$, which is characterized by a suitably chosen prior target presence probability q_0 and a suitably chosen prior state vector pdf $f(\boldsymbol{x}_0)$ according to (1).

It can be shown [3] that if the previous posterior pdf $f(X_{k-1}|Z_{1:k-1})$ is a Bernoulli pdf characterized by $q_{k-1|k-1}$ and $f(\boldsymbol{x}_{k-1}|Z_{1:k-1})$, then the predicted posterior pdf $f(X_k|Z_{1:k-1})$ in (8) is a Bernoulli pdf as well, i.e.,

$$f(X_k|Z_{1:k-1}) = \begin{cases} 1-q_{k|k-1}, & X_k=\emptyset, \\ q_{k|k-1} f(\boldsymbol{x}_k|Z_{1:k-1}), & X_k=\{\boldsymbol{x}_k\}, \end{cases}$$

where

$$q_{k|k-1} = P_{\text{s}} q_{k-1|k-1} + P_{\text{b}}(1-q_{k-1|k-1}) \tag{11}$$

and

$$\begin{aligned} f(\boldsymbol{x}_k|Z_{1:k-1}) &= \frac{P_{\text{s}} q_{k-1|k-1}}{q_{k|k-1}} \int f(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}) f(\boldsymbol{x}_{k-1}|Z_{1:k-1}) \mathrm{d}\boldsymbol{x}_{k-1} \\ &\quad + \frac{P_{\text{b}}(1-q_{k-1|k-1})}{q_{k|k-1}} f_{\text{b}}(\boldsymbol{x}_k). \end{aligned} \tag{12}$$

Furthermore, also the posterior pdf $f(X_k|Z_{1:k})$ in (9) is a Bernoulli pdf [3], i.e.,

$$f(X_k|Z_{1:k}) = \begin{cases} 1-q_{k|k}, & X_k=\emptyset, \\ q_{k|k} f(\boldsymbol{x}_k|Z_{1:k}), & X_k=\{\boldsymbol{x}_k\}, \end{cases}$$

where

$$q_{k|k} = 1 - \frac{f(Z_k|\emptyset)(1-q_{k|k-1})}{f(Z_k|Z_{1:k-1})} \tag{13}$$

and

$$f(\boldsymbol{x}_k|Z_{1:k}) = \frac{f(Z_k|\{\boldsymbol{x}_k\}) f(\boldsymbol{x}_k|Z_{1:k-1})}{C(Z_{1:k})}, \tag{14}$$

with $C(Z_{1:k}) \triangleq \int f(Z_k|\{\boldsymbol{x}_k\}) f(\boldsymbol{x}_k|Z_{1:k-1}) \mathrm{d}\boldsymbol{x}_k$. The constant $f(Z_k|Z_{1:k-1})$ in (13) (cf. (10)) is given by

$$f(Z_k|Z_{1:k-1}) = C(Z_{1:k}) q_{k|k-1} + f(Z_k|\emptyset)(1-q_{k|k-1}). \tag{15}$$

Note that (13)–(15) involve the JLF $f(Z_k|X_k)$ in (6), which in turn involves the measurements of all the sensors.

B. Detection and Estimation

Target detection is performed by comparing $q_{k|k}$ in (13) to a threshold P_{th}, i.e., the target is considered to be present if $q_{k|k} > P_{\text{th}}$ [20, Ch. 2]. If the target is considered to be present, then an estimate of the state vector $\boldsymbol{x}_k$ is produced by the minimum mean-square error (MMSE) estimator [20, Ch. 4]

$$\hat{\boldsymbol{x}}_k^{\text{MMSE}} \triangleq \int \boldsymbol{x}_k f(\boldsymbol{x}_k|Z_{1:k}) \mathrm{d}\boldsymbol{x}_k, \tag{16}$$

with $f(\boldsymbol{x}_k|Z_{1:k})$ given by (14). This Bayesian two-stage detection-estimation procedure is also employed by the JITS filter [21], the JIPDA filter [22], and certain other RFS-based filters besides the BF, such as [23] and [24].

C. Particle-Based Implementation

Typically, the integrals involved in (12)–(16) do not admit closed-form expressions and can be computed only approximately. For a particle-based approximate implementation of the centralized BF [3], the posterior pdf $f(X_k|Z_{1:k})$ in (7) is represented by an approximate target presence probability $\hat{q}_{k|k}$ and a set $\{(\boldsymbol{x}_k^{(i)}, w_k^{(i)})\}_{i=1}^{I_{\text{p}}}$ of I_{p} particles $\boldsymbol{x}_k^{(i)}$ and associated weights $w_k^{(i)}$ that represent the posterior state vector pdf $f(\boldsymbol{x}_k|Z_{1:k})$.

1) Prediction Step: In the prediction step, the posterior target presence probability is predicted according to (11), i.e.,

$$\hat{q}_{k|k-1} = P_{\text{s}} \hat{q}_{k-1|k-1} + P_{\text{b}}(1-\hat{q}_{k-1|k-1}). \tag{17}$$

Furthermore, the predicted state vector pdf $f(\boldsymbol{x}_k|Z_{1:k-1})$ in (12) is represented by weighted particles $\{(\boldsymbol{x}_{k|k-1}^{(i)}, w_{k|k-1}^{(i)})\}_{i=1}^{I_{\text{p}}+I_{\text{b}}}$, where the $\boldsymbol{x}_{k|k-1}^{(i)}$ are sampled from two different distributions: for $i=1,\ldots,I_{\text{p}}$, they are drawn from the pdf $f(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}^{(i)})$, i.e., the state vector transition pdf $f(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})$ evaluated at $\boldsymbol{x}_{k-1} = \boldsymbol{x}_{k-1}^{(i)}$, the respective previous particle; and for $i = I_{\text{p}}+1,\ldots,I_{\text{p}}+I_{\text{b}}$, they are drawn from the target birth pdf $f_{\text{b}}(\boldsymbol{x}_k)$. The resulting two types of particles $\boldsymbol{x}_{k|k-1}^{(i)}$ will be referred to as *predicted particles* and *birth particles*, respectively. According to (12), the corresponding weights are obtained as follows [25]: for $i=1,\ldots,I_{\text{p}}$,

$$w_{k|k-1}^{(i)} = \frac{P_{\text{s}} \hat{q}_{k-1|k-1}}{\hat{q}_{k|k-1}} w_{k-1}^{(i)}, \tag{18}$$

and for $i = I_{\text{p}}+1,\ldots,I_{\text{p}}+I_{\text{b}}$,

$$w_{k|k-1}^{(i)} = \frac{P_{\text{b}}(1-\hat{q}_{k-1|k-1})}{\hat{q}_{k|k-1}} \frac{1}{I_{\text{b}}}. \tag{19}$$

Note that (18) and (19), together with the specific sampling of the particles $\boldsymbol{x}_{k|k-1}^{(i)}$ described earlier, are particle-based computations of, respectively, the first and second component of the pdf $f(\boldsymbol{x}_k|Z_{1:k-1})$ in (12). The overall particle-based computation of $f(\boldsymbol{x}_k|Z_{1:k-1})$ relies on the principle that particles can be drawn from a weighted mixture of component pdfs by drawing particles from each component pdf, weighting the particles by incorporating the component weight, and combining them in a joint particle set [26]. Based on the importance sampling principle [27, Sec. 1.3.2], this particle representation of $f(\boldsymbol{x}_k|Z_{1:k-1})$ will be used in the update step, where $f(\boldsymbol{x}_k|Z_{1:k-1})$ is employed as a proposal pdf.

2) Update Step: In the update step, the posterior target presence probability is calculated according to (13), i.e.,

$$\hat{q}_{k|k} = 1 - \frac{f(Z_k|\emptyset)(1-\hat{q}_{k|k-1})}{\hat{f}(Z_k|Z_{1:k-1})}. \tag{20}$$

Here, $\hat{f}(Z_k|Z_{1:k-1})$ is calculated according to (15), in which $C(Z_{1:k})$ is evaluated via Monte Carlo integration [27, Sec. 1.3.1], i.e.,

$$\hat{f}(Z_k|Z_{1:k-1}) = \hat{C}(Z_{1:k})\hat{q}_{k|k-1} + f(Z_k|\emptyset)(1-\hat{q}_{k|k-1}), \quad (21)$$

with $\hat{C}(Z_{1:k}) \triangleq \sum_{i=1}^{I_\mathrm{p}+I_\mathrm{b}} f(Z_k|\{\boldsymbol{x}_{k|k-1}^{(i)}\})w_{k|k-1}^{(i)}$. Subsequently, weighted particles representing $f(\boldsymbol{x}_k|Z_{1:k})$ in (14) are calculated according to the importance sampling principle by using $f(\boldsymbol{x}_k|Z_{1:k-1})$ as proposal pdf. This leads to the updated weights

$$\tilde{w}_k^{(i)} = f\big(Z_k\big|\{\boldsymbol{x}_{k|k-1}^{(i)}\}\big)w_{k|k-1}^{(i)}, \quad i = 1, \ldots, I_\mathrm{p}+I_\mathrm{b}. \quad (22)$$

(Note that the particle representation $\{(\boldsymbol{x}_{k|k-1}^{(i)}, w_{k|k-1}^{(i)})\}_{i=1}^{I_\mathrm{p}+I_\mathrm{b}}$ of $f(\boldsymbol{x}_k|Z_{1:k-1})$ has been calculated in the prediction step.) Next, from the current set of particles and weights $\{(\boldsymbol{x}_{k|k-1}^{(i)}, \tilde{w}_k^{(i)})\}_{i=1}^{I_\mathrm{p}+I_\mathrm{b}}$, only the I_p particles with the highest weights are retained, and their weights are normalized such that their sum is 1. Denoting the retained I_p particles as $\boldsymbol{x}_k^{(i)}$ and the corresponding normalized weights as $w_k^{(i)}$, with a re-indexing such that $i \in \{1, \ldots, I_\mathrm{p}\}$, we now have a set of particles and normalized weights $\{(\boldsymbol{x}_k^{(i)}, w_k^{(i)})\}_{i=1}^{I_\mathrm{p}}$ that provides an approximate representation of $f(\boldsymbol{x}_k|Z_{1:k})$. Finally, in order to avoid particle degeneracy, a resampling step is executed if the *effective sample size* [28] is below a given threshold.

The above prediction and update steps provide a recursive calculation of the approximate target presence probability $\hat{q}_{k|k}$ and of the weighted particles $\{(\boldsymbol{x}_k^{(i)}, w_k^{(i)})\}_{i=1}^{I_\mathrm{p}}$ representing $f(\boldsymbol{x}_k|Z_{1:k})$. This recursion is initialized at time $k = 1$ by setting $\hat{q}_{0|0}$ equal to the initial prior target presence probability q_0 (see Section III-A), drawing the particles $\{\boldsymbol{x}_0^{(i)}\}_{i=1}^{I_\mathrm{p}}$ from the initial prior vector pdf $f(\boldsymbol{x}_0)$ (again see Section III-A), and choosing equal weights $\{w_0^{(i)} = 1/I_\mathrm{p}\}_{i=1}^{I_\mathrm{p}}$.

