This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSP.2017.2688966, IEEE Transactions on Signal Processing

A Scalable Algorithm for Tracking an Unknown Number of Targets Using Multiple Sensors

Florian Meyer, *Member, IEEE*, Paolo Braca, *Member, IEEE*, Peter Willett, *Fellow, IEEE*, and Franz Hlawatsch, *Fellow, IEEE*

***Abstract*—We propose an algorithm for tracking an unknown number of targets based on measurements provided by multiple sensors. Our algorithm achieves low computational complexity and excellent scalability by running belief propagation on a suitably devised factor graph. A redundant formulation of data association uncertainty and the use of "augmented target states" including binary target indicators make it possible to exploit statistical independencies for a drastic reduction of complexity. An increase in the number of targets, sensors, or measurements leads to additional variable nodes in the factor graph but not to higher dimensions of the messages. As a consequence, the complexity of our method scales only quadratically in the number of targets, linearly in the number of sensors, and linearly in the number of measurements per sensor. The performance of the method compares well with that of previously proposed methods, including methods with a less favorable scaling behavior. In particular, our method can outperform multisensor versions of the probability hypothesis density (PHD) filter, the cardinalized PHD filter, and the multi-Bernoulli filter.**

***Index Terms*—Multitarget tracking, data association, belief propagation, message passing, factor graph, sensor network.**

I. Introduction

A. Multitarget Tracking Using Multiple Sensors

Multitarget tracking is important in many applications including surveillance, autonomous driving, biomedical analytics, robotics, and oceanography [1]–[3]. Multitarget tracking aims at estimating the states—i.e., positions and possibly further parameters—of moving objects (targets) over time, based on measurements provided by sensing devices such as radar, sonar, or cameras [2]. Often information from multiple sensors is required to obtain satisfactory reliability and accuracy. The number of targets is usually unknown [3] and there is a data association uncertainty, i.e., an unknown association between measurements and targets [2].

F. Meyer and P. Braca are with the NATO STO Centre for Maritime Research and Experimentation, La Spezia 19126, Italy (e-mail: {florian.meyer, paolo.braca}@cmre.nato.int). F. Meyer is currently on leave with the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA, USA. P. Willett is with the Department of ECE, University of Connecticut, Storrs, CT 06269, USA (e-mail: willett@engr.uconn.edu). F. Hlawatsch is with the Institute of Telecommunications, TU Wien, 1040 Vienna, Austria and with Brno University of Technology, Brno, Czech Republic (e-mail: franz.hlawatsch@tuwien.ac.at). This work was supported by the NATO Supreme Allied Command Transformation under projects SAC000601 and SAC000608, by the Naval Postgraduate School via ONR contract N00244-16-1-0017, by the Austrian Science Fund (FWF) under grants P27370-N30 and J3886-N31, by the National Sustainability Program of the European Commission under project LO1401, and by the Czech Science Foundation (GAČR) under grant 17-19638S. Parts of this paper were previously presented at Fusion 2015, Washington D.C., USA, July 2015 and at Fusion 2016, Heidelberg, Germany, July 2016.

Traditional methods for multitarget tracking include the joint probabilistic data association (JPDA) filter [2] and the multi-hypothesis tracker (MHT) [4] and their extensions to multiple sensors [5]–[7]. Most of these methods assume that the number of targets is fixed and known, which is typically not true in practice. Because of this assumption, most traditional methods do not solve the track management problem, i.e., they are unable to create or cancel a track when a target appears or disappears, respectively. Track management extensions of traditional methods include the joint integrated probabilistic data association (JIPDA) filter [8], the joint integrated track splitting (JITS) filter [9], and the search-initialize-track filter [10].

A more recent class of multitarget tracking methods is based on finite set statistics (FISST). These methods calculate an approximation of the posterior multiobject probability density function (pdf), which is a joint distribution of the *unordered* target states. Typically, this quantity is then used to estimate a (possibly unordered) set of target states, which is described as a random finite set. Notable examples include the probability hypothesis density (PHD) filter [3], [11]–[13], the cardinalized PHD (CPHD) filter [3], [14], [15], the Bernoulli filter [16], and the multi-Bernoulli (MB) filter [3], [17]. These algorithms avoid a specific ordering of targets and measurements and incorporate the appearance and disappearance of targets in a Bayes-optimum way. However, most existing FISST-based methods are restricted to a single sensor.

Even more recently, FISST-based multitarget tracking methods using labeled random finite sets have been proposed. These filters track an unknown number of targets that are identified by an (unobserved) label, and thus are able to estimate individual target tracks. In particular, the labeled multi-Bernoulli (LMB) filter [18] and the δ-generalized LMB filter [19] achieve good estimation accuracy with a computational complexity that is similar to that of the CPHD filter. Alternatively, the track-oriented marginal Bernoulli/Poisson (TOMB/P) filter and the measurement-oriented marginal Bernoulli/Poisson (MOMB/P) filter proposed in [20] can estimate individual target tracks by integrating probabilistic data association into FISST-based sequential estimation; they are not based on labeled random finite sets.

In the case of low-observable targets, i.e., targets leading to measurements with a low signal-to-noise ratio, reliable detection and tracking using a single sensor may be impossible. Theoretical results [21] suggest that the probability of detection can be strongly improved by increasing the number of sensors. Unfortunately, the computational complexity of optimum multisensor-multitarget tracking scales exponentially

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSP.2017.2688966, IEEE Transactions on Signal Processing

in the number of sensors, number of targets, and number of measurements per sensor [22]–[25]. Computationally feasible multisensor-multitarget tracking algorithms include the iterator-corrector (C)PHD or briefly IC-(C)PHD filter [26], the approximate product multisensor (C)PHD filter [27], the partition-based multisensor (C)PHD (MS-(C)PHD) filter [23], [24], two heuristic extensions of the PHD filter to multiple sensors [28], [29], and an IC version of the δ-generalized LMB filter [30]. These algorithms either use approximations of unknown fidelity and thus may not be able to fully realize the performance gains promised by multiple sensors, or they still scale poorly in relevant system parameters. A further disadvantage of the IC-(C)PHD filter and other IC-based filters is that the performance depends on the order in which the measurements of the different sensors are processed [23], [24], [26], [27]. We note that the TOMB/P and MOMB/P filters [20] have been formulated only for a single sensor.

B. The Proposed Method and Other Message Passing Methods

Here, we propose a multisensor method for multitarget tracking with excellent scalability in the number of targets, number of sensors, and number of measurements per sensor. Our method allows for an unknown, time-varying number of targets (up to a specified *maximally possible* number of targets), i.e., it implicitly performs track management. These advantages are obtained by performing ordered estimation using belief propagation (BP) message passing, based on the sum-product algorithm [31]–[34]. Contrary to most FISST-based algorithms, which calculate an approximation of the *joint* posterior multiobject pdf, BP provides accurate approximations of the *marginal* posterior pdfs for the individual targets. These are then used to perform Bayesian detection and estimation of the target states.

The proposed BP method is derived by formulating a detection-estimation problem involving all the target states, existence variables, and association variables—which are modeled via random vectors—for all times, targets, and sensors. Enabling techniques for our method are the redundant formulation of data association uncertainty and a variant of the corresponding BP-based data association algorithm presented in [35], [36]. A further important principle is the definition of "augmented target states" that include binary target existence indicators. In contrast to FISST-based methods, the joint augmented target state is ordered and has a fixed number of components.

By this new formulation of the multisensor-multitarget detection-estimation problem, the statistical structure of the problem can be described by a factor graph, and the problem can be solved using loopy BP. The advantage of the BP approach is that it exploits conditional statistical independencies for a drastic reduction of complexity [31]–[34]. We use a "detailed" factor graph in which each target state and each association variable is modeled as an individual node. Because this factor graph involves only low-dimensional variables, the resulting BP algorithm does not perform high-dimensional operations. As a consequence, the complexity of our method scales only quadratically in the number of targets, linearly in the number of sensors, and linearly in the number of measurements per sensors (assuming a fixed number of message passing iterations). In addition, because our algorithm uses particle-based calculations of all messages and beliefs, it is suited to general nonlinear, non-Gaussian measurement and state evolution models.

Simulation results in a challenging scenario with intersecting target trajectories demonstrate that our algorithm exhibits excellent scalability and, at the same time, its performance compares well with that of previously proposed methods. This includes methods with a less favorable scaling behavior, namely, cubic in the number of measurements and in the number of targets. In particular, our method can outperform the IC-PHD and IC-CPHD filters [26], an IC extension of the MB filter, and the MS-PHD and MS-CPHD filters [23], [24]. Furthermore, its performance does not depend on an assumed order of processing the measurements of the different sensors.

To the best of our knowledge, previously proposed BP methods for multisensor-multitarget tracking are limited to the method presented in [37] and our previous method in [38]. The model used in [37] results in a tree-structured factor graph for which BP is exact but also in an unappealing scalability in the number of targets. In contrast, our method is based on a detailed (but loopy) factor graph that gives rise to low-dimensional messages and, in turn, results in the attractive scaling properties described above. Furthermore, both methods [37], [38] assume that the number of targets is known, whereas our present method is suited to an unknown number of targets.

BP-based methods have also been proposed for the problems of data association alone or data association within a multitarget tracking scheme where the tracking itself is not done by BP. In particular, BP has been used in [35] and [36] to calculate approximate marginal association probabilities for a single sensor; in [39] to calculate exact marginal association probabilities for a single sensor; in [40] to calculate approximate association probabilities for multiple sensors with overlapping regions of interest; and in [20] to calculate approximate association probabilities for a single sensor. Our method integrates the model and BP algorithm for data association from [35], [36]. However, in contrast to the methods mentioned above, it uses BP for the overall multisensor-multitarget tracking problem, of which data association is only a part. Furthermore, as opposed to [20], which also uses the model and BP algorithm for data association from [35], [36], our method is not based on FISST, can be used for arbitrary nonlinear and non-Gaussian problems, and is suitable for multiple sensors.

C. Paper Organization

This paper is organized as follows. The system model and statistical formulation are described in Section II. In Section III, we briefly review the framework of factor graphs and BP. The factor graph underlying the proposed algorithm is derived in Section IV. Section V develops the proposed multisensor-multitarget tracking algorithm. A particle-based implementation is presented in Section VI. Section VII proposes a scheme for choosing the birth and survival parameters. In Section VIII, relations of the proposed method to existing methods are discussed. Finally, Section IX presents numerical

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSP.2017.2688966, IEEE Transactions on Signal Processing

results for simulated data and for real data from a multistatic sonar tracking experiment. We note that this paper advances over the preliminary account of our algorithm provided in our conference publication [41] by adding a particle-based implementation, a scheme for choosing birth and survival parameters, a detailed discussion of relations to existing methods, additional performance results, and an experimental verification of scaling properties.

D. Notation

We denote vectors by boldface lower-case characters (e.g., $\mathbf{x}$), matrices by boldface upper-case characters (e.g., $\mathbf{X}$), and sets by calligraphic upper-case characters (e.g., $\mathcal{X}$). We write k for the index of a *potential target* (PT), n for the index of a time step, m for the index of a measurement, s for the index of a sensor, and j for the index of a particle. The numbers of PTs, sensors, and measurements of sensor s at time n are denoted by K, S, and $M_n^{(s)}$, respectively. Integrals are over the entire space of the integration variable. $\mathbf{I}_N$ denotes the $N\times N$ identity matrix. We write $f(\mathbf{x})$ for the pdf of a continuous random vector $\mathbf{x}$ and $p(\mathbf{x})$ for the probability mass function (pmf) of a discrete random vector $\mathbf{x}$. The shorthand $\mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma})$ denotes a Gaussian pdf with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$.

II. System Model and Statistical Formulation

In this section, we describe our system model and a corresponding statistical formulation, and we state the multitarget detection-estimation problem to be solved.

A. Potential Targets and Augmented Target States

We consider *at most* K targets with time-varying states. We describe this situation by introducing PTs $k \in \mathcal{K} \triangleq \{1,\dots,K\}$. The existence of the PTs is modeled by binary variables $r_{n,k}\in\{0,1\}$, i.e., PT k exists at time n if and only if $r_{n,k}=1$. We also define the vector $\mathbf{r}_n \triangleq \left[r_{n,1}\cdots r_{n,K}\right]^{\mathrm{T}}$. The state $\mathbf{x}_{n,k}$ of PT k at time n consists of the PT's position and possibly further parameters. It will be convenient to formally consider a PT state $\mathbf{x}_{n,k}$ also if $r_{n,k}=0$. We define the *augmented state* $\mathbf{y}_{n,k} \triangleq [\mathbf{x}_{n,k}^{\mathrm{T}}\ r_{n,k}]^{\mathrm{T}}$ and the *joint augmented state* $\mathbf{y}_n \triangleq \left[\mathbf{y}_{n,1}^{\mathrm{T}}\cdots\mathbf{y}_{n,K}^{\mathrm{T}}\right]^{\mathrm{T}}$.

While in our model a PT state $\mathbf{x}_{n,k}$ is formally defined also if $r_{n,k}=0$, the states of nonexisting PTs are obviously irrelevant. Accordingly, all pdfs and BP messages defined for an augmented state, $\phi(\mathbf{y}_{n,k}) = \phi(\mathbf{x}_{n,k},r_{n,k})$, have the property that for $r_{n,k}=0$,

$$\phi(\mathbf{x}_{n,k},0) = \phi_{n,k}\, f_{\mathrm{D}}(\mathbf{x}_{n,k}), \tag{1}$$

where $f_{\mathrm{D}}(\mathbf{x}_{n,k})$ is a "dummy pdf." The form (1) must be consistent with a message multiplication operation (such as Equation (21) in Section III), in the sense that the resulting message product can still be expressed as in (1). This implies that the dummy pdf $f_{\mathrm{D}}(\mathbf{x}_{n,k})$ satisfies $f_{\mathrm{D}}^2(\mathbf{x}_{n,k}) = f_{\mathrm{D}}(\mathbf{x}_{n,k})$ for all values of $\mathbf{x}_{n,k}$. Because $f_{\mathrm{D}}(\mathbf{x}_{n,k})$ must also integrate to one, it follows that $f_{\mathrm{D}}(\mathbf{x}_{n,k})$ is 1 on an arbitrary support of area/volume 1 and 0 outside that support. Let

$$\varphi(r_{n,k}) \triangleq \int \phi(\mathbf{x}_{n,k},r_{n,k})\,\mathrm{d}\mathbf{x}_{n,k}\,. \tag{2}$$

We then have for $r_{n,k}=0$

$$\varphi(0) = \int \phi(\mathbf{x}_{n,k},0)\,\mathrm{d}\mathbf{x}_{n,k} = \phi_{n,k}\int f_{\mathrm{D}}(\mathbf{x}_{n,k})\,\mathrm{d}\mathbf{x}_{n,k} = \phi_{n,k}\,. \tag{3}$$

Furthermore, using (2) and (3) yields

$$\begin{aligned}\sum_{r_{n,k}\in\{0,1\}} &\int \phi(\mathbf{x}_{n,k},r_{n,k})\,\mathrm{d}\mathbf{x}_{n,k}\\ &= \int \phi(\mathbf{x}_{n,k},0)\,\mathrm{d}\mathbf{x}_{n,k} + \int \phi(\mathbf{x}_{n,k},1)\,\mathrm{d}\mathbf{x}_{n,k}\\ &= \phi_{n,k} + \varphi(1)\,.\end{aligned}$$

Hence, if $\sum_{r_{n,k}\in\{0,1\}}\int \phi(\mathbf{x}_{n,k},r_{n,k})\,\mathrm{d}\mathbf{x}_{n,k} = 1$, i.e., if $\phi(\mathbf{x}_{n,k},r_{n,k})$ is a true pdf/pmf in the sense of being normalized, then $\phi_{n,k}+\varphi(1)=1$. In that case, $\phi_{n,k}=\varphi(0)$ can be interpreted as a probability of nonexistence of PT k, i.e., of the event $r_{n,k}=0$, and $\varphi(1)$ can be interpreted as a probability of existence of PT k, i.e., of the event $r_{n,k}=1$. However, note that in general, $\phi(\mathbf{x}_{n,k},r_{n,k})$ is not necessarily a pdf/pmf.

The augmented target states $\mathbf{y}_{n,k} = [\mathbf{x}_{n,k}^{\mathrm{T}}\ \ r_{n,k}]^{\mathrm{T}}$ are assumed to evolve independently according to Markovian dynamic models [2], [3], and at time $n=0$, they are assumed statistically independent across k with prior pdfs $f(\mathbf{y}_{0,k}) = f(\mathbf{x}_{0,k},r_{0,k})$. Thus, the pdf of $\mathbf{y} \triangleq \left[\mathbf{y}_0^{\mathrm{T}}\cdots\mathbf{y}_n^{\mathrm{T}}\right]^{\mathrm{T}}$ factorizes as

$$f(\mathbf{y}) = \prod_{k=1}^{K} f(\mathbf{y}_{0,k}) \prod_{n'=1}^{n} f(\mathbf{y}_{n',k}|\mathbf{y}_{n'-1,k}). \tag{4}$$

Here, the single-target augmented state transition pdf $f(\mathbf{y}_{n,k}|\mathbf{y}_{n-1,k}) = f(\mathbf{x}_{n,k},r_{n,k}|\mathbf{x}_{n-1,k},r_{n-1,k})$ is given as follows. If PT k did not exist at time $n-1$, i.e., $r_{n-1,k}=0$, then the probability that it exists at time n, i.e., $r_{n,k}=1$, is given by the birth probability $p_{n,k}^{\mathrm{b}}$, and if it does exist at time n, its state $\mathbf{x}_{n,k}$ is distributed according to the birth pdf $f_{\mathrm{b}}(\mathbf{x}_{n,k})$. Thus, for $r_{n-1,k}=0$, we have

$$f(\mathbf{x}_{n,k},r_{n,k}|\mathbf{x}_{n-1,k},0) = \begin{cases}(1-p_{n,k}^{\mathrm{b}})\, f_{\mathrm{D}}(\mathbf{x}_{n,k}), & r_{n,k}=0\\ p_{n,k}^{\mathrm{b}}\, f_{\mathrm{b}}(\mathbf{x}_{n,k}), & r_{n,k}=1.\end{cases} \tag{5}$$

If PT k existed at time $n-1$, i.e., $r_{n-1,k}=1$, then the probability that it still exists at time n, i.e., $r_{n,k}=1$, is given by the survival probability $p_{n,k}^{\mathrm{s}}$, and if it still exists at time n, its state $\mathbf{x}_{n,k}$ is distributed according to the state transition pdf $f(\mathbf{x}_{n,k}|\mathbf{x}_{n-1,k})$. Thus, for $r_{n-1,k}=1$, we have

$$f(\mathbf{x}_{n,k},r_{n,k}|\mathbf{x}_{n-1,k},1) = \begin{cases}(1-p_{n,k}^{\mathrm{s}})\, f_{\mathrm{D}}(\mathbf{x}_{n,k}), & r_{n,k}=0\\ p_{n,k}^{\mathrm{s}}\, f(\mathbf{x}_{n,k}|\mathbf{x}_{n-1,k}), & r_{n,k}=1.\end{cases} \tag{6}$$

A possible strategy for choosing $p_{n,k}^{\mathrm{b}}$, $p_{n,k}^{\mathrm{s}}$, and $f_{\mathrm{b}}(\mathbf{x}_{n,k})$ is presented in Section VII. We note that our previous work in [38], which assumed that the number of targets is known, is a special case of this setup that uses survival probabilities $p_{n,k}^{\mathrm{s}}=1$ (existing targets always survive), birth probabilities $p_{n,k}^{\mathrm{b}}=0$ (no targets are born), and initial prior pdfs $f(\mathbf{x}_{0,k},r_{0,k})$ with $\int f(\mathbf{x}_{0,k},1)\,\mathrm{d}\mathbf{x}_{0,k}=1$ (at time $n=0$, all targets exist with probability 1).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSP.2017.2688966, IEEE Transactions on Signal Processing

B. Sensor Measurements

There are S sensors $s \in \mathcal{S} \triangleq \{1, \ldots, S\}$ that produce "thresholded" measurements resulting from a detection process (as performed, e.g., by a radar or sonar device). Let $\mathbf{z}_{n,m}^{(s)}$, $m \in \mathcal{M}_n^{(s)} \triangleq \{1, \ldots, M_n^{(s)}\}$ denote the measurements produced by sensor s at time n. We also define the stacked measurement vectors[1] $\mathbf{z}_n^{(s)} \triangleq \left[\mathbf{z}_{n,1}^{(s)\mathrm{T}} \cdots \mathbf{z}_{n,M_n^{(s)}}^{(s)\mathrm{T}}\right]^{\mathrm{T}}$ and $\mathbf{z}_n \triangleq \left[\mathbf{z}_n^{(1)\mathrm{T}} \cdots \mathbf{z}_n^{(S)\mathrm{T}}\right]^{\mathrm{T}}$, and the vector of all numbers of measurements $\mathbf{m}_n \triangleq \left[M_n^{(1)} \cdots M_n^{(S)}\right]^{\mathrm{T}}$.

Because the measurements are thresholded, the multitarget tracking problem is complicated by a data association uncertainty: it is unknown which measurement $\mathbf{z}_{n,m}^{(s)}$ originated from which PT k, and it is also possible that a measurement $\mathbf{z}_{n,m}^{(s)}$ did not originate from any PT (false alarm, clutter) or that a PT did not generate any measurement of sensor s (missed detection) [2], [3]. We make the usual assumption that at any time n, an existing target can generate at most one measurement at sensor s, and a measurement at sensor s can be generated by at most one existing target [2], [3]. The PT-measurement associations at sensor s and time n can then be described by the *target-oriented association variables*

$$a_{n,k}^{(s)} \triangleq \begin{cases} m \in \mathcal{M}_n^{(s)}, & \text{if at time } n, \text{ PT } k \text{ generates} \\ & \text{measurement } m \text{ at sensor } s \\ 0, & \text{if at time } n, \text{ PT } k \text{ does not gen-} \\ & \text{erate a measurement at sensor } s. \end{cases} \tag{7}$$

We also define $\mathbf{a}_n^{(s)} \triangleq \left[a_{n,1}^{(s)} \cdots a_{n,K}^{(s)}\right]^{\mathrm{T}}$ and $\mathbf{a}_n \triangleq \left[\mathbf{a}_n^{(1)\mathrm{T}} \cdots \mathbf{a}_n^{(S)\mathrm{T}}\right]^{\mathrm{T}}$.

An existing target k is detected by sensor s (in the sense that the target generates a measurement $\mathbf{z}_{n,m}^{(s)}$ at sensor s) with probability $P_{\mathrm{d}}^{(s)}(\mathbf{x}_{n,k})$, which may depend on the target state $\mathbf{x}_{n,k}$. The number of false alarms at sensor s is modeled by a Poisson pmf with mean $\mu^{(s)}$, and the distribution of each false alarm measurement at sensor s is described by the pdf $f_{\mathrm{FA}}\left(\mathbf{z}_{n,m}^{(s)}\right)$ [2], [3].

The dependence of the measurement vector $\mathbf{z} \triangleq \left[\mathbf{z}_1^{\mathrm{T}} \cdots \mathbf{z}_n^{\mathrm{T}}\right]^{\mathrm{T}}$ on the augmented state vector $\mathbf{y} = \left[\mathbf{y}_0^{\mathrm{T}} \cdots \mathbf{y}_n^{\mathrm{T}}\right]^{\mathrm{T}}$, the association vector $\mathbf{a} \triangleq \left[\mathbf{a}_1^{\mathrm{T}} \cdots \mathbf{a}_n^{\mathrm{T}}\right]^{\mathrm{T}}$, and the numbers-of-measurements vector[2] $\mathbf{m} \triangleq \left[\mathbf{m}_1^{\mathrm{T}} \cdots \mathbf{m}_n^{\mathrm{T}}\right]^{\mathrm{T}}$ is described by the global likelihood function $f(\mathbf{z}|\mathbf{y}, \mathbf{a}, \mathbf{m})$. With the commonly used assumption [2], [3] that given $\mathbf{y}$, $\mathbf{a}$, and $\mathbf{m}$, the measurements $\mathbf{z}_n^{(s)}$ are conditionally independent across time n and sensor index s, the global likelihood function factorizes as

$$f(\mathbf{z}|\mathbf{y}, \mathbf{a}, \mathbf{m}) = \prod_{n'=1}^{n} \prod_{s=1}^{S} f\left(\mathbf{z}_{n'}^{(s)} \middle| \mathbf{y}_{n'}, \mathbf{a}_{n'}^{(s)}, M_{n'}^{(s)}\right). \tag{8}$$

[1] Although the measurements $\mathbf{z}_{n,m}^{(s)}$, $m \in \mathcal{M}_n^{(s)}$ produced by sensor s have no inherent order, we give them an (arbitrary) order by arranging them in the vector $\mathbf{z}_n^{(s)}$. However, it will be seen that the result of the proposed BP method does not depend on the assumed order (see Section V-A).

[2] We note that the approach of modeling the measurement vector $\mathbf{z}$ and the number-of-measurements vector $\mathbf{m}$ as two separate random vectors was previously used in the original derivations of the (J)PDA filters and related methods [2], [8].