3) Detection and Estimation: Following Section III-B, the target is considered to be present if $\hat{q}_{k|k}$ is above a threshold P_th. In that case, a particle-based approximation of the MMSE state estimate $\hat{\boldsymbol{x}}_k^\mathrm{MMSE}$ in (16) is calculated as

$$\hat{\boldsymbol{x}}_k = \sum_{i=1}^{I_\mathrm{p}} w_k^{(i)}\boldsymbol{x}_k^{(i)}.$$

IV. Distributed Computation of the JLF

In the distributed scenario, each sensor s in the network runs a local instance of a BF. This *local BF* achieves quasi-global performance by taking into account the measurements of *all* the sensors up to time k, i.e., the total measurement $Z_{1:k}$, which comprises $Z_{k'} = \big(Z_{k'}^{(s)}\big)_{s=1}^{S}$ for $k' = 1, \ldots, k$. This is done by using at each time recursion the JLF $f(Z_k|X_k)$. Because only the local likelihood function $f(Z_k^{(s)}|X_k)$ is available at sensor s, the proposed distributed BF includes a distributed computation of the JLF by means of a consensus-based algorithm. This computation uses the local weighted particles $\{(\boldsymbol{x}_{k|k-1}^{(s,i)}, w_{k|k-1}^{(s,i)})\}_{i=1}^{m}$, with $m \triangleq I_\mathrm{p} + I_\mathrm{b}$, which represent the predicted vector pdf $f(\boldsymbol{x}_k|Z_{1:k-1})$ as discussed in Section III-C1. The distributed computation of the JLF employs only local communication between neighboring sensors and does not require transmission of the sensor measurements.

From (6), the (natural) logarithm of the JLF is obtained as $\log f(Z_k|X_k) = \sum_{s=1}^{S} \log f(Z_k^{(s)}|X_k)$. Dividing and multiplying by S and exponentiating yields the JLF as

$$f(Z_k|X_k) = \exp\big(SL_k(X_k)\big), \quad (23)$$

with

$$L_k(X_k) \triangleq \frac{1}{S}\log f(Z_k|X_k) = \frac{1}{S}\sum_{s=1}^{S}\log f\big(Z_k^{(s)}|X_k\big). \quad (24)$$

(Note that the dependence of $L_k(X_k)$ on Z_k is not shown by our notation.) According to (23), the computation of the JLF amounts to computing $L_k(X_k)$, which according to (24) is the average of the local log-likelihood functions of all the sensors. To develop a consensus-based distributed computation of $L_k(X_k)$, we separately consider the two hypotheses of target absence ($X_k = \emptyset$) and target presence ($X_k = \{\boldsymbol{x}_k\}$).

A. Hypothesis $X_k = \emptyset$

For $X_k = \emptyset$, Eq. (24) simplifies to

$$L_k(\emptyset) = \frac{1}{S}\sum_{s=1}^{S}\log f(Z_k^{(s)}|\emptyset). \quad (25)$$

This is the average of the S constants $\log f(Z_k^{(s)}|\emptyset)$ and can thus be computed iteratively by the *average consensus algorithm* [14], using only communication with the neighbor sensors $s' \in \mathcal{S}_s$ (see Section II-C). Each sensor s has an "internal state," which it updates at each consensus iteration. More specifically, at iteration j, sensor s broadcasts its previous internal state value $u_{j-1}^{(s)}$ to its neighbor sensors $s' \in \mathcal{S}_s$ and receives their previous values $u_{j-1}^{(s')}$ from them. Then, sensor s updates its internal state according to

$$u_j^{(s)} = \sum_{s' \in \{s\} \cup \mathcal{S}_s} \gamma_{s,s'} u_{j-1}^{(s')}, \quad j = 1, 2, \ldots, \quad (26)$$

where the weights $\gamma_{s,s'}$ satisfy $\sum_{s' \in \{s\} \cup \mathcal{S}_s} \gamma_{s,s'} = 1$. If the internal state is initialized as

$$u_0^{(s)} = \log f\big(Z_k^{(s)}|\emptyset\big)$$

and the weights $\gamma_{s,s'}$ are chosen appropriately (see below), then for $j \to \infty$ the consensus recursion (26) is guaranteed to converge to $L_k(\emptyset)$ in (25) [29]. In practice, only a finite number J of iterations is executed, which is either fixed or chosen online by terminating the contribution of sensor s to the consensus iterations as soon as $|u_j^{(s)} - u_{j-1}^{(s)}|$ is below a threshold. The resulting final internal state $u_J^{(s)}$ then provides an approximation of $L_k(\emptyset)$, i.e., $L_k(\emptyset) \approx u_J^{(s)}$, and hence an approximation of the JLF $f(Z_k|\emptyset)$ is obtained via (23) as

$$f(Z_k|\emptyset) \approx \hat{f}^{(s)}(Z_k|\emptyset) \triangleq \exp\big(Su_J^{(s)}\big). \quad (27)$$

The choice of the weights $\gamma_{s,s'}$ determines how fast the average consensus algorithm converges. Different choices require different prior knowledge of the network topology. A popular choice is given by the Metropolis weights [29]

$$\gamma_{s,s'} \triangleq \begin{cases} \frac{1}{1+\max\{|\mathcal{S}_s|,|\mathcal{S}_{s'}|\}}, & s' \neq s, \\ 1-\sum_{r\in\mathcal{S}_s}\gamma_{s,r}, & s' = s, \end{cases} \tag{28}$$

which have been demonstrated to be well suited to various distributed tracking scenarios, e.g., [11]–[13], [30]. This choice requires that each sensor knows not only the number of its own neighbors but also the number of neighbors of each of its neighbors, i.e., $|\mathcal{S}_{s'}|$ for $s' \in \{s\} \cup \mathcal{S}_s$. This is a reasonable assumption, since each sensor is able to communicate with its neighbors. However, there exist other choices of the weights that do not require this knowledge [29]. Furthermore, the calculation of $\hat{f}^{(s)}(Z_k|\emptyset)$ according to (27) requires that each sensor knows S; this information can be provided to the sensors beforehand or can be determined in a distributed manner by another consensus algorithm [31]. Finally, the consensus iteration steps at the various sensors must be synchronized.

B. *Hypothesis* $X_k = \{\boldsymbol{x}_k\}$

For $X_k = \{\boldsymbol{x}_k\}$, Eq. (24) becomes

$$L_k(\{\boldsymbol{x}_k\}) = \frac{1}{S}\sum_{s=1}^{S}\log f\big(Z_k^{(s)}|\{\boldsymbol{x}_k\}\big). \tag{29}$$

Because the local log-likelihood functions $\log f(Z_k^{(s)}|\{\boldsymbol{x}_k\})$ now are functions of $\boldsymbol{x}_k$, expression (29) cannot be directly computed by the average consensus algorithm. A solution is provided by the LC method, which was originally proposed in the context of distributed particle filtering in [11]–[13]. The idea is to expand the local log-likelihood functions into a dictionary of functions and then use the average consensus algorithm on each expansion coefficient.

1) Likelihood Consensus: In the LC method [12], each local log-likelihood function $\log f(Z_k^{(s)}|\{\boldsymbol{x}_k\})$ is approximated by a finite-order function expansion, i.e.,

$$\log f\big(Z_k^{(s)}|\{\boldsymbol{x}_k\}\big) \approx \sum_{b=1}^{B}\alpha_{k,b}^{(s)}\psi_b(\boldsymbol{x}_k), \tag{30}$$

with a fixed dictionary of functions $\{\psi_b(\boldsymbol{x}_k)\}_{b=1}^{B}$ that is identical at all the sensors and known to each sensor. Different dictionaries can be chosen [11], [12]; a choice that is well suited to the multicomponent, continuous local log-likelihood functions arising in Bernoulli filtering is the Fourier dictionary considered in Section VIII-A. Note that the expansion coefficients $\{\alpha_{k,b}^{(s)}\}_{b=1}^{B}$ are generally different at different sensors s, as they depend on the local measurement $Z_k^{(s)}$ (cf. (30)); they can be determined locally as described in Section IV-B2.

Inserting (30) into (29) and changing the order of summations yields the following approximation of the log-JLF $L_k(\{\boldsymbol{x}_k\})$:

$$L_k(\{\boldsymbol{x}_k\}) \approx \frac{1}{S}\sum_{s=1}^{S}\sum_{b=1}^{B}\alpha_{k,b}^{(s)}\psi_b(\boldsymbol{x}_k) = \sum_{b=1}^{B}\beta_{k,b}\psi_b(\boldsymbol{x}_k), \tag{31}$$

with the global expansion coefficients

$$\beta_{k,b} \triangleq \frac{1}{S}\sum_{s=1}^{S}\alpha_{k,b}^{(s)}, \quad b = 1,\ldots,B. \tag{32}$$

Thus, the computation of $L_k(\{\boldsymbol{x}_k\})$ amounts to computing the global coefficients $\beta_{k,b}$ for $b = 1,\ldots,B$. Expression (32) can be computed by executing B parallel instances of the average consensus algorithm. Sensor s now updates B internal states $y_j^{(s,b)}$, $b = 1,\ldots,B$ in parallel, in analogy to (26); the resulting recursion is initialized as

$$y_0^{(s,b)} = \alpha_{k,b}^{(s)}. \tag{33}$$

After a sufficient number J of consensus iterations, the final internal states $y_J^{(s,b)}$ provide approximations of the global coefficients $\beta_{k,b}$ in (32), i.e., $\beta_{k,b} \approx y_J^{(s,b)}$. Hence, (31) gives

$$L_k(\{\boldsymbol{x}_k\}) \approx \sum_{b=1}^{B} y_J^{(s,b)}\psi_b(\boldsymbol{x}_k). \tag{34}$$

Inserting (34) into (23) yields the following approximation of the JLF:

$$f(Z_k|\{\boldsymbol{x}_k\}) \approx \hat{f}^{(s)}(Z_k|\{\boldsymbol{x}_k\}) \triangleq \exp\left(S\sum_{b=1}^{B} y_J^{(s,b)}\psi_b(\boldsymbol{x}_k)\right). \tag{35}$$

In the local BF at sensor s, the JLF $f(Z_k|\{\boldsymbol{x}_k\})$ is not used as a full-blown function of $\boldsymbol{x}_k$ but evaluated at the particles $\{\boldsymbol{x}_{k|k-1}^{(s,i)}\}_{i=1}^{m}$, with $m = I_{\mathrm{p}} + I_{\mathrm{b}}$ (see (22)). That is, we need the "sampled JLF" $f(Z_k|\{\boldsymbol{x}_{k|k-1}^{(s,i)}\})$, $i = 1,\ldots,m$. According to (35), an approximation of $f(Z_k|\{\boldsymbol{x}_{k|k-1}^{(s,i)}\})$ is given by

$$\hat{f}^{(s)}\big(Z_k\big|\{\boldsymbol{x}_{k|k-1}^{(s,i)}\}\big) = \exp\left(S\sum_{b=1}^{B} y_J^{(s,b)}\psi_b\big(\boldsymbol{x}_{k|k-1}^{(s,i)}\big)\right). \tag{36}$$

2) Computation of the Expansion Coefficients: At each sensor s, the local coefficient vector $\boldsymbol{\alpha}_k^{(s)} \triangleq (\alpha_{k,1}^{(s)} \cdots \alpha_{k,B}^{(s)})^{\mathrm{T}}$ involved in the local function approximation (30) can be computed *locally* by means of least squares (LS) fitting [11], [12]. More specifically, we calculate the $\boldsymbol{\alpha}_k^{(s)}$ minimizing the LS approximation error of the expansion (30) based on the particles $\{\boldsymbol{x}_{k|k-1}^{(s,i)}\}_{i=1}^{m}$, i.e., $\sum_{i=1}^{m}\epsilon_i^2$ with $\epsilon_i = \log f(Z_k^{(s)}|\{\boldsymbol{x}_{k|k-1}^{(s,i)}\}) - \sum_{b=1}^{B}\alpha_{k,b}^{(s)}\psi_b(\boldsymbol{x}_{k|k-1}^{(s,i)})$. Assuming $m \geq B$ (i.e., there are at least as many particles as expansion coefficients), the solution to this minimization problem is given by [32]

$$\boldsymbol{\alpha}_k^{(s)} = \big(\boldsymbol{\Psi}_k^{(s)\mathrm{T}}\boldsymbol{\Psi}_k^{(s)}\big)^{-1}\boldsymbol{\Psi}_k^{(s)\mathrm{T}}\boldsymbol{\lambda}_k^{(s)}, \tag{37}$$

with

$$\boldsymbol{\Psi}_k^{(s)} \triangleq \begin{pmatrix} \psi_1\big(\boldsymbol{x}_{k|k-1}^{(s,1)}\big) & \cdots & \psi_B\big(\boldsymbol{x}_{k|k-1}^{(s,1)}\big) \\ \vdots & \ddots & \vdots \\ \psi_1\big(\boldsymbol{x}_{k|k-1}^{(s,m)}\big) & \cdots & \psi_B\big(\boldsymbol{x}_{k|k-1}^{(s,m)}\big) \end{pmatrix} \tag{38}$$

and

$$\boldsymbol{\lambda}_k^{(s)} \triangleq \Big(\log f(Z_k^{(s)}|\{\boldsymbol{x}_{k|k-1}^{(s,1)}\}) \cdots \log f(Z_k^{(s)}|\{\boldsymbol{x}_{k|k-1}^{(s,m)}\})\Big)^{\mathrm{T}}. \tag{39}$$

We note that the existence of $(\boldsymbol{\Psi}_k^{(s)\mathrm{T}}\boldsymbol{\Psi}_k^{(s)})^{-1}$ in (37) requires that $\boldsymbol{\Psi}_k^{(s)}$ has full rank, i.e., the columns of $\boldsymbol{\Psi}_k^{(s)}$ are linearly independent. Numerical aspects of computing $\boldsymbol{\alpha}_k^{(s)}$ in (37) are discussed in [32], [33]. An alternative to LS fitting is provided by the orthogonal matching pursuit [13].

V. LC With Adaptive Pruning

Next, we propose a variant of the LC that uses an adaptive pruning to reduce the number of coefficients that have to be transmitted. This pruning is achieved by the inclusion of a thresholding operation in each consensus iteration. As will be shown in Section VIII, the pruning results in a significant reduction of intersensor communication without compromising the detection and tracking performance.

A. Active Index Sets and Thresholding

The jth iteration of the proposed "pruned LC" (where $j \in \{1,\ldots,J\}$) works as follows. Let us call an internal state from the previous iteration, $y_{j-1}^{(s,b)}$, *active* if $|y_{j-1}^{(s,b)}| > \eta_{j-1}^{(s)}$ with a suitably chosen positive threshold $\eta_{j-1}^{(s)}$ that depends on the sensor index s and the iteration index j. Furthermore, let us define the *active index set* $\mathcal{B}_{j-1}^{(s)} \triangleq \{b \in \tilde{\mathcal{B}}_{j-1}^{(s)} : |y_{j-1}^{(s,b)}| > \eta_{j-1}^{(s)}\}$, where $\tilde{\mathcal{B}}_{j-1}^{(s)}$ will be explained presently.

Sensor s first determines its active index set $\mathcal{B}_{j-1}^{(s)}$. Then, sensor s broadcasts its active internal states $y_{j-1}^{(s,b)}$, $b \in \mathcal{B}_{j-1}^{(s)}$ to its neighbors $s' \in \mathcal{S}_s$ and receives from them their active internal states $y_{j-1}^{(s',b)}$, $b \in \mathcal{B}_{j-1}^{(s')}$. Next, suitably modifying the standard consensus update in (26), sensor s calculates the linear combinations

$$y_j^{(s,b)} = \sum_{s' \in \{s\}\cup\mathcal{S}_s} \gamma_{s,s'} y_{j-1}^{(s',b)}, \quad b \in \tilde{\mathcal{B}}_j^{(s)}, \tag{40}$$

where $\tilde{\mathcal{B}}_j^{(s)}$ is the union of all the involved active index sets, i.e., $\tilde{\mathcal{B}}_j^{(s)} \triangleq \bigcup_{s'\in\{s\}\cup\mathcal{S}_s} \mathcal{B}_{j-1}^{(s')}$. Note that in (40), all nonactive internal states, $y_{j-1}^{(s',b)}$ for $b \notin \mathcal{B}_{j-1}^{(s')}$ and $s' \in \{s\}\cup\mathcal{S}_s$, are treated as zero. In the next ($(j+1)$th) iteration, $\tilde{\mathcal{B}}_j^{(s)}$ is used to determine the new active index set $\mathcal{B}_j^{(s)} \triangleq \{b \in \tilde{\mathcal{B}}_j^{(s)} : |y_j^{(s,b)}| > \eta_j^{(s)}\}$, etc. This recursion is initialized in iteration $j=1$ by setting $y_0^{(s,b)} = \alpha_{k,b}^{(s)}$ for $b = 1,\ldots,B$ and $\tilde{\mathcal{B}}_0^{(s)} = \{1,\ldots,B\}$.

An adaptive choice of the pruning thresholds $\eta_j^{(s)}$ that has been observed to yield good performance is to set $\eta_j^{(s)} = \omega \cdot \frac{1}{B}\sum_{b=1}^{B} |y_j^{(s,b)}|$ with a fixed factor $\omega > 0$. Note that this adaptive choice of $\eta_j^{(s)}$ is performed locally at sensor s and does not require any additional intersensor communication. Our experiments also suggest that to avoid divergence of the consensus algorithm, pruning should set in only after a certain number of initial consensus iterations, i.e., the pruning threshold should be chosen initially as $\eta_j^{(s)} = 0$. This number of initial consensus iterations, as well as the factor ω, are application-dependent and chosen empirically.

B. Algorithm Summary and Communication Cost

At each time k, each sensor s performs J consensus iterations for a distributed computation of $\hat{f}^{(s)}(Z_k|\emptyset)$ (see (27)) and B parallel instances of J consensus iterations for a distributed computation of $\hat{f}^{(s)}(Z_k|\{\boldsymbol{x}_{k|k-1}^{(s,i)}\})$, $i = 1,\ldots,m$ (see (36)). The overall consensus method for a distributed computation of the JLF is summarized in Algorithm 1. This algorithm includes the pruned LC proposed in the previous subsection.

In consensus iteration j of all the $1+B$ parallel consensus algorithms, sensor s broadcasts to its neighbors the internal states $u_{j-1}^{(s)}$ and $y_{j-1}^{(s,b)}$ for $b \in \mathcal{B}_{j-1}^{(s)}$, as well as a B-dimensional binary vector $\boldsymbol{a} = (a_1 \cdots a_B) \in \{0,1\}^B$ that encodes the active index set $\mathcal{B}_{j-1}^{(s)} \subseteq \{1,\ldots,B\}$, i.e., $a_b = 1$ if $b \in \mathcal{B}_{j-1}^{(s)}$ and $a_b = 0$ otherwise. If each internal state is represented by R bits, this amounts to $R(1+|\mathcal{B}_{j-1}^{(s)}|) + B$ bits. Thus, the communication cost of the proposed BF consensus method at sensor s, defined as the number of bits broadcast at each time k by sensor s to its neighbors during all the J consensus iterations, is

$$\begin{aligned} N_{\mathrm{c}}^{(s)} &= \sum_{j=1}^{J} \big(R\big(1+|\mathcal{B}_{j-1}^{(s)}|\big) + B\big) \\ &= J(R+B) + R\sum_{j=1}^{J} |\mathcal{B}_{j-1}^{(s)}|. \end{aligned}$$

VI. Distributed Bernoulli Filtering

The proposed distributed BF combines the particle-based implementation of the centralized BF reviewed in Section III-C with the consensus-based distributed computation of the JLF presented in Sections IV and V. Each sensor s runs a local BF that is based on the JLF $f(Z_k|X_k)$. The local BF at sensor s propagates at each time k a local estimate $\hat{q}_{k|k}^{(s)}$ of the target presence probability $q_{k|k}$ and local particle representations $\{(\boldsymbol{x}_k^{(s,i)}, w_k^{(s,i)})\}_{i=1}^{I_{\mathrm{p}}}$ of the posterior state vector pdf $f(\boldsymbol{x}_k|Z_{1:k})$ (see Section III-C2). Note that the local particle sets of the individual sensors are updated separately but not altogether independently since the update at sensor s involves the (approximate) JLF of all the sensors and not merely the local likelihood function of sensor s. Thereby, each local BF takes into account the measurements of *all* the sensors and, thus, achieves a performance similar to that of a centralized BF that has direct access to all the measurements.

The operations of the local BF executed by sensor s are stated in Algorithm 2. These operations are similar to those of the particle-based centralized BF reviewed in Section III-C. The key difference is the consensus-based distributed computation of the JLF (using Algorithm 1) in Step 10 of Algorithm 2, which

Algorithm 1: Consensus Method for Distributed JLF Computation—Operations Performed by Sensor s at Time $k \in \{1, 2, \ldots\}$.

```
Input: $\{\boldsymbol{x}_{k|k-1}^{(s,i)}\}_{i=1}^{m}$ and $Z_k^{(s)}$;
Compute $\log f(Z_k^{(s)}|\emptyset)$ according to (3) and set $u_0^{(s)} = \log f(Z_k^{(s)}|\emptyset)$;
Compute $\boldsymbol{\Psi}_k^{(s)}$ and $\boldsymbol{\lambda}_k^{(s)}$ using (38) and (39), respectively;
Compute $\boldsymbol{\alpha}_k^{(s)}$ using (37), and set $y_0^{(s,b)} = \alpha_{k,b}^{(s)}$ for $b = 1, \ldots, B$ and $\tilde{\mathcal{B}}_0^{(s)} = \{1, \ldots, B\}$;
for $j = 1$ to $J$ do
    Determine the active index set $\mathcal{B}_{j-1}^{(s)} = \big\{b \in \tilde{\mathcal{B}}_{j-1}^{(s)} : |y_{j-1}^{(s,b)}| > \eta_{j-1}^{(s)}\big\}$;
    Broadcast $u_{j-1}^{(s)}$ as well as the active internal states $y_{j-1}^{(s,b)}$, $b \in \mathcal{B}_{j-1}^{(s)}$ and their indices $b \in \mathcal{B}_{j-1}^{(s)}$ to the neighbor sensors $s' \in \mathcal{S}_s$;
    Receive $u_{j-1}^{(s')}$ as well as the active internal states $y_{j-1}^{(s',b)}$, $b \in \mathcal{B}_{j-1}^{(s')}$ and their indices $b \in \mathcal{B}_{j-1}^{(s')}$ from the neighbor sensors $s' \in \mathcal{S}_s$;
    Compute $u_j^{(s)} = \sum_{s' \in \{s\} \cup \mathcal{S}_s} \gamma_{s,s'} u_{j-1}^{(s')}$;
    Form $\tilde{\mathcal{B}}_j^{(s)} = \bigcup_{s' \in \{s\} \cup \mathcal{S}_s} \mathcal{B}_{j-1}^{(s')}$;
    for $b \in \tilde{\mathcal{B}}_j^{(s)}$ do
        Compute $y_j^{(s,b)} = \sum_{s' \in \{s\} \cup \mathcal{S}_s} \gamma_{s,s'} y_{j-1}^{(s',b)}$, where nonactive $y_{j-1}^{(s',b)}$ are considered to be zero;
    end for
end for
Compute an approximation of $f(Z_k|\emptyset)$ according to $\hat{f}^{(s)}(Z_k|\emptyset) = \exp\big(S u_J^{(s)}\big)$;
for $i = 1$ to $m$ do
    Compute an approximation of $f\big(Z_k\big|\{\boldsymbol{x}_{k|k-1}^{(s,i)}\}\big)$ according to $\hat{f}^{(s)}\big(Z_k\big|\{\boldsymbol{x}_{k|k-1}^{(s,i)}\}\big) = \exp\Big(S \sum_{b=1}^{B} y_J^{(s,b)} \psi_b\big(\boldsymbol{x}_{k|k-1}^{(s,i)}\big)\Big)$, where missing $y_J^{(s,b)}$ are considered to be zero;
end for
Output: $\hat{f}^{(s)}(Z_k|\emptyset)$ and $\hat{f}^{(s)}\big(Z_k\big|\{\boldsymbol{x}_{k|k-1}^{(s,i)}\}\big)$, $i = 1, \ldots, m$.
```