Assuming in addition that the different measurements $\mathbf{z}_{n,m}^{(s)}$ at sensor s are conditionally independent given $\mathbf{y}_n$, $\mathbf{a}_n^{(s)}$, and $M_n^{(s)}$, we have the further factorization [2], [3]

$$f\left(\mathbf{z}_n^{(s)} \middle| \mathbf{y}_n, \mathbf{a}_n^{(s)}, M_n^{(s)}\right) = \left(\prod_{m=1}^{M_n^{(s)}} f_{\mathrm{FA}}\left(\mathbf{z}_{n,m}^{(s)}\right)\right) \times \prod_{k \in \mathcal{D}_{\mathbf{a}_n, \mathbf{r}_n}^{(s)}} \frac{f\left(\mathbf{z}_{n,a_{n,k}^{(s)}}^{(s)} \middle| \mathbf{x}_{n,k}\right)}{f_{\mathrm{FA}}\left(\mathbf{z}_{n,a_{n,k}^{(s)}}^{(s)}\right)}. \tag{9}$$

Here, $\mathcal{D}_{\mathbf{a}_n, \mathbf{r}_n}^{(s)}$ denotes the set of existing targets detected at sensor s and time n, i.e., $\mathcal{D}_{\mathbf{a}_n, \mathbf{r}_n}^{(s)} \triangleq \left\{k \in \mathcal{K} : r_{n,k} = 1, a_{n,k}^{(s)} \neq 0\right\}$. If $\mathbf{z}_n^{(s)}$ is observed and thus fixed, $M_n^{(s)}$ is fixed as well and (9) can be written as

$$f\left(\mathbf{z}_n^{(s)} \middle| \mathbf{y}_n, \mathbf{a}_n^{(s)}, M_n^{(s)}\right) = C\left(\mathbf{z}_n^{(s)}\right) \prod_{k=1}^{K} g\left(\mathbf{x}_{n,k}, r_{n,k}, a_{n,k}^{(s)}; \mathbf{z}_n^{(s)}\right), \tag{10}$$

where $C\left(\mathbf{z}_n^{(s)}\right)$ is a normalization factor that depends only on $\mathbf{z}_n^{(s)}$ (and, thus, also on $M_n^{(s)}$) and $g\left(\mathbf{x}_{n,k}, r_{n,k}, a_{n,k}^{(s)}; \mathbf{z}_n^{(s)}\right)$ is defined as

$$g\left(\mathbf{x}_{n,k}, 1, a_{n,k}^{(s)}; \mathbf{z}_n^{(s)}\right) = \begin{cases} \dfrac{f\left(\mathbf{z}_{n,m}^{(s)} \middle| \mathbf{x}_{n,k}\right)}{f_{\mathrm{FA}}\left(\mathbf{z}_{n,m}^{(s)}\right)}, & a_{n,k}^{(s)} = m \in \mathcal{M}_n^{(s)} \\ 1, & a_{n,k}^{(s)} = 0 \end{cases}$$

$$g\left(\mathbf{x}_{n,k}, 0, a_{n,k}^{(s)}; \mathbf{z}_n^{(s)}\right) = 1. \tag{11}$$

Inserting (10) into (8) yields

$$f(\mathbf{z}|\mathbf{y}, \mathbf{a}, \mathbf{m}) = C(\mathbf{z}) \prod_{n'=1}^{n} \prod_{s=1}^{S} \prod_{k=1}^{K} g\left(\mathbf{x}_{n',k}, r_{n',k}, a_{n',k}^{(s)}; \mathbf{z}_{n'}^{(s)}\right), \tag{12}$$

where $C(\mathbf{z})$ is a normalization factor that depends only on $\mathbf{z}$.

C. Joint Prior Distribution of Association Variables and Numbers of Measurements

Under the assumption that given $\mathbf{y}$, the $\mathbf{a}_n^{(s)}$ and the $M_n^{(s)}$ are conditionally independent across n and s [2], [3], the joint prior pmf of the association vector $\mathbf{a}$ and the numbers-of-measurements vector $\mathbf{m}$ given $\mathbf{y}$ factorizes as

$$p(\mathbf{a}, \mathbf{m}|\mathbf{y}) = \prod_{n'=1}^{n} \prod_{s=1}^{S} p\left(\mathbf{a}_{n'}^{(s)}, M_{n'}^{(s)} \middle| \mathbf{y}_{n'}\right). \tag{13}$$

Assuming a random order of the measurements $\mathbf{z}_{n,m}^{(s)}$, $m \in \mathcal{M}_n^{(s)}$ at sensor s, with each possible order equally likely, and using the fact that the number of false alarm measurements is Poisson distributed with mean $\mu^{(s)}$, it is shown in [2], [10], [39] that

$$\begin{aligned} &p\left(\mathbf{a}_n^{(s)}, M_n^{(s)} \middle| \mathbf{y}_n\right) \\ &= \psi\left(\mathbf{a}_n^{(s)}\right) \frac{e^{-\mu^{(s)}} \left(\mu^{(s)}\right)^{M_n^{(s)} - \left|\mathcal{D}_{\mathbf{a}_n, \mathbf{r}_n}^{(s)}\right|}}{M_n^{(s)}!} \left(\prod_{k \in \mathcal{D}_{\mathbf{a}_n, \mathbf{r}_n}^{(s)}} P_{\mathrm{d}}^{(s)}(\mathbf{x}_{n,k})\right) \\ &\quad \times \prod_{k' \notin \mathcal{D}_{\mathbf{a}_n, \mathbf{r}_n}^{(s)}} \left[1\left(a_{n,k'}^{(s)}\right) - r_{n,k'} P_{\mathrm{d}}^{(s)}(\mathbf{x}_{n,k'})\right], \end{aligned} \tag{14}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSP.2017.2688966, IEEE Transactions on Signal Processing

where

$$\psi\big(\mathbf{a}_n^{(s)}\big) \triangleq \begin{cases} 0, & \exists k,k' \in \mathcal{K} \text{ with } k \neq k' \text{ such that} \\ & \quad a_{n,k}^{(s)} = a_{n,k'}^{(s)} \neq 0 \\ 1, & \text{otherwise,} \end{cases} \tag{15}$$

and $1(a)$ denotes the indicator function of the event $a = 0$ (i.e., $1(a) = 1$ if $a = 0$ and 0 otherwise). Note that the factor $\psi\big(\mathbf{a}_n^{(s)}\big)$ enforces the exclusion assumptions stated in Section II-B, i.e., that each existing target can generate at most one measurement at sensor s and each measurement at sensor s can be generated by at most one target. Since the dimension of $\mathbf{a}_n^{(s)}$ increases with the number of PTs K, the factor $\psi\big(\mathbf{a}_n^{(s)}\big)$ is a major reason for the poor scalability of the JPDA and related methods [2]. An alternative approach resulting in significantly improved scalability will be presented in Section IV-A. We can express (14) more compactly as

$$p\big(\mathbf{a}_n^{(s)}, M_n^{(s)} | \mathbf{y}_n\big) = C\big(M_n^{(s)}\big)\, \psi\big(\mathbf{a}_n^{(s)}\big) \times \prod_{k=1}^{K} h\big(\mathbf{x}_{n,k}, r_{n,k}, a_{n,k}^{(s)}; M_n^{(s)}\big), \tag{16}$$

where $C\big(M_n^{(s)}\big)$ is a normalization factor depending only on $M_n^{(s)}$ and $h\big(\mathbf{x}_{n,k}, r_{n,k}, a_{n,k}^{(s)}; M_n^{(s)}\big)$ is defined as

$$h\big(\mathbf{x}_{n,k}, 1, a_{n,k}^{(s)}; M_n^{(s)}\big) = \begin{cases} \dfrac{P_{\mathrm{d}}^{(s)}(\mathbf{x}_{n,k})}{\mu^{(s)}}, & a_{n,k}^{(s)} \in \mathcal{M}_n^{(s)} \\ 1 - P_{\mathrm{d}}^{(s)}(\mathbf{x}_{n,k}), & a_{n,k}^{(s)} = 0 \end{cases}$$

$$h\big(\mathbf{x}_{n,k}, 0, a_{n,k}^{(s)}; M_n^{(s)}\big) = 1\big(a_{n,k}^{(s)}\big). \tag{17}$$

D. Target Detection and State Estimation

The problem considered is detection of the PTs $k \in \mathcal{K}$ (i.e., of the binary target existence variables $r_{n,k}$) and estimation of the target states $\mathbf{x}_{n,k}$ from the past and present measurements of all the sensors $s \in \mathcal{S}$, i.e., from the total measurement vector $\mathbf{z} = \big[\mathbf{z}_1^{\mathrm{T}} \cdots \mathbf{z}_n^{\mathrm{T}}\big]^{\mathrm{T}}$. In our Bayesian setting, this essentially amounts to calculating the marginal posterior existence probabilities $p(r_{n,k}=1|\mathbf{z})$ and the marginal posterior pdfs $f(\mathbf{x}_{n,k}|r_{n,k} = 1, \mathbf{z})$. Target detection is performed by comparing $p(r_{n,k} = 1|\mathbf{z})$ to a threshold P_{th}, i.e., PT k is considered to exist if $p(r_{n,k} = 1|\mathbf{z}) > P_{\mathrm{th}}$ [42, Ch. 2]. For the detected targets k, an estimate of $\mathbf{x}_{n,k}$ is then produced by the minimum mean-square error (MMSE) estimator [42, Ch. 4]

$$\hat{\mathbf{x}}_{n,k}^{\mathrm{MMSE}} \triangleq \int \mathbf{x}_{n,k}\, f(\mathbf{x}_{n,k}|r_{n,k}=1, \mathbf{z})\, \mathrm{d}\mathbf{x}_{n,k}\,. \tag{18}$$

This Bayesian two-stage detection-estimation procedure is also employed by the JITS filter [9], the JIPDA filter [8], and certain FISST-based methods, e.g., [3], [20]. The main problem to be solved now is to find a computationally feasible recursive (sequential) calculation of $p(r_{n,k} = 1|\mathbf{z})$ and $f(\mathbf{x}_{n,k}|r_{n,k}=1, \mathbf{z})$.

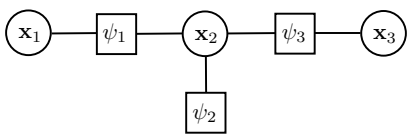

Fig. 1. Factor graph representing the factorization $f(\mathbf{x}|\mathbf{z}) \propto \psi_1(\mathbf{x}_1, \mathbf{x}_2)\, \psi_2(\mathbf{x}_2)\, \psi_3(\mathbf{x}_2, \mathbf{x}_3)$.

III. Review of BP

We briefly review factor graphs and the generic BP scheme, which constitute the main methodological basis of the proposed multisensor-multitarget tracking method. Consider the problem of estimating parameter vectors $\mathbf{x}_k$, $k \in \{1, \ldots, K\}$ from a measurement vector $\mathbf{z}$. Bayesian estimation of $\mathbf{x}_k$ relies on the posterior pdf $f(\mathbf{x}_k|\mathbf{z})$ [43]. This pdf is a marginal pdf of the joint posterior pdf $f(\mathbf{x}|\mathbf{z})$, where $\mathbf{x} = \big[\mathbf{x}_1^{\mathrm{T}} \cdots \mathbf{x}_K^{\mathrm{T}}\big]^{\mathrm{T}}$; however, direct marginalization of $f(\mathbf{x}|\mathbf{z})$ is usually infeasible. An efficient marginalization can be achieved if $f(\mathbf{x}|\mathbf{z})$ factorizes, i.e.,

$$f(\mathbf{x}|\mathbf{z}) \propto \prod_{q=1}^{Q} \psi_q\big(\mathbf{x}^{(q)}\big)\,. \tag{19}$$

Here, each factor argument $\mathbf{x}^{(q)}$ comprises certain parameter vectors $\mathbf{x}_k$ (each $\mathbf{x}_k$ can appear in several $\mathbf{x}^{(q)}$) and $\propto$ indicates equality up to a normalization factor. Note that for compactness, our notation does not indicate the dependence of the factors $\psi_q\big(\mathbf{x}^{(q)}\big)$ on $\mathbf{z}$.

The factorization structure (19) can be represented by a graphical model known as *factor graph*[3] [34]. As an example, for $\mathbf{x} = \big[\mathbf{x}_1^{\mathrm{T}}\, \mathbf{x}_2^{\mathrm{T}}\, \mathbf{x}_3^{\mathrm{T}}\big]^{\mathrm{T}}$, the factor graph representing the factorization $f(\mathbf{x}|\mathbf{z}) \propto \psi_1(\mathbf{x}_1, \mathbf{x}_2)\, \psi_2(\mathbf{x}_2)\, \psi_3(\mathbf{x}_2, \mathbf{x}_3)$ is shown in Fig. 1. In a factor graph, each parameter variable $\mathbf{x}_k$ is represented by a variable node and each factor $\psi_q(\cdot)$ by a factor node (depicted in Fig. 1 by a circle and a square, respectively). Variable node "$\mathbf{x}_k$" and factor node "ψ_q" are *adjacent*, i.e., connected by an edge, if the variable $\mathbf{x}_k$ is an argument of the factor $\psi_q(\cdot)$.

Belief propagation (BP), also known as the *sum-product algorithm* [31], is based on a factor graph and aims at computing the marginal posterior pdfs $f(\mathbf{x}_k|\mathbf{z})$ in an efficient way. For each node, certain messages are calculated, each of which is passed to one of the adjacent nodes. For each variable node, the incoming and outgoing messages are functions of the corresponding variable. More specifically, consider a variable node "$\mathbf{x}_k$" and an adjacent factor node "ψ_q", i.e., the variable $\mathbf{x}_k$ is part of the argument $\mathbf{x}^{(q)}$ of $\psi_q\big(\mathbf{x}^{(q)}\big)$. We define the neighborhood set $\mathcal{N}_q$ as the set of the indices k of all variable nodes "$\mathbf{x}_k$" adjacent to factor node "ψ_q". Then, the message passed from factor node "ψ_q" to variable node "$\mathbf{x}_k$" is given by

$$\zeta_{\psi_q \to \mathbf{x}_k}(\mathbf{x}_k) = \int \psi_q\big(\mathbf{x}^{(q)}\big) \prod_{k' \in \mathcal{N}_q \setminus \{k\}} \eta_{\mathbf{x}_{k'} \to \psi_q}(\mathbf{x}_{k'})\, \mathrm{d}\mathbf{x}_{\bar{k}}\,, \tag{20}$$

[3]We note that the factorization structure of probability distributions can alternatively be represented by other graphical models, such as Markov random fields [44].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSP.2017.2688966, IEEE Transactions on Signal Processing

where $\int \dots \mathrm{d}\mathbf{x}_{\bar{k}}$ denotes integration with respect to all vectors $\mathbf{x}_{k'} \in \mathcal{N}_q$ except $\mathbf{x}_k$, and the $\eta_{\mathbf{x}_{k'} \to \psi_q}(\mathbf{x}_{k'})$ will be explained presently. For example, the message passed from factor node "ψ_1" to variable node "$\mathbf{x}_2$" in Fig. 1 is $\zeta_{\psi_1 \to \mathbf{x}_2}(\mathbf{x}_2) = \int \psi_1(\mathbf{x}_1, \mathbf{x}_2) \eta_{\mathbf{x}_1 \to \psi_1}(\mathbf{x}_1) \mathrm{d}\mathbf{x}_1$; note that $\mathbf{x}^{(1)} = \left[\mathbf{x}_1^{\mathrm{T}}\, \mathbf{x}_2^{\mathrm{T}}\right]^{\mathrm{T}}$. The message $\eta_{\mathbf{x}_k \to \psi_q}(\mathbf{x}_k)$ passed from variable node "$\mathbf{x}_k$" to factor node "ψ_q" is given by the product of the messages passed to variable node "$\mathbf{x}_k$" from all adjacent factor nodes except "ψ_q". For example, in Fig. 1, the message passed from variable node "$\mathbf{x}_2$" to factor node "ψ_1" is $\eta_{\mathbf{x}_2 \to \psi_1}(\mathbf{x}_2) = \zeta_{\psi_2 \to \mathbf{x}_2}(\mathbf{x}_2) \zeta_{\psi_3 \to \mathbf{x}_2}(\mathbf{x}_2)$. Message passing is started at variable nodes with only one edge (which pass a constant message) and/or factor nodes with only one edge (which pass the corresponding factor). For example, in Fig. 1, the message from variable node "$\mathbf{x}_1$" to factor node "ψ_1" is $\eta_{\mathbf{x}_1 \to \psi_1}(\mathbf{x}_1) = C$, where C is an arbitrary nonzero constant. Note that BP can also be applied to factorizations involving discrete variables by replacing integration with summation in (20).

After all messages have been passed as described above, for each variable node "$\mathbf{x}_k$", a *belief* $\tilde{f}(\mathbf{x}_k)$ is calculated as the product of all incoming messages (passed from all adjacent factor nodes) followed by a normalization such that $\int \tilde{f}(\mathbf{x}_k) \mathrm{d}\mathbf{x}_k = 1$. For example, in Fig. 1,

$$\tilde{f}(\mathbf{x}_2) \propto \zeta_{\psi_1 \to \mathbf{x}_2}(\mathbf{x}_2) \zeta_{\psi_2 \to \mathbf{x}_2}(\mathbf{x}_2) \zeta_{\psi_3 \to \mathbf{x}_2}(\mathbf{x}_2). \tag{21}$$

If the factor graph is a tree, i.e., without loops, then the belief $\tilde{f}(\mathbf{x}_k)$ is exactly equal to the marginal posterior pdf $f(\mathbf{x}_k|\mathbf{z})$. For factor graphs with loops, BP is applied in an iterative manner, and the beliefs $\tilde{f}(\mathbf{x}_k)$ are only approximations of the respective marginal posterior pdfs $f(\mathbf{x}_k|\mathbf{z})$; these approximations have been observed to be very accurate in many applications [31]–[33]. In these iterative "loopy BP" schemes, there is no canonical order in which the messages should be calculated, and different orders may lead to different beliefs. The choice of an appropriate order of message calculation will be an important aspect in our development of the proposed method.

A useful modification of factor graphs and loopy BP schemes is suggested by the "opening factors" principle [32, Sec. 5.2.2]: by introducing additional variables that depend deterministically on certain variables in the factor graph, lower-dimensional messages can be obtained. This tends to result in improved scalability and reduced computational complexity, but often introduces additional loops in the factor graph. We will apply the "opening factors" principle to the factor $\psi\big(\mathbf{a}_n^{(s)}\big)$, as described next.

IV. Derivation of the Factor Graph

In this section, we derive the joint posterior pdf and, in turn, the factor graph underlying the proposed multitarget tracking algorithm.

A. Opening the Factor $\psi\big(\mathbf{a}_n^{(s)}\big)$

So far, we have described the possible PT-measurement associations at sensor s and time n in terms of the target-oriented association variables $a_{n,k}^{(s)}$ defined in (7). Hereafter, following [35] and [36], we also use an alternative description of the PT-measurement associations in terms of the *measurement-oriented association variables*

$$b_{n,m}^{(s)} \triangleq \begin{cases} k \in \mathcal{K}, & \text{if at time } n \text{, measurement } m \text{ at} \\ & \text{sensor } s \text{ is generated by PT } k \\ 0, & \text{if at time } n \text{, measurement } m \text{ at} \\ & \text{sensor } s \text{ is not generated by a PT.} \end{cases}$$

We also define $\mathbf{b}_n^{(s)} \triangleq \big[b_{n,1}^{(s)} \cdots b_{n,M_n^{(s)}}^{(s)}\big]^{\mathrm{T}}$ and $\mathbf{b}_n \triangleq \big[\mathbf{b}_n^{(1)\mathrm{T}} \cdots \mathbf{b}_n^{(S)\mathrm{T}}\big]^{\mathrm{T}}$.

The description in terms of $\mathbf{b}_n^{(s)}$ is redundant in that $\mathbf{b}_n^{(s)}$ can be derived from $\mathbf{a}_n^{(s)}$ and vice versa. However, this redundant formulation is the basis for opening the factor $\psi\big(\mathbf{a}_n^{(s)}\big)$, as discussed next. In fact, using the $\mathbf{b}_n^{(s)}$ alongside with the $\mathbf{a}_n^{(s)}$, the exclusion-enforcing function $\psi\big(\mathbf{a}_n^{(s)}\big)$ in (15) can be formally replaced by the function [35], [36]

$$\psi\big(\mathbf{a}_n^{(s)}, \mathbf{b}_n^{(s)}\big) = \prod_{k=1}^{K} \prod_{m=1}^{M_n^{(s)}} \Psi\big(a_{n,k}^{(s)}, b_{n,m}^{(s)}\big), \tag{22}$$

with

$$\Psi\big(a_{n,k}^{(s)}, b_{n,m}^{(s)}\big) \triangleq \begin{cases} 0, & a_{n,k}^{(s)} = m,\ b_{n,m}^{(s)} \neq k \\ & \quad \text{or } b_{n,m}^{(s)} = k,\ a_{n,k}^{(s)} \neq m \\ 1, & \text{otherwise.} \end{cases}$$

Defining $\mathbf{b} \triangleq \big[\mathbf{b}_1^{\mathrm{T}} \cdots \mathbf{b}_n^{\mathrm{T}}\big]^{\mathrm{T}}$, the prior pmf $p(\mathbf{a}, \mathbf{m}|\mathbf{y})$ in (13) can now be formally rewritten as

$$p(\mathbf{a}, \mathbf{b}, \mathbf{m}|\mathbf{y}) = \prod_{n'=1}^{n} \prod_{s=1}^{S} p\big(\mathbf{a}_{n'}^{(s)}, \mathbf{b}_{n'}^{(s)}, M_{n'}^{(s)} \big| \mathbf{y}_{n'}\big), \tag{23}$$

with the single-sensor prior pmfs (cf. (16) and (22))

$$p\big(\mathbf{a}_n^{(s)}, \mathbf{b}_n^{(s)}, M_n^{(s)} \big| \mathbf{y}_n\big) = C\big(M_n^{(s)}\big) \prod_{k=1}^{K} h\big(\mathbf{x}_{n,k}, r_{n,k}, a_{n,k}^{(s)}; M_n^{(s)}\big) \prod_{m=1}^{M_n^{(s)}} \Psi\big(a_{n,k}^{(s)}, b_{n,m}^{(s)}\big).$$

Thus, Equation (23) can be expressed as

$$p(\mathbf{a}, \mathbf{b}, \mathbf{m}|\mathbf{y}) = C(\mathbf{m}) \prod_{n'=1}^{n} \prod_{s=1}^{S} \prod_{k=1}^{K} h\big(\mathbf{x}_{n',k}, r_{n',k}, a_{n',k}^{(s)}; M_{n'}^{(s)}\big) \times \prod_{m=1}^{M_{n'}^{(s)}} \Psi\big(a_{n',k}^{(s)}, b_{n',m}^{(s)}\big), \tag{24}$$

where $C(\mathbf{m})$ is a normalization factor depending only on $\mathbf{m}$. Note that the factorization in (24) is more "detailed" than our original factorization in (13) and (14) because the high-dimensional factors $\psi\big(\mathbf{a}_n^{(s)}\big)$ have been opened, i.e., replaced by the many low-dimensional factors $\Psi\big(a_{n,k}^{(s)}, b_{n,m}^{(s)}\big)$. This detailed factorization constitutes an important basis for our development of the proposed algorithm in Section V.

B. Joint Posterior pdf and Factor Graph

The marginal posterior existence probability $p(r_{n,k} = 1|\mathbf{z})$ underlying target detection as discussed in Section II-D can

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSP.2017.2688966, IEEE Transactions on Signal Processing

be obtained from the marginal posterior pdf of the augmented target state, $f(\mathbf{y}_{n,k}|\mathbf{z}) = f(\mathbf{x}_{n,k}, r_{n,k}|\mathbf{z})$, according to

$$p(r_{n,k}=1|\mathbf{z}) = \int f(\mathbf{x}_{n,k}, r_{n,k}=1|\mathbf{z})\,\mathrm{d}\mathbf{x}_{n,k}\,, \qquad (25)$$

and the marginal posterior pdf $f(\mathbf{x}_{n,k}|r_{n,k}=1,\mathbf{z})$ underlying MMSE state estimation (see (18)) can be obtained from $f(\mathbf{x}_{n,k}, r_{n,k}|\mathbf{z})$ according to

$$f(\mathbf{x}_{n,k}|r_{n,k}=1,\mathbf{z}) = \frac{f(\mathbf{x}_{n,k}, r_{n,k}=1|\mathbf{z})}{p(r_{n,k}=1|\mathbf{z})}\,. \qquad (26)$$

Thus, the underlying task is to compute the pdf $f(\mathbf{y}_{n,k}|\mathbf{z}) = f(\mathbf{x}_{n,k}, r_{n,k}|\mathbf{z})$. This pdf is a marginal density of the joint posterior pdf $f(\mathbf{y}, \mathbf{a}, \mathbf{b}|\mathbf{z})$, which involves all the augmented states, all the target-oriented and measurement-oriented association variables, and all the measurements of all sensors, at all times up to the current time n. As explained in Section III, an efficient approximate implementation of the marginalization converting $f(\mathbf{y}, \mathbf{a}, \mathbf{b}|\mathbf{z})$ into $f(\mathbf{y}_{n,k}|\mathbf{z})$ can be obtained by performing BP on a factor graph that expresses the factorization of the joint posterior pdf $f(\mathbf{y}, \mathbf{a}, \mathbf{b}|\mathbf{z})$.

In the following derivation of the factorization of $f(\mathbf{y}, \mathbf{a}, \mathbf{b}|\mathbf{z})$, the measurements $\mathbf{z}$ are considered observed and thus fixed, and consequently the $M_n^{(s)}$ and $\mathbf{m}$ are fixed as well. Then, using Bayes' rule and the fact that $\mathbf{a}$ implies $\mathbf{b}$ and $\mathbf{z}$ implies $\mathbf{m}$, we obtain

$$\begin{aligned} f(\mathbf{y}, \mathbf{a}, \mathbf{b}|\mathbf{z}) &= f(\mathbf{y}, \mathbf{a}, \mathbf{b}, \mathbf{m}|\mathbf{z}) \\ &\propto f(\mathbf{z}|\mathbf{y}, \mathbf{a}, \mathbf{b}, \mathbf{m})\, f(\mathbf{y}, \mathbf{a}, \mathbf{b}, \mathbf{m}) \\ &= f(\mathbf{z}|\mathbf{y}, \mathbf{a}, \mathbf{m})\, p(\mathbf{a}, \mathbf{b}, \mathbf{m}|\mathbf{y})\, f(\mathbf{y})\,. \end{aligned}$$

Inserting (4) for $f(\mathbf{y})$, (12) for $f(\mathbf{z}|\mathbf{y}, \mathbf{a}, \mathbf{m})$, and (24) for $p(\mathbf{a}, \mathbf{b}, \mathbf{m}|\mathbf{y})$ then yields the final factorization

$$\begin{aligned} f(\mathbf{y}, \mathbf{a}, \mathbf{b}|\mathbf{z}) \propto \prod_{k=1}^{K} f(\mathbf{y}_{0,k}) \prod_{n'=1}^{n} f(\mathbf{y}_{n',k}|\mathbf{y}_{n'-1,k}) \\ \times \prod_{s=1}^{S} \upsilon\big(\mathbf{y}_{n',k}, a_{n',k}^{(s)}; \mathbf{z}_{n'}^{(s)}\big) \prod_{m=1}^{M_{n'}^{(s)}} \Psi\big(a_{n',k}^{(s)}, b_{n',m}^{(s)}\big)\,, \end{aligned} \qquad (27)$$

with

$$\begin{aligned} \upsilon\big(\mathbf{y}_{n,k}, a_{n,k}^{(s)}; \mathbf{z}_n^{(s)}\big) \triangleq\; & g\big(\mathbf{x}_{n,k}, r_{n,k}, a_{n,k}^{(s)}; \mathbf{z}_n^{(s)}\big) \\ & \times h\big(\mathbf{x}_{n,k}, r_{n,k}, a_{n,k}^{(s)}; M_n^{(s)}\big)\,. \end{aligned} \qquad (28)$$

The factor graph representing this factorization as explained in Section III is depicted for one time step in Fig. 2.