Algorithm 2: Distributed Bernoulli Filter—Operations Performed by Sensor s at Time $k \in \{1, 2, \ldots\}$.

```
Input: $\hat{q}_{k-1|k-1}^{(s)}$, $\big\{\big(\boldsymbol{x}_{k-1}^{(s,i)}, w_{k-1}^{(s,i)}\big)\big\}_{i=1}^{I_\mathrm{p}}$, and $Z_k^{(s)}$
(Initialization at $k = 1$: $\hat{q}_{0|0}^{(s)}$ is set to the initial prior target presence probability $q_0$, the particles $\{\boldsymbol{x}_0^{(s,i)}\}_{i=1}^{I_\mathrm{p}}$ are drawn from the initial prior pdf $f(\boldsymbol{x}_0)$, and the weights are chosen all equal, i.e., $w_0^{(s,i)} = 1/I_\mathrm{p}$)
Compute $\hat{q}_{k|k-1}^{(s)}$ using (17), i.e.,
    $$\hat{q}_{k|k-1}^{(s)} = P_\mathrm{s} \hat{q}_{k-1|k-1}^{(s)} + P_\mathrm{b}\big(1 - \hat{q}_{k-1|k-1}^{(s)}\big);$$
for $i = 1$ to $I_\mathrm{p}$ do
    For each particle $\boldsymbol{x}_{k-1}^{(s,i)}$, draw a predicted particle $\boldsymbol{x}_{k|k-1}^{(s,i)}$ from the state vector transition pdf $f\big(\boldsymbol{x}_k\big|\boldsymbol{x}_{k-1}^{(s,i)}\big)$;
    Compute a corresponding weight using (18), i.e.,
        $$w_{k|k-1}^{(s,i)} = \frac{P_\mathrm{s} \hat{q}_{k-1|k-1}^{(s)}}{\hat{q}_{k|k-1}^{(s)}} w_{k-1}^{(s,i)};$$
end for
for $i = I_\mathrm{p} + 1$ to $I_\mathrm{p} + I_\mathrm{b}$ do
    Draw a birth particle $\boldsymbol{x}_{k|k-1}^{(s,i)}$ from the birth pdf $f_\mathrm{b}(\boldsymbol{x}_k)$;
    Compute a corresponding weight according to (19), i.e.,
        $$w_{k|k-1}^{(s,i)} = \frac{P_\mathrm{b}\big(1 - \hat{q}_{k-1|k-1}^{(s)}\big)}{\hat{q}_{k|k-1}^{(s)}} \frac{1}{I_\mathrm{b}};$$
end for
Execute Algorithm 1 with $m = I_\mathrm{p} + I_\mathrm{b}$ to obtain $\hat{f}^{(s)}(Z_k|\emptyset)$ and $\hat{f}^{(s)}\big(Z_k\big|\{\boldsymbol{x}_{k|k-1}^{(s,i)}\}\big)$, $i = 1, \ldots, I_\mathrm{p} + I_\mathrm{b}$;
Compute $\hat{q}_{k|k}^{(s)}$ using (20), i.e.,
    $$\hat{q}_{k|k}^{(s)} = 1 - \frac{\hat{f}^{(s)}(Z_k|\emptyset)(1 - \hat{q}_{k|k-1}^{(s)})}{\hat{f}^{(s)}(Z_k|Z_{1:k-1})},$$
    where (cf. (21))
    $$\hat{f}^{(s)}(Z_k|Z_{1:k-1}) = \hat{C}^{(s)}(Z_{1:k}) \hat{q}_{k|k-1}^{(s)} + \hat{f}^{(s)}(Z_k|\emptyset) \times (1 - \hat{q}_{k|k-1}^{(s)})$$
    with $\hat{C}^{(s)}(Z_{1:k}) = \sum_{i=1}^{I_\mathrm{p}+I_\mathrm{b}} \hat{f}^{(s)}\big(Z_k\big|\{\boldsymbol{x}_{k|k-1}^{(s,i)}\}\big) w_{k|k-1}^{(s,i)}$;
for $i = 1$ to $I_\mathrm{p} + I_\mathrm{b}$ do
    Compute a nonnormalized weight $\tilde{w}_k^{(s,i)}$ using (22), i.e.,
        $$\tilde{w}_k^{(s,i)} = \hat{f}^{(s)}\big(Z_k\big|\{\boldsymbol{x}_{k|k-1}^{(s,i)}\}\big) w_{k|k-1}^{(s,i)};$$
end for
```

(continued on next page)

provides $\hat{f}^{(s)}(Z_k|\emptyset)$ and $\hat{f}^{(s)}(Z_k|\{\boldsymbol{x}_{k|k-1}^{(s,i)}\})$, $i = 1, \ldots, I_\mathrm{p} + I_\mathrm{b}$ to sensor s. These approximate quantities are then used instead of the exact quantities $f(Z_k|\emptyset)$ and $f(Z_k|\{\boldsymbol{x}_{k|k-1}^{(s,i)}\})$, $i = 1, \ldots, I_\mathrm{p} + I_\mathrm{b}$ in the subsequent steps of the filter.

Because each local BF uses its own set of local particles, the random number generators of different sensors (which are used to draw the local particles from the local proposal pdfs $f(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}^{(s,i)})$ and $f_\mathrm{b}(\boldsymbol{x}_k)$) do not have to be synchronized. This is different from the "consensus-over-weights" based distributed particle filtering methods proposed in [34], [35], in which all the sensors have to use exactly the same set of particles and, hence, the random number generators of different sensors have to be

Algorithm 2 (continued)

```
Retain the $I_\mathrm{p}$ particles $\boldsymbol{x}_{k|k-1}^{(s,i)}$ with the highest weights
    $\tilde{w}_k^{(s,i)}$ and normalize these weights to obtain (after
    re-indexing) a set of particles and normalized weights
    $\{(\boldsymbol{x}_k^{(s,i)}, w_k^{(s,i)})\}_{i=1}^{I_\mathrm{p}}$ with $\sum_{i=1}^{I_\mathrm{p}} w_k^{(s,i)} = 1$;
Perform resampling if the effective sample size [28] is
    below a given threshold.
Output: $\hat{q}_{k|k}^{(s)}$ and $\{(\boldsymbol{x}_k^{(s,i)}, w_k^{(s,i)})\}_{i=1}^{I_\mathrm{p}}$
```

synchronized. Note that the methods in [34], [35] are suitable only for scenarios where the target is always present.

Another advantage of the proposed LC-based method relative to the consensus-over-weights approach is that the communication requirements are significantly reduced: In the consensus-over-weights approach, $m = I_\mathrm{p} + I_\mathrm{b}$ average consensus algorithms would have to be run in parallel (note that I_b has to be equal at all sensors), which means that at each consensus iteration, each sensor would need to broadcast m real values. By contrast, in the proposed approach, each sensor broadcasts only its active internal states. Typically, m is on the order of several thousands whereas the number of internal states is significantly smaller (see Section VIII).

VII. Group LC

We next propose a variant of the pruned LC presented in Section V, referred to as *group LC* (GLC), which can lead to a further substantial reduction of intersensor communication. We consider sensor networks that comprise a certain number of star-connected *sensor groups* $\mathcal{G}_c \subseteq \{1, \ldots, S\}$ as depicted schematically in Fig. 1. Each sensor group $\mathcal{G}_c$ consists of a central *consensus node* $c \in \{1, \ldots, S\}$ and an arbitrary number of *auxiliary nodes* that are directly connected to the consensus node. If a consensus node c is not connected to an auxiliary node, then $\mathcal{G}_c = \{c\}$. We denote by $\mathcal{C} \subseteq \{1, \ldots, S\}$ the set of consensus nodes c. Each consensus node $c \in \mathcal{C}$ is able to communicate with a subset of the other consensus nodes, $\mathcal{N}_c \subseteq \mathcal{C} \backslash \{c\}$; the consensus nodes $c' \in \mathcal{N}_c$ will be referred to as the *neighbors* of consensus node c. The neighbor sets $\mathcal{N}_c$ of all the consensus nodes $c \in \mathcal{C}$ define the *consensus network*. We make the following further assumptions:

1) Each node is a consensus node or an auxiliary node.
2) Each auxiliary node is connected to exactly one consensus node and no other auxiliary node.
3) The graph of the consensus network is connected.

Under these assumptions, the sensor groups $\mathcal{G}_c$ form a partition of the entire sensor network, i.e., $\bigcup_{c\in\mathcal{C}} \mathcal{G}_c = \{1, \ldots, S\}$ and $\mathcal{G}_c \cap \mathcal{G}_{c'} = \emptyset$ if $c \neq c'$. We note that any connected network can always be split into consensus nodes and auxiliary nodes in the sense of the above definition: each consensus node is adjacent to an arbitrary number of other nodes (typically at least two), whereas each auxiliary node is adjacent to a single node. However, the reduction in communication cost achieved by the GLC is greatest when there are few consensus nodes and many auxiliary nodes (unless the dimension of the measurements is very high). We assume that the network configuration relevant to the GLC is known to all the sensor nodes at the start of the filtering operation. The development of an efficient distributed method for establishing the GLC network configuration—i.e., identifying the consensus nodes and auxiliary nodes and, possibly, removing certain edges in order to convert some of the consensus nodes into auxiliary nodes—and for disseminating the relevant information throughout the network is beyond the scope of this paper.

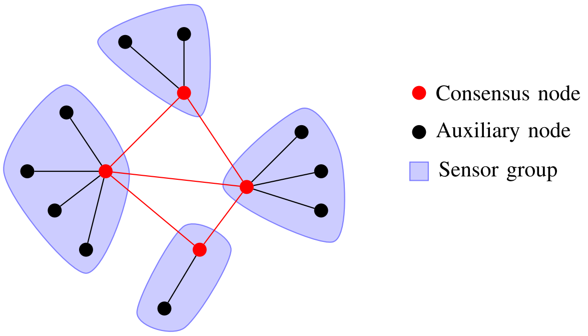


Fig. 1. Hierarchical sensor network with star-connected sensor groups.

Let $Z_k^{(\mathcal{G}_c)} \triangleq (Z_k^{(s)})_{s\in\mathcal{G}_c}$ denote the ordered sequence of the measurement sets of all the sensors in sensor group $\mathcal{G}_c$. The factorization of the JLF in (6) can now be rewritten in a hierarchical way as

$$f(Z_k|X_k) = \prod_{c\in\mathcal{C}} f\big(Z_k^{(\mathcal{G}_c)}\big|X_k\big),$$

with

$$f\big(Z_k^{(\mathcal{G}_c)}\big|X_k\big) = \prod_{s\in\mathcal{G}_c} f\big(Z_k^{(s)}\big|X_k\big). \tag{41}$$

Note that $f(Z_k^{(\mathcal{G}_c)}|X_k)$ is the JLF of sensor group $\mathcal{G}_c$. The proposed GLC is based on this hierarchical JLF factorization and performs the following steps at time k. First, the auxiliary nodes $s \in \mathcal{G}_c$ of each sensor group $\mathcal{G}_c$ transmit their measurements $Z_k^{(s)}$ to the associated consensus node c. The consensus node then has knowledge of $Z_k^{(\mathcal{G}_c)}$, and it calculates the JLF of its sensor group $\mathcal{G}_c$, $f(Z_k^{(\mathcal{G}_c)}|X_k)$, according to (41). Next, the consensus nodes execute the pruned LC (Algorithm 1) *within the consensus network*, with each consensus node c using its group JLF $f(Z_k^{(\mathcal{G}_c)}|X_k)$ instead of just its own local likelihood function $f(Z_k^{(c)}|X_k)$. Let $u_J^{(c)}$ and $y_J^{(c,b)}$, $b = 1, \ldots, B$ denote the internal states obtained at consensus node c after the final (Jth) consensus iteration. Consensus node c forwards these final internal states to its auxiliary nodes. Thus, the internal states of all the nodes s in sensor group $\mathcal{G}_c$—both of the consensus node and of the auxiliary nodes—are given by $u_J^{(s)} = u_J^{(c)}$ and $y_J^{(s,b)} = y_J^{(c,b)}$, $b = 1, \ldots, B$, for $s \in \mathcal{G}_c$. Finally, each node uses its internal states to calculate an approximation to the JLF according to lines 14–17 of Algorithm 1.