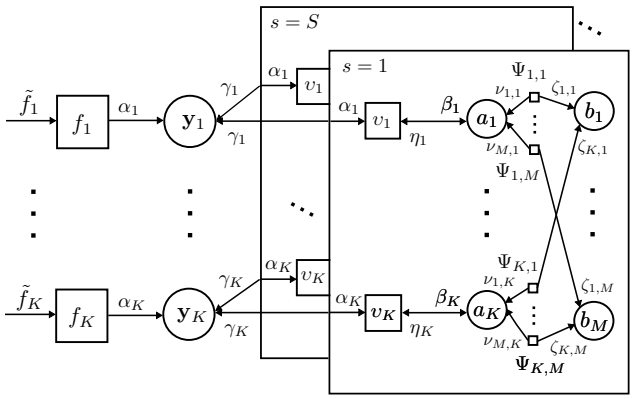

Fig. 2. Factor graph representing the factorization of the joint posterior pdf $f(\mathbf{y}, \mathbf{a}, \mathbf{b}|\mathbf{z})$ in (27), depicted for one time step. For simplicity, the time index n and sensor index s are omitted, and the following short notations are used: $f_k \triangleq f(\mathbf{y}_{n,k}|\mathbf{y}_{n-1,k})$, $\upsilon_k \triangleq \upsilon\big(\mathbf{y}_{n,k}, a_{n,k}^{(s)}; \mathbf{z}_n^{(s)}\big)$, $\tilde{f}_k \triangleq \tilde{f}(\mathbf{y}_{n,k})$, $\Psi_{k,m} \triangleq \Psi\big(a_{n,k}^{(s)}, b_{n,m}^{(s)}\big)$, $\alpha_k \triangleq \alpha(\mathbf{y}_{n,k})$, $\beta_k \triangleq \beta\big(a_{n,k}^{(s)}\big)$, $\eta_k \triangleq \eta\big(a_{n,k}^{(s)}\big)$, $\gamma_k \triangleq \gamma^{(s)}(\mathbf{y}_{n,k})$, $\nu_{m,k} \triangleq \nu_{m\to k}^{(p)}\big(a_{n,k}^{(s)}\big)$, and $\zeta_{k,m} \triangleq \zeta_{k\to m}^{(p)}\big(b_{n,m}^{(s)}\big)$. The messages $\alpha(\mathbf{y}_{n,k})$ etc. are described in Section V-A.

V. The Proposed BP-based Multisensor-Multitarget Tracking Algorithm

In this section, we present the proposed BP-based multisensor-multitarget tracking algorithm and investigate its scalability.

A. BP Method

As discussed in Section III, approximations $\tilde{f}(\mathbf{y}_{n,k})$ of the marginal posterior pdfs $f(\mathbf{y}_{n,k}|\mathbf{z})$ can be obtained in an efficient way by running iterative BP on the factor graph in Fig. 2. Since this factor graph is loopy, we have to decide on a specific order of message computation. We choose this order according to the following rules: (i) Messages are not sent backward in time[4] [45]–[47]. (ii) Iterative message passing is only performed at each time step and at each sensor separately—i.e., in particular, for the loops connecting different sensors we perform only a single message passing iteration—and only for data association. (iii) Measurements from different sensors are processed in parallel. Note that the first rule enables real-time processing, the second rule minimizes the repeated use of information, and the third rule avoids sequential processing of information from different sensors (as is done in IC methods). In addition, the first two rules yield a reduction of computational complexity. With these rules, the generic BP rules for calculating messages and beliefs as summarized in Section III yield the following BP operations at time n.

First, a *prediction* step is performed for all PTs $k \in \mathcal{K}$, i.e.,

$$\begin{aligned} \alpha(\mathbf{x}_{n,k}, r_{n,k}) = \sum_{r_{n-1,k}\in\{0,1\}} \int & f(\mathbf{x}_{n,k}, r_{n,k}|\mathbf{x}_{n-1,k}, r_{n-1,k}) \\ & \times \tilde{f}(\mathbf{x}_{n-1,k}, r_{n-1,k})\,\mathrm{d}\mathbf{x}_{n-1,k}\,. \end{aligned}$$

Here, $\tilde{f}(\mathbf{x}_{n-1,k}, r_{n-1,k})$ was calculated at the previous time n–1. Inserting (5) and (6) for $f(\mathbf{x}_{n,k}, r_{n,k}|\mathbf{x}_{n-1,k}, r_{n-1,k})$, we obtain the following expressions of $\alpha(\mathbf{x}_{n,k}, r_{n,k})$: for $r_{n,k} = 1$,

$$\begin{aligned} \alpha(\mathbf{x}_{n,k}, 1) = {} & p_{n,k}^{\mathrm{b}}\, f_{\mathrm{b}}(\mathbf{x}_{n,k})\, \tilde{f}_{n-1,k} + p_{n,k}^{\mathrm{s}} \\ & \times \int f(\mathbf{x}_{n,k}|\mathbf{x}_{n-1,k}) \tilde{f}(\mathbf{x}_{n-1,k}, 1)\,\mathrm{d}\mathbf{x}_{n-1,k}\,, \end{aligned} \qquad (29)$$

and for $r_{n,k} = 0$,

$$\alpha_{n,k} = \big(1 - p_{n,k}^{\mathrm{b}}\big)\tilde{f}_{n-1,k} + \big(1 - p_{n,k}^{\mathrm{s}}\big)\int \tilde{f}(\mathbf{x}_{n-1,k}, 1)\,\mathrm{d}\mathbf{x}_{n-1,k}\,. \qquad (30)$$

[4]This is equivalent to the assumption that the target states are conditionally independent given the past measurements. This assumption is also made in the derivation of the JPDA filter [2].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSP.2017.2688966, IEEE Transactions on Signal Processing

We note that $\tilde{f}_{n-1,k} = \int \tilde{f}(\mathbf{x}_{n-1,k}, 0)\,\mathrm{d}\mathbf{x}_{n-1,k}$ and $\alpha_{n,k} = \int \alpha(\mathbf{x}_{n,k}, 0)\,\mathrm{d}\mathbf{x}_{n,k}$ (cf. (3)). Furthermore, since $\tilde{f}(\mathbf{x}_{n-1,k}, r_{n-1,k})$ is normalized, so is $\alpha(\mathbf{x}_{n,k}, r_{n,k})$, i.e., $\sum_{r_{n,k}\in\{0,1\}} \int \alpha(\mathbf{x}_{n,k}, r_{n,k})\,\mathrm{d}\mathbf{x}_{n,k} = 1$. Thus, we have $\alpha_{n,k} = 1 - \int \alpha(\mathbf{x}_{n,k}, 1)\,\mathrm{d}\mathbf{x}_{n,k}$.

After the prediction step, the following steps are performed for all PTs $k \in \mathcal{K}$ and all sensors $s \in \mathcal{S}$ in parallel:

1) *Measurement evaluation*:

$$\begin{aligned}\beta\big(a_{n,k}^{(s)}\big) &= \sum_{r_{n,k}\in\{0,1\}} \int \upsilon\big(\mathbf{x}_{n,k}, r_{n,k}, a_{n,k}^{(s)}; \mathbf{z}_n^{(s)}\big) \\ &\qquad \times \alpha(\mathbf{x}_{n,k}, r_{n,k})\,\mathrm{d}\mathbf{x}_{n,k} \\ &= \int \upsilon\big(\mathbf{x}_{n,k}, 1, a_{n,k}^{(s)}; \mathbf{z}_n^{(s)}\big)\,\alpha(\mathbf{x}_{n,k}, 1)\,\mathrm{d}\mathbf{x}_{n,k} \\ &\qquad + 1\big(a_{n,k}^{(s)}\big)\,\alpha_{n,k}\,. \end{aligned} \tag{31}$$

In the last expression, we used $\upsilon\big(\mathbf{x}_{n,k}, 0, a_{n,k}^{(s)}; \mathbf{z}_n^{(s)}\big) = 1\big(a_{n,k}^{(s)}\big)$, which follows from (28) with (11) and (17).

2) *Iterative data association* (this part of the BP method closely follows [35], [36], [48]). In iteration $p \in \{1, \ldots, P\}$, the following calculations are performed for all measurements $m \in \mathcal{M}_n^{(s)}$:

$$\begin{aligned}\nu_{m\to k}^{(p)}\big(a_{n,k}^{(s)}\big) &= \sum_{b_{n,m}^{(s)}=0}^{K} \Psi\big(a_{n,k}^{(s)}, b_{n,m}^{(s)}\big) \\ &\qquad \times \prod_{k'\in\mathcal{K}\setminus\{k\}} \zeta_{k'\to m}^{(p-1)}\big(b_{n,m}^{(s)}\big)\end{aligned} \tag{32}$$

and (except for $p = P$)

$$\begin{aligned}\zeta_{k\to m}^{(p)}\big(b_{n,m}^{(s)}\big) &= \sum_{a_{n,k}^{(s)}=0}^{M_n^{(s)}} \beta\big(a_{n,k}^{(s)}\big)\,\Psi\big(a_{n,k}^{(s)}, b_{n,m}^{(s)}\big) \\ &\qquad \times \prod_{m'\in\mathcal{M}_n^{(s)}\setminus\{m\}} \nu_{m'\to k}^{(p)}\big(a_{n,k}^{(s)}\big)\,.\end{aligned} \tag{33}$$

The operations (32) and (33) constitute an iteration loop, which is initialized (for $p = 0$) by

$$\zeta_{k\to m}^{(0)}\big(b_{n,m}^{(s)}\big) = \sum_{a_{n,k}^{(s)}=0}^{M_n^{(s)}} \beta\big(a_{n,k}^{(s)}\big)\,\Psi\big(a_{n,k}^{(s)}, b_{n,m}^{(s)}\big)\,. \tag{34}$$

An efficient implementation of (32) and (33) is described in [36], [48], and Matlab code for that implementation is provided in [49]. After the last iteration $p = P$, the messages $\nu_{m\to k}^{(P)}\big(a_{n,k}^{(s)}\big)$, $m \in \mathcal{M}_n^{(s)}$ are multiplied, i.e.,

$$\eta\big(a_{n,k}^{(s)}\big) = \prod_{m=1}^{M_n^{(s)}} \nu_{m\to k}^{(P)}\big(a_{n,k}^{(s)}\big)\,. \tag{35}$$

3) *Measurement update*:

$$\begin{aligned}\gamma^{(s)}(\mathbf{x}_{n,k}, 1) &= \sum_{a_{n,k}^{(s)}=0}^{M_n^{(s)}} \upsilon\big(\mathbf{x}_{n,k}, 1, a_{n,k}^{(s)}; \mathbf{z}_n^{(s)}\big)\,\eta\big(a_{n,k}^{(s)}\big) \\ \gamma_{n,k}^{(s)} &= \eta\big(a_{n,k}^{(s)} = 0\big)\,.\end{aligned} \tag{36}$$

Finally, *beliefs* $\tilde{f}(\mathbf{y}_{n,k}) = \tilde{f}(\mathbf{x}_{n,k}, r_{n,k})$ approximating the marginal posterior pdfs $f(\mathbf{y}_{n,k}|\mathbf{z}) = f(\mathbf{x}_{n,k}, r_{n,k}|\mathbf{z})$ are obtained as the products

$$\tilde{f}(\mathbf{x}_{n,k}, 1) = \frac{1}{C_{n,k}}\,\alpha(\mathbf{x}_{n,k}, 1) \prod_{s=1}^{S} \gamma^{(s)}(\mathbf{x}_{n,k}, 1) \tag{37}$$

$$\tilde{f}_{n,k} = \frac{1}{C_{n,k}}\,\alpha_{n,k} \prod_{s=1}^{S} \gamma_{n,k}^{(s)}\,, \tag{38}$$

with the normalization constants

$$C_{n,k} = \int \alpha(\mathbf{x}_{n,k}, 1) \prod_{s=1}^{S} \gamma^{(s)}(\mathbf{x}_{n,k}, 1)\,\mathrm{d}\mathbf{x}_{n,k} + \alpha_{n,k} \prod_{s=1}^{S} \gamma_{n,k}^{(s)}\,. \tag{39}$$

Note that $f_{\mathrm{D}}^2(\mathbf{x}_{n,k}) = f_{\mathrm{D}}(\mathbf{x}_{n,k})$ (cf. Section II-A) was used to obtain (38). Because the belief $\tilde{f}(\mathbf{x}_{n,k}, 1)$ in (37) approximates $f(\mathbf{x}_{n,k}, r_{n,k} = 1|\mathbf{z})$, it can be substituted for $f(\mathbf{x}_{n,k}, r_{n,k} = 1|\mathbf{z})$ in (25) and (26), thus providing the basis for Bayesian target detection and state estimation as discussed in Section II-D. A particle-based implementation of the above BP method that avoids the explicit evaluation of integrals and message products will be presented in Section VI.

The "data association" step (32)–(35) involves solely messages related to discrete random variables. Being based on loopy BP, it does not perform an exact marginalization [31], [33], [34]. However, its high accuracy has been demonstrated numerically [35], [36] (see also Section IX-B), and its convergence has been proven [36], [48]. We also note that the result of the proposed BP method does not depend on the order in which the measurements $\mathbf{z}_{n,m}^{(s)}$, $m \in \mathcal{M}_n^{(s)}$ appear in the sensor measurement vector $\mathbf{z}_n^{(s)}$, the order in which the $\mathbf{z}_n^{(s)}$, $s \in \mathcal{S}$ appear in the stacked measurement vector $\mathbf{z}_n$, or the order in which the augmented target states $\mathbf{y}_{n,k}$, $k \in \mathcal{K}$ appear in the vector $\mathbf{y}_n$.

B. Scalability

The main advantage of the BP method described in Section V-A is its scalability. Assuming a fixed number P of message passing iterations, the computational complexity of calculating the (approximate) marginal posterior pdfs of all the target states is only linear in the number of sensors S (see (37), (38)). Moreover, the complexity of the operations (31)–(36) performed for a given sensor $s \in \mathcal{S}$ scales as $\mathcal{O}\big(KM_n^{(s)}\big)$, where $M_n^{(s)}$ increases linearly with the number of PTs K and with the number of false alarms. (Here, it is assumed that the efficient implementation of (32), (33) described in [36], [48] is used.) Thus, the overall complexity of our method scales linearly in the number of sensors and quadratically in the number of PTs. Note that in practical implementations, measurement gating [2] can be used to further improve the scalability.

Such favorable scaling is a consequence of the "detailed" factorization (27). This factorization, in turn, is based on the redundant formulation of the joint state-association estimation task and the corresponding opening of the factor $\psi\big(\mathbf{a}_n^{(s)}\big)$ described in Section IV-A. Due to this factor opening, an increase in the number of PTs, sensors, or measurements leads

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSP.2017.2688966, IEEE Transactions on Signal Processing

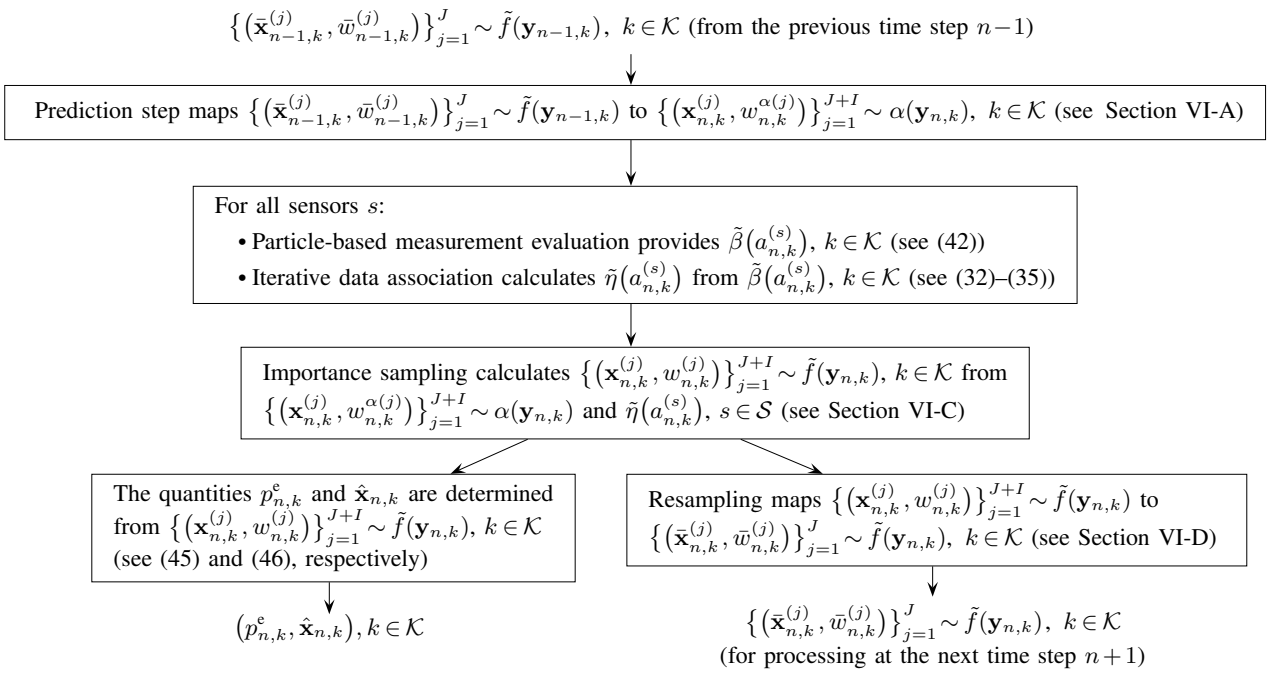

Fig. 3. Flowchart of a particle-based implementation of the proposed BP method. The symbol $\sim$ expresses the fact that a set of particles and weights represents a certain distribution.

to additional variable nodes in the factor graph (see Fig. 2) but not to higher dimensions of the messages passed between the nodes. The scalability of the proposed algorithm will be further analyzed in Section IX-C.

VI. Particle-based Implementation

For general nonlinear and non-Gaussian measurement and state evolution models, the integrals in (18), (29), (30), and (31) as well as the message products in (37) and (38) typically cannot be evaluated in closed form and are computationally infeasible. Therefore, we next present an approximate particle-based implementation of these operations. In this implementation, each belief $\tilde{f}(\mathbf{x}_{n,k}, r_{n,k})$ is represented by a set of particles and corresponding weights $\{(\bar{\mathbf{x}}_{n,k}^{(j)}, \bar{w}_{n,k}^{(j)})\}_{j=1}^{J}$. More specifically, $\tilde{f}(\mathbf{x}_{n,k}, 1)$ is represented by $\{(\bar{\mathbf{x}}_{n,k}^{(j)}, \bar{w}_{n,k}^{(j)})\}_{j=1}^{J}$, and $\tilde{f}(\mathbf{x}_{n,k}, 0) = \tilde{f}_{n,k} f_{\mathrm{D}}(\mathbf{x}_{n,k})$ is given implicitly by the normalization property of $\tilde{f}(\mathbf{x}_{n,k}, r_{n,k})$, i.e., $\tilde{f}_{n,k} = 1 - \int \tilde{f}(\mathbf{x}_{n,k}, 1)\,\mathrm{d}\mathbf{x}_{n,k}$. Contrary to conventional particle filtering [50], [51], the particle weights $\bar{w}_{n,k}^{(j)}$, $j \in \{1, \ldots, J\}$ do not sum to one; instead,

$$p_{n,k}^{\mathrm{e}} \triangleq \sum_{j=1}^{J} \bar{w}_{n,k}^{(j)} \approx \int \tilde{f}(\mathbf{x}_{n,k}, 1)\,\mathrm{d}\mathbf{x}_{n,k}\,. \tag{40}$$

Note that since $\int \tilde{f}(\mathbf{x}_{n,k}, 1)\,\mathrm{d}\mathbf{x}_{n,k}$ approximates the posterior probability of target existence $p(r_{n,k}=1|\mathbf{z})$, it follows that the sum of weights $p_{n,k}^{\mathrm{e}}$ is approximately equal to $p(r_{n,k}=1|\mathbf{z})$.

The particle operations described in the remainder of this section are performed for all PTs $k \in \mathcal{K}$ in parallel. The resulting particle-based implementation of the overall BP method is summarized in Fig. 3.

A. Prediction

For $n \geq 1$ and for each PT $k \in \mathcal{K}$, J particles and weights $\{(\bar{\mathbf{x}}_{n-1,k}^{(j)}, \bar{w}_{n-1,k}^{(j)} = p_{n-1,k}^{\mathrm{e}}/J)\}_{j=1}^{J}$ representing the previous belief $\tilde{f}(\mathbf{x}_{n-1,k}, r_{n-1,k})$ were calculated at the previous time $n-1$ as described further below. Weighted particles $\{(\mathbf{x}_{n,k}^{(j)}, w_{n,k}^{\alpha(j)})\}_{j=1}^{J+I}$ representing the message $\alpha(\mathbf{x}_{n,k}, 1)$ in (29) are now obtained as follows.[5] First, for each particle $\bar{\mathbf{x}}_{n-1,k}^{(j)}$, $j \in \{1, \ldots, J\}$, one particle $\mathbf{x}_{n,k}^{(j)}$ is drawn from $f(\mathbf{x}_{n,k}|\bar{\mathbf{x}}_{n-1,k}^{(j)})$. Next, I additional "birth particles" $\{\mathbf{x}_{n,k}^{(j)}\}_{j=J+1}^{J+I}$ are drawn from the birth pdf $f_{\mathrm{b}}(\mathbf{x}_{n,k})$. Finally, weights $w_{n,k}^{\alpha(j)}$, $j \in \{1, \ldots, J+I\}$ are obtained as

$$w_{n,k}^{\alpha(j)} = \begin{cases} p_{n,k}^{\mathrm{s}} \bar{w}_{n-1,k}^{(j)}, & j \in \{1, \ldots, J\} \\ p_{n,k}^{\mathrm{b}}(1 - p_{n-1,k}^{\mathrm{e}})/I, & j \in \{J+1, \ldots, J+I\}. \end{cases} \tag{41}$$

Here, $p_{n-1,k}^{\mathrm{e}} = \sum_{j=1}^{J} \bar{w}_{n-1,k}^{(j)}$ (cf. (40)). Note that the proposal distribution [50], [51] underlying (41) is $f(\mathbf{x}_{n,k}|\bar{\mathbf{x}}_{n-1,k}^{(j)})$ for $j \in \{1, \ldots, J\}$ and $f_{\mathrm{b}}(\mathbf{x}_{n,k})$ for $j \in \{J+1, \ldots, J+I\}$. A more general expression for particle-based prediction with an arbitrary proposal distribution can be found in [16].

B. Measurement Evaluation

For each sensor $s \in \mathcal{S}$, an approximation $\tilde{\beta}(a_{n,k}^{(s)})$ of the message $\beta(a_{n,k}^{(s)})$ in (31) can be calculated from the weighted particles $\{(\mathbf{x}_{n,k}^{(j)}, w_{n,k}^{\alpha(j)})\}_{j=1}^{J+I}$ representing $\alpha(\mathbf{x}_{n,k}, r_{n,k})$ as

$$\tilde{\beta}(a_{n,k}^{(s)}) = \sum_{j=1}^{J+I} \upsilon(\mathbf{x}_{n,k}^{(j)}, 1, a_{n,k}^{(s)}; \mathbf{z}_n^{(s)})\, w_{n,k}^{\alpha(j)}$$

[5] Note that $\alpha_{n,k}$ in (30) is again given implicitly by these weighted particles since $\alpha(\mathbf{x}_{n,k}, r_{n,k})$ is normalized.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSP.2017.2688966, IEEE Transactions on Signal Processing

$$+\,1\big(a_{n,k}^{(s)}\big)\left(1-\sum_{j=1}^{J+I} w_{n,k}^{\alpha(j)}\right). \qquad (42)$$

Here, $\sum_{j=1}^{J+I} \upsilon\big(\mathbf{x}_{n,k}^{(j)},1,a_{n,k}^{(s)};\mathbf{z}_n^{(s)}\big)\, w_{n,k}^{\alpha(j)}$ provides a Monte Carlo approximation [51] of $\int \upsilon\big(\mathbf{x}_{n,k},1,a_{n,k}^{(s)};\mathbf{z}_n^{(s)}\big)\,\alpha(\mathbf{x}_{n,k},1)\times \mathrm{d}\mathbf{x}_{n,k}$ in (31), and $1\big(a_{n,k}^{(s)}\big)\big(1-\sum_{j=1}^{J+I} w_{n,k}^{\alpha(j)}\big)$ provides an approximation of $1\big(a_{n,k}^{(s)}\big)\,\alpha_{n,k}$ in (31). Note that $\sum_{j=1}^{J+I} w_{n,k}^{\alpha(j)}$ can be interpreted as a "predicted existence probability," and thus $1-\sum_{j=1}^{J+I} w_{n,k}^{\alpha(j)}$ can be interpreted as a "predicted nonexistence probability," which approximates $\alpha_{n,k}$.

C. Data Association, Measurement Update, Belief Calculation

The approximate messages $\tilde{\beta}\big(a_{n,k}^{(s)}\big)$ obtained in (42) are used in the iterative data association step, i.e., they are substituted for the messages $\beta\big(a_{n,k}^{(s)}\big)$ in (33) and (34). After convergence of the data association iteration, approximate messages $\tilde{\eta}\big(a_{n,k}^{(s)}\big)$ (approximating the messages $\eta\big(a_{n,k}^{(s)}\big)$ in (35)) are then available for all PTs k and all sensors s.

Next, the measurement update step (36) and the belief calculation step (37), (38) are implemented by means of importance sampling [50], [51]. To that end, we first rewrite the belief $\tilde{f}(\mathbf{x}_{n,k},r_{n,k})$ in (37), (38) by inserting (36), i.e.,

$$\tilde{f}(\mathbf{x}_{n,k},1) \propto \alpha(\mathbf{x}_{n,k},1)\prod_{s=1}^{S}\sum_{a_{n,k}^{(s)}=0}^{M_n^{(s)}} \upsilon\big(\mathbf{x}_{n,k},1,a_{n,k}^{(s)};\mathbf{z}_n^{(s)}\big)\,\tilde{\eta}\big(a_{n,k}^{(s)}\big) \qquad (43)$$

$$\tilde{f}_{n,k} \propto \alpha_{n,k}\prod_{s=1}^{S}\tilde{\eta}\big(a_{n,k}^{(s)}=0\big). \qquad (44)$$

Here, we also replaced $\eta\big(a_{n,k}^{(s)}\big)$ by its particle-based approximation $\tilde{\eta}\big(a_{n,k}^{(s)}\big)$, even though we do not indicate this additional approximation in our notation $\tilde{f}(\mathbf{x}_{n,k},r_{n,k})$. We now calculate nonnormalized weights corresponding to (43) as

$$w_{n,k}^{\mathrm{A}(j)} = w_{n,k}^{\alpha(j)}\prod_{s=1}^{S}\sum_{a_{n,k}^{(s)}=0}^{M_n^{(s)}} \upsilon\big(\mathbf{x}_{n,k}^{(j)},1,a_{n,k}^{(s)};\mathbf{z}_n^{(s)}\big)\,\tilde{\eta}\big(a_{n,k}^{(s)}\big), \quad j\in\{1,\dots,J+I\}.$$

Note that this expression is based on importance sampling with proposal density $\alpha(\mathbf{x}_{n,k},1)$ (represented by the weighted particles $\big\{\big(\mathbf{x}_{n,k}^{(j)},w_{n,k}^{\alpha(j)}\big)\big\}_{j=1}^{J+I}$). Similarly, we calculate a single nonnormalized weight corresponding to (44) as

$$w_{n,k}^{\mathrm{B}} = \left(1-\sum_{j=1}^{J+I} w_{n,k}^{\alpha(j)}\right)\prod_{s=1}^{S}\tilde{\eta}\big(a_{n,k}^{(s)}=0\big),$$

in which $1-\sum_{j=1}^{J+I} w_{n,k}^{\alpha(j)}$ approximates $\alpha_{n,k}$.