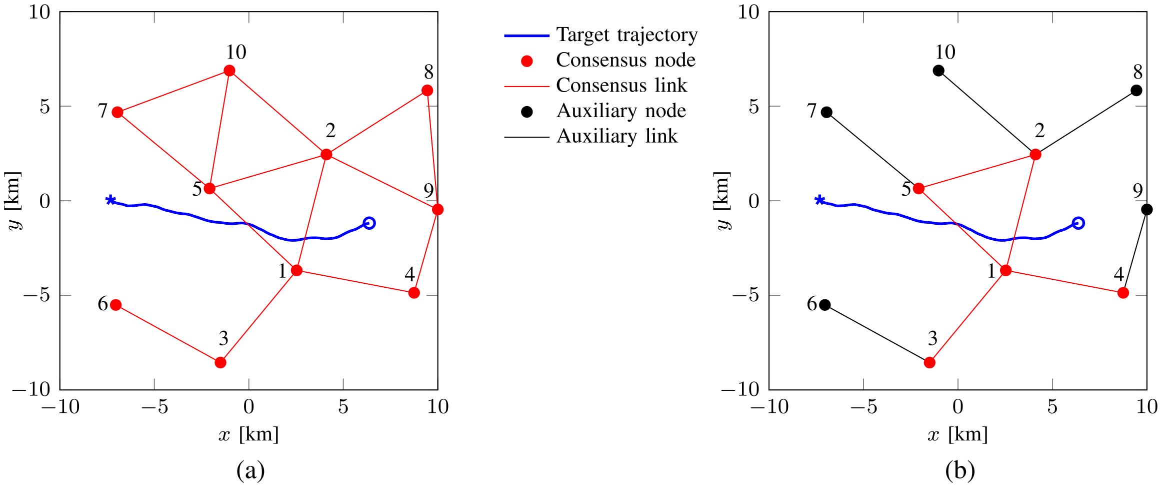


Fig. 2. Surveillance region with the sensor network and the target trajectory. The starting point of the trajectory is indicated by a star. (a) Network topology used for simulating the LC. (b) Modified network topology used for simulating the GLC.

VIII. Simulation Results

In this section, we demonstrate the performance of the proposed distributed BF in two different simulation scenarios. The first scenario consists of a homogeneous sensor network, where each sensor measures its distance (range) and its angle (bearing) relative to the target and all the sensors have identical characteristics. In the second scenario, we consider a heterogeneous sensor network where each sensor acquires either range or bearing measurements with different sensor characteristics. We compare the results of our proposed distributed BF with those of the optimal centralized BF and of the random exchange distributed BF presented in [8], which was adapted to incorporate our measurement model. A second state-of-the-art distributed BF, presented in [6], was found to be unsuitable for our simulation scenarios since its inherent linearization of the measurement model led to excessive errors and a divergent behavior. Before discussing the two scenarios individually, we describe common simulation parameters.

A. Common Simulation Parameters

Fig. 2(a) shows the 2-D surveillance region $[-10\,\text{km}, 10\,\text{km}] \times [-10\,\text{km}, 10\,\text{km}]$, the sensor positions, the intersensor communication links, and the target trajectory used in our simulations. A modified network topology is used for our simulations demonstrating the performance of the GLC: here, as depicted in Fig. 2(b), four of the communication links are removed in order to better conform to the hierarchical topology considered in Section VII.

We consider time scans up to $k = 120$. The target appears at time $k = 10$, traverses the surveillance region from left to right, and disappears at time $k = 111$. The 4-D state vector is given by $\boldsymbol{x}_k \triangleq (x_k\ y_k\ \dot{x}_k\ \dot{y}_k)^{\mathrm{T}}$, where x_k and y_k are the target positions and $\dot{x}_k$ and $\dot{y}_k$ are the target velocities in the two coordinate directions. When the target is present, the state vector evolves according to the near constant velocity model (cf. (2)) $\boldsymbol{x}_k = \boldsymbol{F}\boldsymbol{x}_{k-1} + \boldsymbol{A}\boldsymbol{v}_k$. Here, $\boldsymbol{F} \in \mathbb{R}^{4\times 4}$ and $\boldsymbol{A} \in \mathbb{R}^{4\times 2}$ are chosen as in [36, Sec. 6.3.2] (their definition involves a time scan duration T), and the 2-D driving noise vector $\boldsymbol{v}_k$ is modeled as an i.i.d. random process with Gaussian pdf $\mathcal{N}(\boldsymbol{v}_k; \boldsymbol{0}, \sigma_v^2 \mathbf{I}_2)$. Note that $f(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}) = \mathcal{N}(\boldsymbol{x}_k; \boldsymbol{F}\boldsymbol{x}_{k-1}, \sigma_v^2 \boldsymbol{A}\boldsymbol{A}^{\mathrm{T}})$. The initial prior target presence probability is $q_0 = 0.5$, i.e., the filters are completely unaware if the target is present or not. Furthermore, initially, the filters do not have any information about the position of the target in the surveillance region and the direction in which it is moving, but the distribution of the initial target speed is known. Accordingly, the initial prior vector pdf $f(\boldsymbol{x}_0)$ is defined implicitly by the following sampling procedure:

1) The position (x_0, y_0) is drawn uniformly from the surveillance region.
2) A nonnegative speed value v_0 is drawn from a truncated Gaussian distribution with mean 2 and standard deviation $1/3$.
3) A heading angle θ_0 is drawn uniformly from the interval $[-180°, 180°)$.
4) The velocity components are calculated as $\dot{x}_0 = v_0 \cos\theta_0$ and $\dot{y}_0 = v_0 \sin\theta_0$.

The target birth pdf $f_{\mathrm{b}}(\boldsymbol{x}_k)$ is chosen identical to the initial prior vector pdf $f(\boldsymbol{x}_0)$.

The proposed distributed BF uses for the (G)LC a 2-D Fourier dictionary with B atoms given by [12] $\psi_b(\boldsymbol{x}_k) = \phi_{\tilde{b}_1}(x_k)\phi_{\tilde{b}_2}(y_k)$, with an index transformation that maps $b \in \{1, \ldots, B\}$ to $(\tilde{b}_1, \tilde{b}_2) \in \{1, \ldots, 2\tilde{B}+1\}^2$. Here, $\tilde{B}$ is a nonnegative integer parameter that determines the size of the dictionary, which is $B = (2\tilde{B}+1)^2$. The 1-D atoms $\phi_{\tilde{b}}(x)$ are given by

$$\phi_{\tilde{b}}(x) = \begin{cases} 1, & \tilde{b} = 1 \\ \cos\!\left(\frac{2\pi}{d_a}(\tilde{b}-1)x\right), & \tilde{b} = 2, \ldots, \tilde{B}+1 \\ \sin\left(\frac{2\pi}{d_a}(\tilde{b}-1-\tilde{B})x\right), & \tilde{b} = \tilde{B}+2, \ldots, 2\tilde{B}+1, \end{cases}$$

where $d_a = 20$ km is the width of the surveillance area in each coordinate direction. Note that we use a 2-D, rather than 4-D, dictionary because the sensors produce range and/or bearing measurements and thus the local likelihood functions depend only on the x_k and y_k components of the state $\boldsymbol{x}_k$. Furthermore,

TABLE I
COMMON PARAMETER VALUES

Parameter	Value	Description
d	4	Dimension of state vector $\boldsymbol{x}_k$
S	10	Number of sensors
T	40 s	Time scan duration
P_{b}	0.1	Birth probability
P_{s}	0.999	Survival probability
σ_v	$0.01\,\mathrm{m/s}^2$	Driving noise standard deviation
I_{p}	1000	Number of predicted particles
I_{b}	5000	Number of birth particles
J	25	Number of consensus iterations

the (G)LC uses Metropolis weights $\gamma_{s,s'}$ given by (28) and performs $J = 25$ consensus iterations. Adaptive pruning sets in after 15 consensus iterations, i.e., for $j = 1, \ldots, 15$ the pruning thresholds are set to $\eta_j^{(s)} = 0$, and for $j = 16, \ldots, 25$ they are calculated adaptively as $\eta_j^{(s)} = \omega \cdot \frac{1}{B} \sum_{b=1}^{B} |y_j^{(s,b)}|$, where $\omega = 0.1$ was determined empirically. Communication between sensors is performed in a binary format. More specifically, the real values (i.e., the internal states and, in the case of the GLC, also the measurements) to be broadcast are encoded as 32-bit floating point numbers. In addition, if adaptive pruning is used, the set of active indices $b \in \{1, \ldots, B\}$ is specified by a binary word of length B, where the bth bit is 1 if index b is active and 0 otherwise. The numerical values of the common simulation parameters are listed in Table I.

For comparison, we also considered a variant of the random exchange distributed BF proposed in [8] that was adapted to our range-bearing measurement model by using the appropriate type of local likelihood function. Our implementation of the random exchange distributed BF uses a single Gaussian for approximating the posterior distributions; this conforms to the recommendation given in [8]. The other parameters are as given in Table I (except for J, which is not applicable).

B. Homogeneous Sensor Network

In this simulation scenario, all sensors have equal characteristics and acquire range and bearing measurements. Hence, equation (5) becomes

$$\boldsymbol{z}_k^{(s)} = \begin{pmatrix} z_{k,1}^{(s)} \\ z_{k,2}^{(s)} \end{pmatrix} = \begin{pmatrix} \sqrt{(x_k - x^{(s)})^2 + (y_k - y^{(s)})^2} \\ \arctan\left(\frac{y_k - y^{(s)}}{x_k - x^{(s)}}\right) \end{pmatrix} + \boldsymbol{w}_k^{(s)}, \tag{42}$$