Next, weighted particles $\big\{\big(\mathbf{x}_{n,k}^{(j)},w_{n,k}^{(j)}\big)\big\}_{j=1}^{J+I}$ representing the belief $\tilde{f}(\mathbf{x}_{n,k},r_{n,k})$ are obtained by using the particles $\big\{\mathbf{x}_{n,k}^{(j)}\big\}_{j=1}^{J+I}$ representing $\alpha(\mathbf{x}_{n,k},r_{n,k})$ and calculating the corresponding weights as

$$w_{n,k}^{(j)} = \frac{w_{n,k}^{\mathrm{A}(j)}}{\sum_{j'=1}^{J+I} w_{n,k}^{\mathrm{A}(j')} + w_{n,k}^{\mathrm{B}}}.$$

Here, $\sum_{j=1}^{J+I} w_{n,k}^{\mathrm{A}(j)} + w_{n,k}^{\mathrm{B}}$ is a particle-based approximation of the normalization constant $C_{n,k}$ in (39). We note that

$$p_{n,k}^{\mathrm{e}} = \sum_{j=1}^{J+I} w_{n,k}^{(j)}. \qquad (45)$$

D. Target Detection, State Estimation, Resampling

The weighted particles $\big\{\big(\mathbf{x}_{n,k}^{(j)},w_{n,k}^{(j)}\big)\big\}_{j=1}^{J+I}$ can now be used for target detection and estimation. First, for each PT k, an approximation $p_{n,k}^{\mathrm{e}}$ of the existence probability $p(r_{n,k}=1|\mathbf{z})$ is calculated from the particle weights $\big\{w_{n,k}^{(j)}\big\}_{j=1}^{J+I}$ according to (45). PT k is then detected (i.e., considered to exist) if $p_{n,k}^{\mathrm{e}}$ is above a threshold P_{th} (cf. Section II-D). For the detected targets k, an approximation of the MMSE state estimate $\hat{\mathbf{x}}_{n,k}^{\mathrm{MMSE}}$ in (18) is calculated according to

$$\hat{\mathbf{x}}_{n,k} = \frac{1}{p_{n,k}^{\mathrm{e}}}\sum_{j=1}^{J+I} w_{n,k}^{(j)}\,\mathbf{x}_{n,k}^{(j)}. \qquad (46)$$

Finally, as a preparation for the next time step $n+1$, a resampling step [50], [51] is performed to reduce the number of particles to J and to avoid degeneracy effects. The resampling results in equally weighted particles $\big\{\bar{\mathbf{x}}_{n,k}^{(j)}\big\}_{j=1}^{J}$; the corresponding weights are given by $\bar{w}_{n,k}^{(j)} = \bar{w}_{n,k} = \frac{1}{J}\sum_{j'=1}^{I+J} w_{n,k}^{(j')}$, $j\in\{1,\dots,J\}$. We note that as a consequence, $p_{n,k}^{\mathrm{e}}$ in (45) can also be expressed as in (40).

VII. Choice of Birth and Survival Parameters

Next, we present a scheme for choosing the birth pdfs $f_{\mathrm{b}}(\mathbf{x}_{n,k})$, birth probabilities $p_{n,k}^{\mathrm{b}}$, and survival probabilities $p_{n,k}^{\mathrm{s}}$ (see Section II-A). This scheme is inspired by the standard Poisson birth model, where the number of newly born targets obeys a Poisson distribution with mean μ^{b} and existing targets survive with a fixed, specified probability p^{s} [3]. The scheme is heuristic but results in scalability with respect to the number of sensors S and, as demonstrated in Section IX-B, in good detection and tracking performance.

We first distinguish between "reliable" and "unreliable" PTs at time $n-1$ by comparing the PT existence probabilities $p_{n-1,k}^{\mathrm{e}}$ with a reliability threshold R_{th}: PT k is considered reliable at time $n-1$ if $p_{n-1,k}^{\mathrm{e}} > R_{\mathrm{th}}$ and unreliable otherwise. Let $\mathcal{K}_{n-1}^{\mathrm{r}}$ and $\mathcal{K}_{n-1}^{\mathrm{u}}$ denote the sets of indices k of reliable and unreliable PTs at time $n-1$, respectively. For PTs $k\in\mathcal{K}_{n-1}^{\mathrm{r}}$, we set the birth and survival probabilities at time n to $p_{n,k}^{\mathrm{b}}=0$ and $p_{n,k}^{\mathrm{s}}=p^{\mathrm{s}}$, respectively. Since $p_{n,k}^{\mathrm{b}}=0$, no birth pdf is needed at time n (cf. (5)).

For PTs $k\in\mathcal{K}_{n-1}^{\mathrm{u}}$, we set $p_{n,k}^{\mathrm{s}}=0$ and $p_{n,k}^{\mathrm{b}}=\mu^{\mathrm{b}}/|\mathcal{K}_{n-1}^{\mathrm{u}}|$ (here, it is assumed that $\mu^{\mathrm{b}}\leq|\mathcal{K}_{n-1}^{\mathrm{u}}|$). Furthermore, we construct the birth pdf $f_{\mathrm{b}}(\mathbf{x}_{n,k})$ as follows.[6] Consider an arbitrary sensor s_0, and let $\mathcal{Z}_{n-1}^{(s_0)} \triangleq \big\{\mathbf{z}_{n-1,m}^{(s_0)}\big\}_{m\in\mathcal{M}_{n-1}^{(s_0)}}$ denote the set of measurements of that sensor at time $n-1$. (Note that $\mathcal{Z}_{n-1}^{(s_0)}$ corresponds to the measurement vector $\mathbf{z}_{n-1}^{(s_0)} = \Big[\mathbf{z}_{n-1,1}^{(s_0)\mathrm{T}}\cdots\mathbf{z}_{n-1,\mathcal{M}_{n-1}^{(s_0)}}^{(s_0)\mathrm{T}}\Big]^{\mathrm{T}}$, with the difference that the elements

[6] We note that for $|\mathcal{K}_{n-1}^{\mathrm{u}}|\to\infty$, this scheme is equivalent to the standard Poisson birth model.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSP.2017.2688966, IEEE Transactions on Signal Processing

$\mathbf{z}_{n-1,m}^{(s_0)}$ are considered unordered in $\mathcal{Z}_{n-1}^{(s_0)}$.) We partition $\mathcal{Z}_{n-1}^{(s_0)}$ into disjoint subsets $\mathcal{Z}_{n-1,k}^{\mathrm{b}}$, $k \in \mathcal{K}_{n-1}^{\mathrm{u}}$ such that the cardinalities of the $\mathcal{Z}_{n-1,k}^{\mathrm{b}}$ differ at most by 1, i.e., $\big||\mathcal{Z}_{n-1,k}^{\mathrm{b}}| - |\mathcal{Z}_{n-1,l}^{\mathrm{b}}|\big| \leq 1$ for any $k, l \in \mathcal{K}_{n-1}^{\mathrm{u}}$. In this way, for $|\mathcal{K}_{n-1}^{\mathrm{u}}| \geq |\mathcal{Z}_{n-1}^{(s_0)}|$, at most a single measurement $\mathbf{z}_{n-1,m}^{(s_0)}$ is assigned to each PT $k \in \mathcal{K}_{n-1}^{\mathrm{u}}$, and for $|\mathcal{K}_{n-1}^{\mathrm{u}}| < |\mathcal{Z}_{n-1}^{(s_0)}|$, the maximum number of measurements assigned to the individual PTs $k \in \mathcal{K}_{n-1}^{\mathrm{u}}$ is as small as possible. Then, based on the kth measurement set $\mathcal{Z}_{n-1,k}^{\mathrm{b}}$, we construct a corresponding "adaptive birth pdf" as

$$f_{\mathrm{b}}(\mathbf{x}_{n,k}) \triangleq \int f(\mathbf{x}_{n,k}|\mathbf{x}_{n-1,k})\, f_{\mathrm{b}}(\mathbf{x}_{n-1,k}; \mathcal{Z}_{n-1,k}^{\mathrm{b}})\, \mathrm{d}\mathbf{x}_{n-1,k}\,,$$

where the pdf $f_{\mathrm{b}}(\mathbf{x}_{n-1,k}; \mathcal{Z}_{n-1,k}^{\mathrm{b}})$ is constructed using $\mathcal{Z}_{n-1,k}^{\mathrm{b}}$ and prior knowledge (e.g., about the target velocity) as discussed in [52]. Particles representing $f_{\mathrm{b}}(\mathbf{x}_{n,k})$ are obtained by drawing particles from $f_{\mathrm{b}}(\mathbf{x}_{n-1,k}; \mathcal{Z}_{n-1,k}^{\mathrm{b}})$ and performing particle-based prediction [51].

VIII. Relation to Existing Methods

Several aspects of the proposed method are related to existing methods, as discussed next.

- The redundant model for data association using both target-oriented and measurement-oriented association variables was previously proposed in [35] and [36]. In [35], BP is used to estimate optical flow parameters. In [36], BP is used for data association (without multitarget tracking).
- Our model for target existence was previously used by the search-initialize-track filter in [10], which, however, is not BP-based, considers only a single sensor, does not employ the redundant model for data association, and uses a different track initialization scheme.
- In the case of a single target and a single sensor, the proposed method reduces to the particle-based implementation of the Bernoulli filter [16] (which is derived using the FISST framework).
- The TOMB/P filter [20], which is effectively a FISST-based variant of the JIPDA filter that uses BP and the redundant data association model, differs from the proposed method in the following respects: it is restricted to a single sensor and to linear-Gaussian state evolution and measurement models—see [53] for an extension to nonlinear, non-Gaussian models—and the number of PTs (tracks) varies over time. We note that a simple multisensor extension of the TOMB/P filter based on the IC principle would lead to a performance that depends on the order in which the measurements from different sensors are processed.
- If the parameters of the proposed method are chosen such that all targets exist at all times (this special case was mentioned in Section II-A, and was considered in our previous work in [38]), then the method becomes similar to the Monte Carlo JPDA filter [7] in that it uses a similar particle-based processing scheme. However, contrary to the Monte Carlo JPDA filter, the proposed method performs data association by means of BP, is based on the redundant model for data association, and can also be used when the number of targets is unknown.

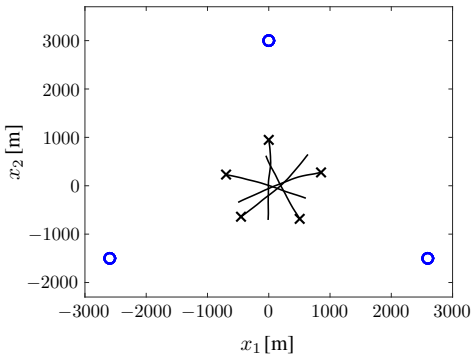

Fig. 4. Sensor positions (marked by circles), initial target positions (marked by crosses), and example target trajectories.

IX. Simulation and Experimental Results

Next, we report simulation and experimental results assessing the performance of our method and comparing it with that of five previously proposed methods for multisensor-multitarget tracking.

A. Simulation Setting

We simulated up to five actual targets whose states consist of two-dimensional (2D) position and velocity, i.e., $\mathbf{x}_{n,k} = [x_{1,n,k} \;\; x_{2,n,k} \;\; \dot{x}_{1,n,k} \;\; \dot{x}_{2,n,k}]^{\mathrm{T}}$. The targets move in a region of interest (ROI) given by $[-3000\,\mathrm{m}, 3000\,\mathrm{m}] \times [-3000\,\mathrm{m}, 3000\,\mathrm{m}]$ according to the near constant-velocity motion model, i.e., $\mathbf{x}_{n,k} = \mathbf{A}\mathbf{x}_{n-1,k} + \mathbf{W}\mathbf{u}_{n,k}$, where $\mathbf{A} \in \mathbb{R}^{4\times 4}$ and $\mathbf{W} \in \mathbb{R}^{4\times 2}$ are chosen as in [54, Sec. 6.3.2] with $T = 1\,\mathrm{s}$, and $\mathbf{u}_{n,k} \sim \mathcal{N}(\mathbf{0}, \sigma_u^2 \mathbf{I}_2)$ with $\sigma_u^2 = 0.025\,\mathrm{m}^2/\mathrm{s}^4$ is an independent and identically distributed (iid) sequence of 2D Gaussian random vectors. The birth distribution $f_{\mathrm{b}}(\mathbf{x}_{n,k})$, birth probabilities $p_{n,k}^{\mathrm{b}}$, and survival probabilities $p_{n,k}^{\mathrm{s}}$ were chosen as described in Section VII, using mean $\mu^{\mathrm{b}} = 0.01$ and "global" survival probability $p^{\mathrm{s}} = 0.999$. The number of PTs was set to $K = 8$ unless noted otherwise. We considered a challenging scenario where the five target trajectories intersect at the ROI center. The target trajectories were generated by first assuming that the five targets start moving toward the ROI center from positions uniformly placed on a circle of radius 1000 m about the ROI center, with an initial speed of 1 m/s, and then letting the targets start to exist at times $n = 5$, 10, 15, 20, and 25. We considered $S = 3$ sensors unless noted otherwise. The sensors are located uniformly on a circle of radius 3000 m about the ROI center. Fig. 4 shows the sensor positions, the initial target positions, and example target trajectories.