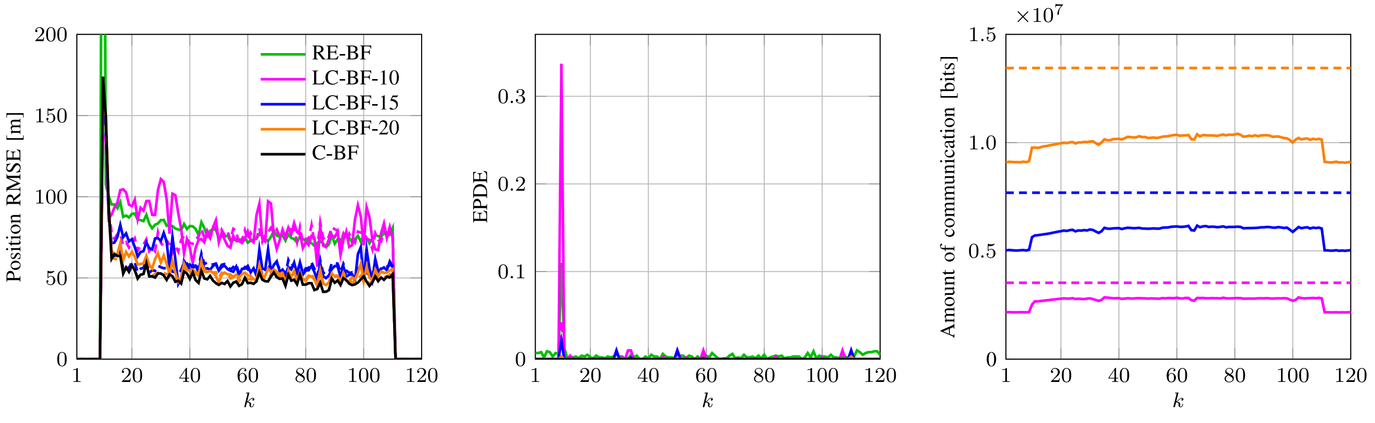
where x_k and y_k are the target coordinates at time k, $x^{(s)}$ and $y^{(s)}$ are the coordinates of sensor s, and $\boldsymbol{w}_k^{(s)}$ is an i.i.d. zero-mean Gaussian 2-D random process with covariance matrix $\boldsymbol{C}_w = \mathrm{diag}\{\sigma_{\mathrm{r}}^2, \sigma_{\mathrm{b}}^2\}$. Here, the range and bearing standard deviations are chosen as $\sigma_{\mathrm{r}} = 150\,\mathrm{m}$ and $\sigma_{\mathrm{b}} = 1^\circ$, respectively. At each sensor s, when the target is present, target-originated measurements $\boldsymbol{z}_k^{(s)}$ are randomly generated with a detection probability of $P_{\mathrm{d}}^{(s)}(\boldsymbol{x}_k) = 0.9$ according to (42). The clutter measurements are uniformly distributed with pdf (cf. (3)) $f_{\mathrm{c}}^{(s)}(\boldsymbol{z}) = 1/(360^\circ R)$ for $z_1^{(s)} \leq R$ and $f_{\mathrm{c}}^{(s)}(\boldsymbol{z}) = 0$ for $z_1^{(s)} > R$, where $R = 30\,\mathrm{km}$ is the maximum range of the sensors. The pmf of the number of clutter measurements, $p_{\mathrm{c}}^{(s)}(n)$ in (3), is Poisson with mean $\mu^{(s)} = 5$.

Fig. 3 shows three averaged performance metrics for the proposed LC-based distributed BF with dictionary size parameter $\tilde{B} \in \{10, 15, 20\}$, abbreviated LC-BF-10 etc. (Hence, the size of the 2-D dictionary is $B = (2\tilde{B} + 1)^2 \in \{441, 961, 1681\}$.) Fig. 4 shows these metrics for the distributed BF using the GLC, abbreviated GLC-BF-10 etc. For each dictionary size, we show the results obtained with and without adaptive pruning. For comparison, we also show the results of the centralized BF (abbreviated C-BF) and of the random exchange BF (abbreviated RE-BF). All results were averaged over 100 simulation runs, and those for the distributed BFs also over the ten sensors.

In particular, Fig. 3(a) and Fig. 4(a) show the root-mean-square error (RMSE) of the position estimates. For times where the target is present, i.e., $k = 10, \ldots, 110$, the RMSE was calculated at each individual time k by averaging over all simulation runs with successful target detections. For times where the target is absent, the RMSE was set to zero. One can see in Fig. 3(a) that after the initial detection of the target at time $k = 10$, the RMSEs of all filters decrease to an error floor. As expected, larger dictionary sizes lead to smaller RMSE values. For k between about 10 and 40, the RMSE of LC-BF-10 without pruning is significantly smaller than that of RE-BF. For the remaining k, the RMSE of LC-BF-10 with or without pruning is generally similar to that of RE-BF and about 30 m larger than that of C-BF. However, at time $k = 10$, it is much smaller than that of RE-BF. The RMSEs of LC-BF-15 and LC-BF-20 are considerably smaller than those of LC-BF-10 and RE-BF and, in fact, very close to that of C-BF. The adaptive pruning is generally seen to produce effectively no loss in RMSE performance; an exception is observed in the case of LC-BF-10 for k between about 10 and 40, where pruning increases the RMSE. Finally, a comparison with Fig. 4(a) shows that GLC-BF has effectively the same position RMSE as LC-BF. Thus, the significant reduction of communication obtained with GLC-BF—shown in Fig. 4(c)—does not result in a reduced accuracy of position estimation.

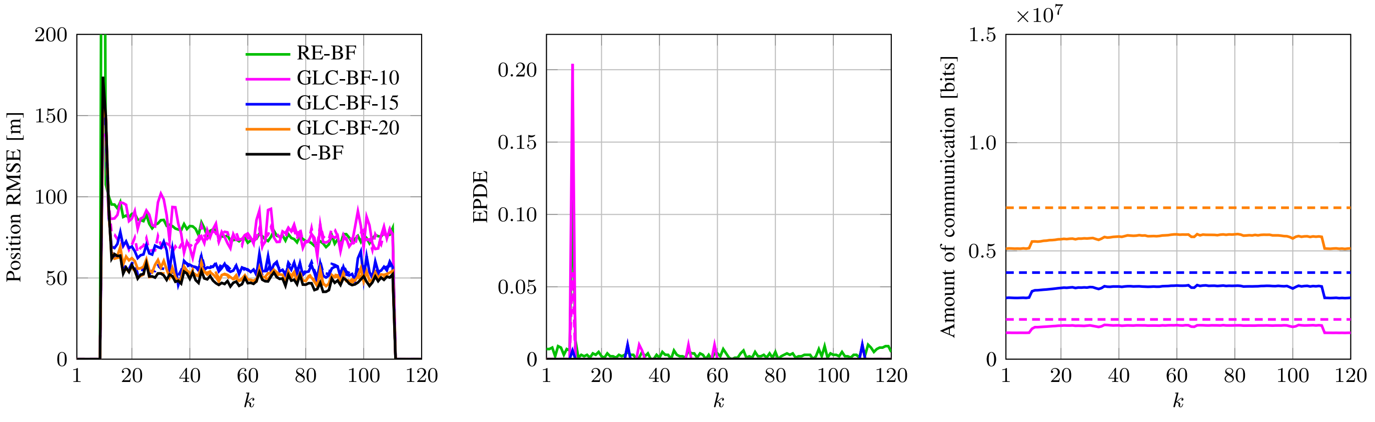
Fig. 3(b) and Fig. 4(b) show the estimated probability of a detection error (EPDE), which is defined as the average of an indicator variable that is one if the tracking method did not detect the target even though it was present or it detected the target even though it was absent, and zero otherwise. One can see in Fig. 3(b) that the EPDEs of all filters are effectively zero at almost all times. For all filters, a peak occurs at time $k = 10$, i.e., when the target is born. This is because the target has been observed only at this single time instant (if at all), which makes a decision about its presence unreliable. However, after just one time step, the EPDE drops to zero or almost zero, which means that the target is almost always detected correctly. One can also see that adaptive pruning effectively does not increase the EPDE. Furthermore, a comparison with Fig. 4(b) shows that the EPDE of GLC-BF is effectively equal to that of LC-BF.

Fig. 3(c) and Fig. 4(c) show the amount of communication required by the proposed distributed BF. More specifically, these



(a) (b) (c)

Fig. 3. Average performance metrics of LC-BF-10, LC-BF-15, LC-BF-20, RE-BF, and C-BF versus time in the homogeneous scenario. The results of LC-BF-10, LC-BF-15, and LC-BF-20 with and without adaptive pruning are depicted by solid and dashed lines, respectively. (a) Position RMSE, (b) EPDE, (c) amount of binary data broadcast by all the ten sensors during one filtering step.



(a) (b) (c)

Fig. 4. Same as Fig. 3, but showing the results of GLC-BF rather than LC-BF.

figures depict the total amount of binary data transmitted by all the ten sensors during one filtering step, i.e., in the course of all the $J = 25$ consensus iterations, using the binary representation described in Section VIII-A. Both with and without adaptive pruning, a larger dictionary leads to a higher amount of communication but also, as was shown by Fig. 3(a), (b) and Fig. 4(a), (b), to better detection/tracking performance. Thus, as expected, there is a tradeoff between communication cost and detection/tracking performance. One can see in Fig. 3(c) that with adaptive pruning, the communication cost of LC-BF can be reduced by about 10–25%. This reduction is larger—both absolutely and relatively—for a larger dictionary size. Note that it is not paid for by a noticeable reduction of the detection/tracking performance (with the exception of LC-BF-10 for k between 10 and 40).

As can be seen in Fig. 4(c), GLC-BF requires only about half the amount of communication required by LC-BF. This reduction in communication is not paid for by any loss in detection/tracking performance. For RE-BF, encoding the measurements and Gaussian parameters to be transmitted as 32-bit floating point numbers, the amount of communication is obtained as about 12 kbit per filtering step.[1] This is much less than the amount of communication required by the proposed distributed BF. In particular, GLC-BF-10 with adaptive pruning (which has the lowest communication cost among all the considered variants of the proposed distributed BF) transmits about 1.5 Mbit in total with about the same detection and tracking performance as RE-BF. For (G)LC-BF-15 and (G)LC-BF-20,

[1] This value is obtained by noting that for RE-BF, the average number of bits transmitted by all the sensors in the network in one time step is given by $32 \cdot 10 \cdot (N_{\mathrm{G}} + N_{\mathrm{m}})$, where the factor 32 is the binary wordlength, the factor 10 is the number of sensors in the network, N_{G} is the total number of real values representing the Gaussian parameters for one sensor, and N_{m} is the average total number of real values representing the measurements for one sensor. Here, $N_{\mathrm{G}} = 2 \cdot 14 = 28$, where the factor 2 arises since each exchange between two sensors comprises two transmissions and the factor 14 is the number of real values required to represent the parameters of a 4-D Gaussian (4 for the mean and 10 for the covariance matrix). Furthermore, in the homogeneous sensor network scenario considered, $N_{\mathrm{m}} = 2 \cdot n_{\mathrm{m}}$, where the factor 2 arises since each measurement consists of two real values (range and bearing) and the factor n_{m} is the average number of measurements per sensor (both target-originated and due to clutter).

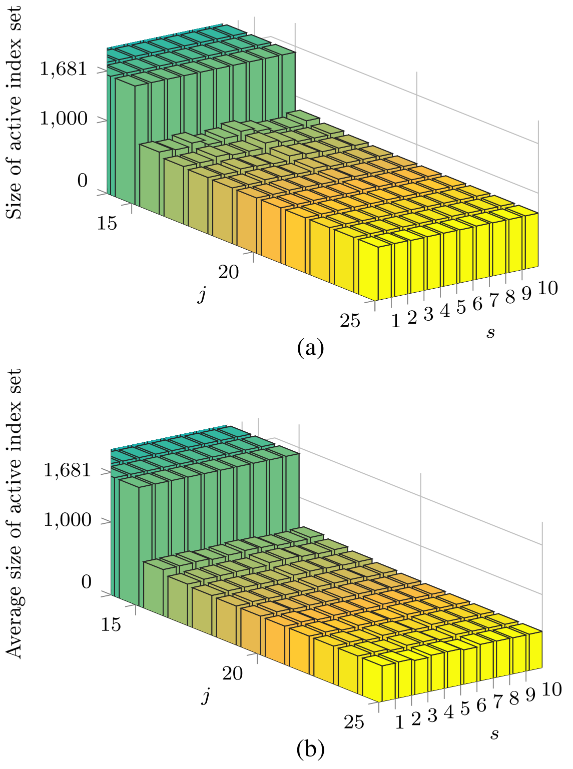


Fig. 5. Number of active internal states of each sensor versus the consensus iteration index j. (a) For a single time step and simulation run. (b) Averaged over all time steps and simulation runs.

the detection performance and tracking performance are significantly better and, for (G)LC-BF-20, they are already very close to the optimal performance of C-BF; however, this improved performance requires a further increase of the amount of communication. In conclusion, the significantly better performance of the proposed distributed BF comes at the cost of a significantly larger amount of communication.

Finally, we show how the size of the active index set of each sensor evolves with progressing consensus iterations. For LC-BF-20, Fig. 5 depicts the number of active (thus, broadcast) internal states of each of the ten sensors as a function of the consensus iteration index j. We only show the results for $j = 14, \ldots, 25$ because for $j = 1, \ldots, 15$, no pruning is employed and thus the number of active internal states of each sensor is always $B = 1681$. One can see in Fig. 5(b) that when pruning is employed, the average number of active internal states immediately drops to a little more than one third of the number without pruning; it then further decreases (albeit slowly) with progressing consensus iterations, and it depends only weakly on the sensor index.

C. Heterogeneous Sensor Network

Next, we consider a heterogeneous sensor network where sensors $s \in \mathcal{S}_{\mathrm{b}} \triangleq \{1, 5, 6, 8, 10\}$ acquire only bearing measurements while sensors $s \in \mathcal{S}_{\mathrm{r}} \triangleq \{2, 3, 4, 7, 9\}$ acquire only range measurements. Thus, for $s \in \mathcal{S}_{\mathrm{b}}$, equation (5) becomes

$$z_k^{(s)} = \arctan\left(\frac{y_k - y^{(s)}}{x_k - x^{(s)}}\right) + w_k^{(s)}, \qquad (43)$$

TABLE II
SENSOR PARAMETERS FOR THE HETEROGENEOUS SCENARIO

s	$\sigma_{\mathrm{r}}^{(s)}$	$\sigma_{\mathrm{b}}^{(s)}$	$P_{\mathrm{d}}^{(s)}$	$\mu^{(s)}$
1	–	2°	0.8	2
2	100 m	–	0.95	5
3	150 m	–	0.9	3
4	200 m	–	0.85	2
5	–	0.5°	0.95	5
6	–	1°	0.9	3
7	100 m	–	0.95	5
8	–	2°	0.8	2
9	200 m	–	0.8	2
10	–	1°	0.9	3

where $w_k^{(s)}$ is an i.i.d. zero-mean Gaussian random process with standard deviation $\sigma_{\mathrm{b}}^{(s)}$. On the other hand, for $s \in \mathcal{S}_{\mathrm{r}}$, equation (5) reads

$$z_k^{(s)} = \sqrt{(x_k - x^{(s)})^2 + (y_k - y^{(s)})^2} + w_k^{(s)}, \qquad (44)$$

where $w_k^{(s)}$ is an i.i.d. zero-mean Gaussian random process with standard deviation $\sigma_{\mathrm{r}}^{(s)}$. At each sensor s, when the target is present, target-originated measurements $z_k^{(s)}$ are randomly generated with detection probability $P_{\mathrm{d}}^{(s)}(\boldsymbol{x}_k) = P_{\mathrm{d}}^{(s)}$ according to (43) or (44). The clutter measurements are uniformly distributed with the following pdfs: for $s \in \mathcal{S}_{\mathrm{b}}$, $f_{\mathrm{c}}^{(s)}(z) = 1/360°$ for $z \in [-180°, 180°)$ and $f_{\mathrm{c}}^{(s)}(z) = 0$ otherwise, and for $s \in \mathcal{S}_{\mathrm{r}}$, $f_{\mathrm{c}}^{(s)}(z) = 1/R$ for $z \leq R$ and $f_{\mathrm{c}}^{(s)}(z) = 0$ otherwise, where $R = 30\,\mathrm{km}$ is the maximum range of the sensors. The number of clutter measurements has a Poisson distribution with mean $\mu^{(s)}$. The sensor parameters $\sigma_{\mathrm{r}}^{(s)}$, $\sigma_{\mathrm{b}}^{(s)}$, $P_{\mathrm{d}}^{(s)}$, and $\mu^{(s)}$ are chosen differently for different sensors; their values are listed in Table II.

Figs. 6 and 7 show the averaged performance metrics previously shown for the homogeneous scenario in Figs. 3 and 4. Again, 100 simulation runs were performed. One can see in Fig. 6(a) that, similarly to the homogeneous scenario, the RMSE of all filters quickly decreases to an error floor after the initial detection of the target at time $k = 10$. However, differently from the homogeneous scenario, already LC-BF-10 performs significantly better than RE-BF. Again, larger dictionary sizes yield better results: LC-BF-20 performs quite close to C-BF, and in turn LC-BF-15 performs quite close to LC-BF-20. Furthermore, adaptive pruning in the LC does not increase the RMSE noticeably, and the RMSE of GLC-BF effectively equals that of LC-BF.

Fig. 6(b) shows that the EPDE of the proposed distributed BF is again effectively zero at almost all times but with a larger peak at the initial detection time $k = 10$ than in the homogeneous scenario. Differently from the homogeneous scenario, the EPDE of RE-BF also has a much larger initial peak when the target is born and afterwards takes about six time steps until it settles at a floor of about 3%. This means that already LC-BF-10 has a considerably better detection performance than RE-BF. Again, as in the homogeneous scenario, adaptive pruning does not have

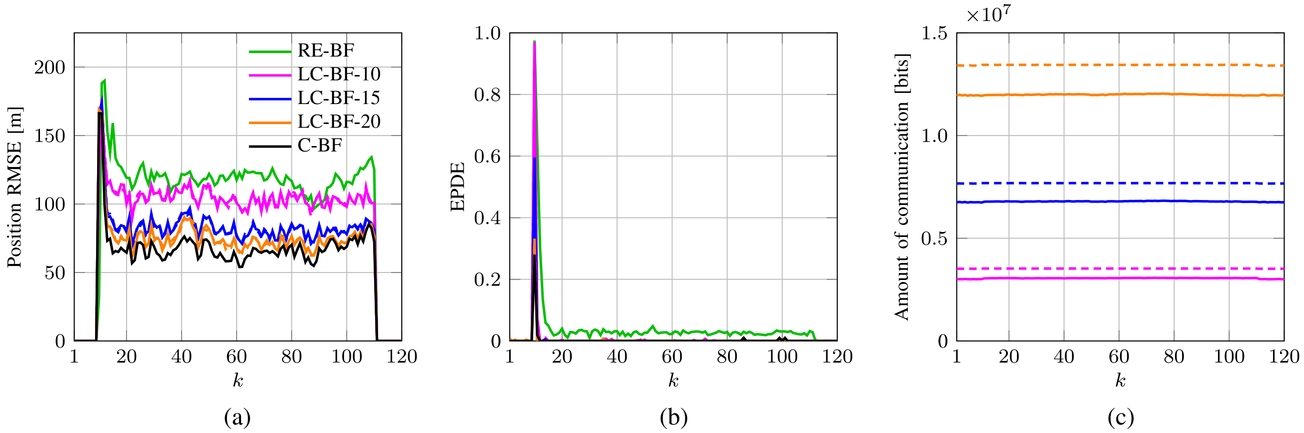


Fig. 6. Same as Fig. 3, but showing the results for the heterogeneous scenario.

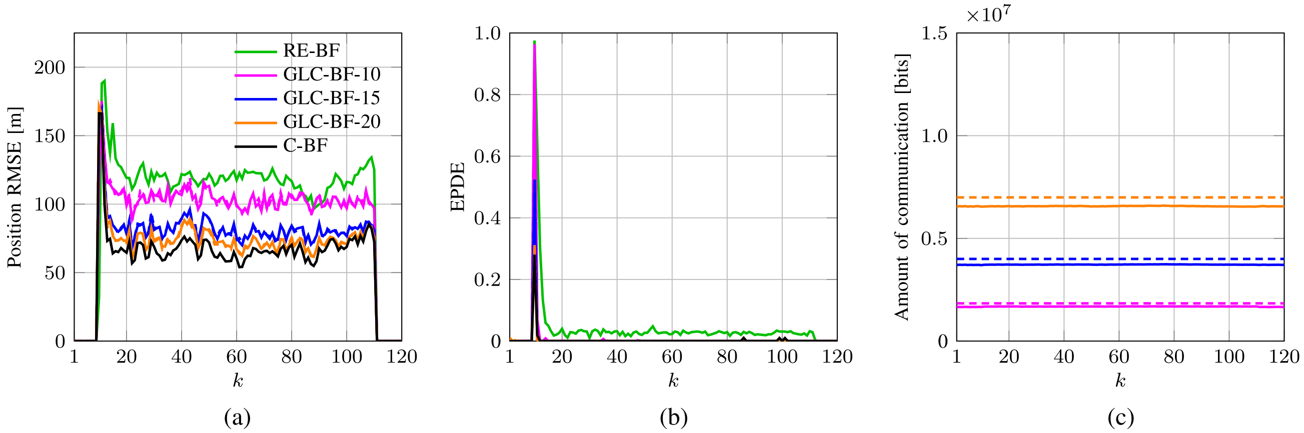


Fig. 7. Same as Fig. 4, but showing the results for the heterogeneous scenario.

a noticeable effect on the EPDE, and the EPDE of GLC-BF—shown in Fig. 7(b)—is effectively equal to that of LC-BF.

Fig. 6(c) shows that the amounts of communication required per filtering step by LC-BF-10, LC-BF-15, and LC-BF-20 are quite similar to those obtained in the homogeneous scenario, with the exception that the reduction achieved by pruning is smaller. Comparing with Fig. 7(c), we again see that GLC-BF requires only about half the communication of LC-BF; again, this does not imply a loss in detection/tracking performance. RE-BF now transmits about 9.5 kbit per filtering step. Accordingly, our discussion of Figs. 3(c) and 4(c) still applies here, with the difference that the performance gains of the proposed distributed BF over RE-BF are larger than in the homogeneous scenario.

We also investigated the dependence of the number of active internal states on the consensus iteration index j for the heterogeneous network scenario. Our simulation results are similar to those shown for the homogeneous network scenario in Fig. 5. In particular, they demonstrate that pruning yields a noticeable reduction of the number of active internal states, even though this reduction is smaller than in the homogeneous scenario.

IX. Conclusion

We addressed the problem of target tracking in decentralized sensor networks under the complicating assumptions that the presence of the target is uncertain, the measurements provided by the sensors are affected by clutter and missed detections, and the system model may be nonlinear and non-Gaussian. For this practically relevant scenario, we developed a distributed multisensor Bernoulli filter (BF) that approximates the optimal Bayesian multisensor filter while using only local intersensor communications. The proposed distributed BF employs an extended form of the likelihood consensus (LC) method and a particle-based implementation. The LC method is extended to account for both the target presence and target absence hypotheses, and to include an adaptive pruning of the LC expansion coefficients via thresholding operations. We also proposed a variant of the LC, referred to as group LC, that is advantageous for networks containing star-connected sensor groups.

We presented simulation results demonstrating the performance of the proposed distributed BF for homogeneous and heterogeneous sensor networks with substantial measurement noise and clutter. We verified that for increasing size of the LC dictionary, the performance of the proposed distributed BF approaches that of the optimal centralized multisensor BF. We also found that for a sufficient size of the LC dictionary, the proposed distributed BF outperforms the distributed BF proposed in [8] with respect to both detection and estimation. This performance advantage comes at the expense of a significantly higher amount of intersensor communication. Finally, we demonstrated that both the proposed adaptive pruning of the LC expansion coefficients and the group LC result in substantial reductions of intersensor communication without significantly reducing the detection and tracking performance.

Promising directions for future research include the use of sparse reconstruction techniques [37] and alternative dictionaries in the LC method. We expect that this will result in an improved accuracy of approximation and/or an additional reduction of the communication requirements. In the context of the GLC method, the development of a distributed algorithm for establishing and disseminating the relevant network configuration would be desirable. Finally, extensions of the proposed method to the cases of an unknown number of multiple targets (cf. [1], [17], [24], [38]–[40]) and of mobile sensors with uncertain positions (cf. [41]–[43]) are further interesting research topics.

Acknowledgment

The authors would like to thank Dr. P. Rajmic for stimulating discussions. They are also grateful to the anonymous reviewers, whose comments led to an improvement of this paper.

References

[1] Y. Bar-Shalom, P. K. Willett, and X. Tian, *Tracking and Data Fusion: A Handbook of Algorithms*. Storrs, CT, USA: YBS Publishing, 2011.

[2] F. Zhao and L. Guibas, *Wireless Sensor Networks: An Information Processing Approach*. San Francisco, CA, USA: Morgan Kaufmann, 2004.

[3] B. Ristic, B.-T. Vo, B.-N. Vo, and A. Farina, "A tutorial on Bernoulli filters: Theory, implementation and applications," *IEEE Trans. Signal Process.*, vol. 61, no. 13, pp. 3406–3430, Jul. 2013.

[4] B. T. Vo, C. M. See, N. Ma, and W. T. Ng, "Multi-sensor joint detection and tracking with the Bernoulli filter," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 48, no. 2, pp. 1385–1402, Apr. 2012.

[5] G. Papa, P. Braca, S. Horn, S. Marano, V. Matta, and P. Willett, "Multisensor adaptive Bayesian tracking under time-varying target detection probability," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 52, no. 5, pp. 2193–2209, Oct. 2016.

[6] M. B. Guldogan, "Consensus Bernoulli filter for distributed detection and tracking using multi-static Doppler shifts," *IEEE Signal Process. Lett.*, vol. 21, no. 6, pp. 672–676, Jun. 2014.

[7] D. Clark, S. Julier, R. P. S. Mahler, and B. Ristić, "Robust multi-object sensor fusion with unknown correlations," in *Proc. Sensor Signal Process. Defence*, London, UK, Sep. 2010, pp. 1–5.

[8] S. S. Dias and M. G. S. Bruno, "Distributed Bernoulli filters for joint detection and tracking in sensor networks," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 2, no. 3, pp. 260–275, Sep. 2016.

[9] A. H. Sayed, "Adaptive networks," *Proc. IEEE*, vol. 102, no. 4, pp. 460–497, Apr. 2014.

[10] C. Fantacci, B.-N. Vo, B.-T. Vo, G. Battistelli, and L. Chisci, "Consensus labeled random finite set filtering for distributed multi-object tracking," 2015. [Online]. http://arxiv.org/abs/1501.01579

[11] O. Hlinka, O. Slučiak, F. Hlawatsch, P. M. Djurić, and M. Rupp, "Likelihood consensus and its application to distributed particle filtering," *IEEE Trans. Signal Process.*, vol. 60, no. 8, pp. 4334–4349, Aug. 2012.

[12] O. Hlinka, F. Hlawatsch, and P. M. Djurić, "Consensus-based distributed particle filtering with distributed proposal adaptation," *IEEE Trans. Signal Process.*, vol. 62, no. 12, pp. 3029–3041, Jun. 2014.

[13] R. Repp, P. Rajmic, F. Meyer, and F. Hlawatsch, "Target tracking using a distributed particle-PDA filter with sparsity-promoting likelihood consensus," in *Proc. IEEE Statist. Signal Process.*, Freiburg, Germany, Jun. 2018, pp. 653–657.

[14] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proc. IEEE*, vol. 95, no. 1, pp. 215–233, Jan. 2007.

[15] A. G. Dimakis, S. Kar, J. M. F. Moura, M. G. Rabbat, and A. Scaglione, "Gossip algorithms for distributed signal processing," *Proc. IEEE*, vol. 98, no. 11, pp. 1847–1864, Nov. 2010.

[16] R. Repp, G. Papa, F. Meyer, P. Braca, and F. Hlawatsch, "A distributed Bernoulli filter based on likelihood consensus with adaptive pruning," in *Proc. FUSION-18*, Cambridge, U.K., Jul. 2018, pp. 2449–2456.

[17] R. P. S. Mahler, *Statistical Multisource-Multitarget Information Fusion*. Norwood, MA, USA: Artech House, 2007.

[18] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2508–2530, Jun. 2006.

[19] P. Braca, S. Marano, and V. Matta, "Enforcing consensus while monitoring the environment in wireless sensor networks," *IEEE Trans. Signal Process.*, vol. 56, no. 7, pp. 3375–3380, Jul. 2008.

[20] H. V. Poor, *An Introduction to Signal Detection and Estimation*. New York, NY, USA: Springer, 1994.

[21] D. Musicki and R. J. Evans, "Multiscan multitarget tracking in clutter with integrated track splitting filter," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 45, no. 4, pp. 1432–1447, Oct. 2009.

[22] D. Musicki and R. Evans, "Joint integrated probabilistic data association: JIPDA," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 40, no. 3, pp. 1093–1099, Jul. 2004.

[23] B.-T. Vo, B.-N. Vo, and A. Cantoni, "The cardinality balanced multitarget multi-Bernoulli filter and its implementations," *IEEE Trans. Signal Process.*, vol. 57, no. 2, pp. 409–423, Feb. 2009.

[24] J. L. Williams, "Marginal multi-Bernoulli filters: RFS derivation of MHT, JIPDA and association-based MeMBer," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 51, no. 3, pp. 1664–1687, Jul. 2015.

[25] B.-N. Vo, S. Singh, and A. Doucet, "Sequential Monte Carlo methods for multi-target filtering with random finite sets," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 41, no. 4, pp. 1224–1245, Oct. 2005.

[26] V. Elvira, L. Martino, D. Luengo, and M. F. Bugallo, "Efficient multiple importance sampling estimators," *IEEE Signal Process. Lett.*, vol. 22, no. 10, pp. 1757–1761, Oct. 2015.

[27] A. Doucet, N. de Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*. New York, NY, USA: Springer, 2001.

[28] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174–188, Feb. 2002.

[29] L. Xiao, S. Boyd, and S. Lall, "A scheme for robust distributed sensor fusion based on average consensus," in *Proc. 16th ACM/IEEE Int. Conf. Inf. Process. Sensor Netw*, Boise, ID, USA, Apr. 2005, pp. 63–70.

[30] O. Hlinka, F. Hlawatsch, and P. M. Djurić, "Distributed particle filtering in agent networks: A survey, classification, and comparison," *IEEE Signal Process. Mag.*, vol. 30, no. 1, pp. 61–81, Jan. 2013.

[31] D. Mosk-Aoyama and D. Shah, "Fast distributed algorithms for computing separable functions," *IEEE Trans. Inf. Theory*, vol. 54, no. 7, pp. 2997–3007, Jul. 2008.

[32] Å. Björck, *Numerical Methods for Least Squares Problems*. Philadelphia, PA, USA: SIAM, 1996.

[33] C. L. Lawson and R. J. Hanson, *Solving Least Squares Problems*. Philadelphia, PA: SIAM, 1995.

[34] S. Farahmand, S. I. Roumeliotis, and G. B. Giannakis, "Set-membership constrained particle filter: Distributed adaptation for sensor networks," *IEEE Trans. Signal Process.*, vol. 59, no. 9, pp. 4122–4138, Sep. 2011.

[35] V. Savic, H. Wymeersch, and S. Zazo, "Belief consensus algorithms for fast distributed target tracking in wireless sensor networks," *Signal Process.*, vol. 95, pp. 149–160, 2014.

[36] Y. Bar-Shalom, T. Kirubarajan, and X.-R. Li, *Estimation with Applications to Tracking and Navigation*. New York, NY, USA: Wiley, 2002.

[37] S. Foucart and H. Rauhut, *A Mathematical Introduction to Compressive Sensing*. Basel, Switzerland: Birkhäuser, 2013.

[38] F. Meyer, P. Braca, P. Willett, and F. Hlawatsch, "A scalable algorithm for tracking an unknown number of targets using multiple sensors," *IEEE Trans. Signal Process.*, vol. 65, no. 13, pp. 3478–3493, Jul. 2017.

[39] F. Meyer *et al.*, "Message passing algorithms for scalable multitarget tracking," *Proc. IEEE*, vol. 106, no. 2, pp. 221–259, Feb. 2018.

[40] D. B. Reid, "An algorithm for tracking multiple targets," *IEEE Trans. Autom. Control*, vol. 24, no. 6, pp. 843–854, Dec. 1979.

[41] A. T. Ihler, J. W. Fisher, R. L. Moses, and A. S. Willsky, "Nonparametric belief propagation for self-localization of sensor networks," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 4, pp. 809–819, Apr. 2005.

[42] H. Wymeersch, J. Lien, and M. Z. Win, "Cooperative localization in wireless networks," *Proc. IEEE*, vol. 97, no. 2, pp. 427–450, Feb. 2009.

[43] 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.

Giuseppe Papa received the B.Sc. degree (*summa cum laude*) and the M.Sc. degree (*summa cum laude*) in computer engineering in 2007 and 2011, respectively, and the Ph.D. degree in electronics and computer engineering in 2015, all from Second University of Naples, Aversa (CE), Italy. In 2011–2012, 2014, and 2015, he was a Visiting Researcher with the Centre for Maritime Research and Experimentation (CMRE), La Spezia, Italy, in the Maritime Situational Awareness program. In 2015, he was a Postdoctoral Associate with the Institute of Telecommunications, TU Wien, Vienna, Austria. Since 2016, he has been working in the software industry.

Rene Repp received the B.Sc. degree in electrical engineering and the Dipl.-Ing. degree (M.Sc. equivalent) in telecommunications engineering from TU Wien, Vienna, Austria, in 2012 and 2016, respectively. He is currently working toward the Ph.D. degree at the Institute of Telecommunications, TU Wien, where he has been a University Assistant since 2016. His research interests include multi-object tracking, distributed signal processing, point process theory, computer vision, and machine learning.

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 joined CMRE as a Research Scientist. He is currently a Postdoctoral Associate with the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA, USA. His research interests include signal processing for wireless sensor networks, localization and tracking, information-seeking control, message passing algorithms, and finite set statistics. He was 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. He is an Erwin Schrödinger Fellow.

Paolo Braca (M'14–SM'17) received the Laurea degree (*summa cum laude*) in electronic engineering and the Ph.D. degree (highest rank) in information engineering from the University of Salerno, Fisciano, Italy, in 2006 and 2010, respectively. In 2009, he was a Visiting Scholar with the Department of Electrical and Computer Engineering, University of Connecticut, Storrs, CT, USA. In 2010–2011, he was a Postdoctoral Associate with the University of Salerno, Italy. In 2011, he joined the NATO Science & Technology Organization Centre for Maritime Research and Experimentation (CMRE), La Spezia, Italy, where he is currently a Senior Scientist with the Research Department. He conducts research in the general area of statistical signal processing with emphasis on detection and estimation theory, wireless sensor networks, multi-agent algorithms, target tracking and data fusion, adaptation and learning over graphs, and radar and sonar signal processing. He is coauthor of about 100 publications in international scientific journals and conference proceedings. He was awarded the National Scientific Qualification to function as an Associate and Full Professor in Italian universities in 2017 and 2018, respectively. He was a recipient of the Best Student Paper Award (first runner-up) at FUSION 2009. He received the NATO STO Scientific Achievement Award 2017. He is an Associate Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING, the IEEE TRANSACTIONS ON AEROSPACE AND ELECTRONIC SYSTEMS, the *ISIF Journal of Advances in Information Fusion*, and the *EURASIP Journal on Advances in Signal Processing*. In 2017, he was the Lead Guest Editor of the Special Issue on "Sonar Multi-Sensor Applications and Techniques" of *IET Radar, Sonar & Navigation*. He was an Associate Editor of the IEEE SIGNAL PROCESSING MAGAZINE (E-Newsletter) from 2014 to 2016.

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. His research interests include statistical and compressive signal processing methods and their application to localization and sensor networks.

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 was on the technical committees of numerous IEEE conferences. He was an Associate Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING from 2003 to 2007, the IEEE TRANSACTIONS ON INFORMATION THEORY from 2008 to 2011, and 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. He is a EURASIP Fellow.