The sensors perform range and bearing measurements within a measurement range of 6000 m. More specifically, within the measurement range, the target-generated measurements are given by

$$\mathbf{z}_{n,m}^{(s)} = \big[\, \big\|\tilde{\mathbf{x}}_{n,k} - \mathbf{p}^{(s)}\big\| \;\; \varphi\big(\tilde{\mathbf{x}}_{n,k}, \mathbf{p}^{(s)}\big) \,\big]^{\mathrm{T}} + \mathbf{v}_{n,m}^{(s)}\,,$$

where $\tilde{\mathbf{x}}_{n,k} \triangleq [x_{1,n,k} \;\; x_{2,n,k}]^{\mathrm{T}}$, $\mathbf{p}^{(s)}$ is the position of sensor s, $\varphi\big(\tilde{\mathbf{x}}_{n,k}, \mathbf{p}^{(s)}\big)$ is the angle (in degrees) of the vector $\tilde{\mathbf{x}}_{n,k}$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSP.2017.2688966, IEEE Transactions on Signal Processing

relative to the vector $\mathbf{p}^{(s)}$, and $\mathbf{v}_{n,m}^{(s)} \sim \mathcal{N}(\mathbf{0}, \mathbf{C}_v)$ with $\mathbf{C}_v = \mathrm{diag}\{(10\mathrm{m})^2, (0.5^\circ)^2\}$ is an iid sequence of 2D Gaussian random vectors. The false alarm pdf $f_{\mathrm{FA}}(\mathbf{z}_{n,m}^{(s)})$ is linearly increasing on $[0\mathrm{m}, 6000\mathrm{m}]$ and zero outside that interval with respect to the range component, and uniform on $[0^\circ, 360^\circ)$ with respect to the angle component. (In Cartesian coordinates, this corresponds to a uniform distribution on the sensor's measurement area.) The mean number of false alarm measurements is $\mu^{(s)}=2$ if not noted otherwise.

Our implementation of the proposed method used $J = 3000$ "legacy" particles and $I = 3000$ birth particles for each PT. We performed $P = 20$ BP iterations. The threshold for target detection was $P_{\mathrm{th}} = 0.5$, and the reliability threshold was $R_{\mathrm{th}} = 10^{-3}$. We simulated 150 time steps n.

B. Performance Comparison Using Simulated Data

We compare the proposed BP method with particle implementations of the IC-PHD filter [3], [12], [26], the IC-CPHD filter [3], [15], [26], the IC-MB filter [3], [17], and the partition-based MS-PHD and MS-CPHD filters [23], [24]. The "IC-" filters are straightforward multisensor extensions performing a single-sensor update step sequentially for each sensor [3], [5], [26]. In addition, we consider a (yet unpublished) multisensor extension of the TOMB/P filter [20] based on the IC principle, using the particle implementation presented in [53], which will be referred to as IC-TOMB/P filter. We note that for the special case of a known number of targets, a comparison of our method with multisensor JPDA methods has been presented in [38].

The partition-based MS-(C)PHD filters approximate the exact multisensor (C)PHD filters; they can outperform the IC-(C)PHD filters but have a higher computational complexity. Since the trellis algorithm used for partition extraction in the original formulation of the MS-(C)PHD filter [23], [24] is only suitable for a Gaussian mixture implementation, it was adapted to a particle-based implementation. We note that the exact multisensor (C)PHD filters are not computationally feasible for the simulated scenario since their complexity scales exponentially in the number of sensors and in the number of measurements per sensor [22], [25], [55]. The performance of the various methods is measured by the Euclidean distance based OSPA metric with a cutoff parameter of 200 [56].

The (C)PHD-type filters use 24000 particles to represent the PHD of the target states. As in [12], [15], they perform kmeans++ clustering [57] for state estimation. The IC-MB filter uses 3000 particles to represent each Bernoulli component. The maximum numbers of subsets and partitions used by the MS-(C)PHD filter are 120 and 720, respectively, similarly to [23], [24]. The IC-TOMB/P filter uses 3000 particles for each Bernoulli component. The mean of the intensity function for undetected targets at time $n = 0$ is zero. After each single-sensor update step, Bernoulli components with existence probability smaller then 10^{-5} are removed.

With the above-mentioned parameters (in particular, $K = 8$ PTs and $S = 3$ sensors) and a detection probability of $P_{\mathrm{d}}^{(s)}(\mathbf{x}_{n,k}) = P_{\mathrm{d}} = 0.8$, the runtime per time step for a MATLAB implementation without gating on a single core of an Intel Xeon E5-2640 v3 CPU was measured as 0.07 s for the proposed algorithm, 0.11 s for the IC-PHD filter, 0.14 s for the IC-CPHD filter, 0.27 s for the IC-MB filter, 13.21 s for the MS-PHD filter, 13.82 s for MS-CPHD filter, and 0.08 s for the IC-TOMB/P filter. All runtimes are based on plain MATLAB code without any precompiled C/C++ parts. The high runtimes of the MS-(C)PHD filters are due to the fact that the trellis algorithm used for partition extraction is tailored to a Gaussian mixture implementation and becomes computationally intensive in a particle-based implementation. We note that the efficient extraction of high-quality partitions in a particle-based implementation of the MS-(C)PHD filter is still an open problem.

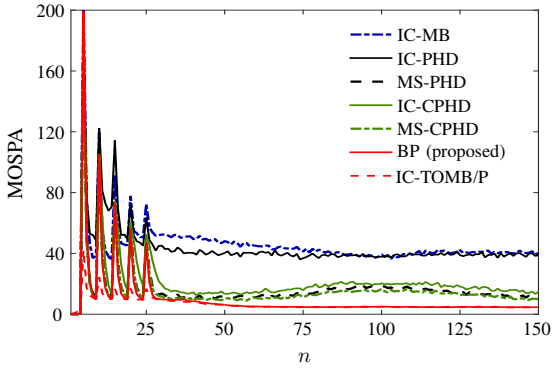

Fig. 5. MOSPA error versus time n for $S = 3$ sensors, $P_{\mathrm{d}} = 0.8$, and $\mu^{(s)} = 2$.

Fig. 5 shows the mean OSPA (MOSPA) error—averaged over 400 simulation runs—of all methods versus time n, assuming $S = 3$ sensors with a detection probability of $P_{\mathrm{d}} = 0.8$. The error exhibits peaks at times $n = 5$, 10, 15, 20, and 25 because of target births. However, very soon after a target birth, the proposed method as well as the IC-CPHD, MS-PHD, MS-CPHD, and IC-TOMB/P filters are able to reliably estimate the number of targets. The proposed method is seen to outperform all the other methods except the IC-TOMB/P filter. In particular, it outperforms the IC-CPHD, MS-PHD, and MS-CPHD filters mainly because particle implementations of (C)PHD filters involve a potentially unreliable clustering step. This clustering step is especially unreliable for targets that are close to each other. This fact explains the higher MOSPA error of the IC-CPHD, MS-PHD, and MS-CPHD filters around $n = 100$, i.e., around the time when the target trajectories intersect in the ROI center. Furthermore, the IC-CPHD filter suffers from the "spooky effect" [58], i.e., when a missed detection occurs, PHD weight is shifted from the undetected part of the PHD to the detected part, no matter how distant and unrelated these parts may be. The MOSPA error of the IC-PHD and IC-MB filters is seen to be significantly larger than that of the other methods; this is caused by the inability of these filters to reliably estimate the number of targets. We note that for sensors with different probabilities of detection, the performance loss of IC-(C)PHD filters relative to MS-(C)PHD filters tends to be larger than in Fig. 5 [23], [24]. Finally, the proposed method performs similarly to the IC-TOMB/P filter, at a computational complexity that is similar for the considered scenario. (However, we will show in Section IX-C that the computational complexities of the two methods

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSP.2017.2688966, IEEE Transactions on Signal Processing

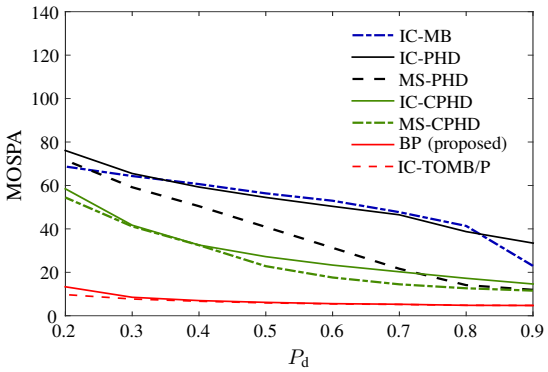

Fig. 6. Time-averaged MOSPA error versus P_d for $S=3$ sensors and $\mu^{(s)}=2$.

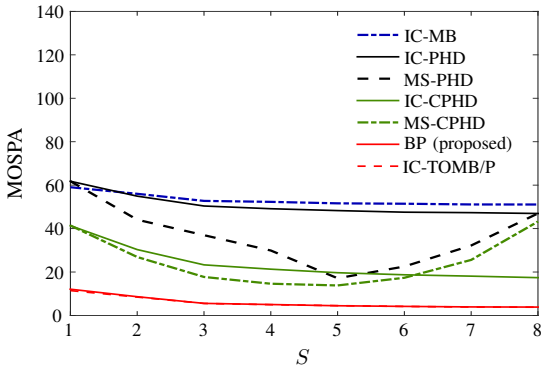

Fig. 7. Time-averaged MOSPA error versus number of sensors S for $P_\mathrm{d}=0.6$ and $\mu^{(s)}=2$.

scale very differently.)

Fig. 6 shows the time-averaged MOSPA error—averaged over time steps $n \in \{50,\dots,150\}$—versus the detection probability P_d for $S=3$ sensors. For all methods, as expected, the MOSPA error decreases with decreasing P_d. Fig. 7 shows the time-averaged MOSPA error versus the number of sensors S for $P_\mathrm{d}=0.6$. It can be seen that the MOSPA error of the MS-PHD and MS-CPHD filters increases for S larger than 5; this is because the chosen maximum numbers of subsets (120) and partitions (720) are too small for that case. (We note that choosing larger maximum numbers of subsets and partitions leads to excessive simulation times.) Finally, Fig. 8 shows the time-averaged MOSPA error versus the mean number of false alarms $\mu^{(s)}$ for $S=3$ and $P_\mathrm{d}=0.6$. As expected, the MOSPA error of all methods increases with growing $\mu^{(s)}$. In addition, Figs. 6–8 again show that the proposed method outperforms the other methods except the IC-TOMB/P filter. We note that the poor performance of the IC-MB filter is due to the approximation used by that filter, which is accurate only for a high P_d and a very low $\mu^{(s)}$ [17].

C. Scalability

Finally, we investigate how the runtime of our method scales in the number of sensors S and in the number of actual targets, and how it depends on the mean number of false alarms $\mu^{(s)}$. Fig. 9 shows the average runtime per time step n versus S, $\mu^{(s)}$, and the number of actual targets for a MATLAB implementation on a single core of an Intel Xeon E5-2640 v3 CPU. The runtime was averaged over 150 time

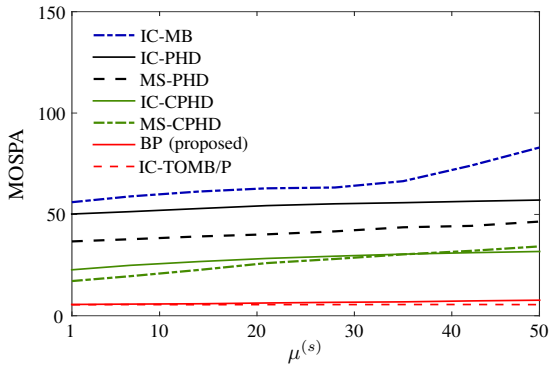

Fig. 8. Time-averaged MOSPA error versus mean number of false alarms $\mu^{(s)}$ for $S=3$ and $P_\mathrm{d}=0.6$.

steps and 400 simulation runs. The probability of detection was set to $P_\mathrm{d}=0.6$. These results confirm the linear scaling of the runtime in S and in the mean number of measurements per sensor (which grows linearly with $\mu^{(s)}$). The scaling in the number of actual targets is seen to be roughly quadratic. Further investigation showed that the scaling in the number of actual targets is linear if the number of PTs is held fixed, and similarly, the scaling in the number of PTs is linear if the number of actual targets is held fixed. The low absolute complexity of the proposed method is evidenced by the fact that for 20 actual targets, $S=3$ sensors, and $\mu^{(s)}=10$, the computations per time step n require less than 0.5 s.

While the proposed method was observed in Section IX-B to perform similarly to the IC-TOMB/P filter, its advantage over the IC-TOMB/P filter is its drastically improved scalability. In order to demonstrate this advantage, we depict in Fig. 10 the runtime of the IC-TOMB/P filter, again versus S, $\mu^{(s)}$, and the number of actual targets to permit an easy comparison with Fig. 9. It is seen that for scenarios with a high $\mu^{(s)}$ and for scenarios with a large number of actual targets, the runtime of the IC-TOMB/P filter is significantly higher than that of the proposed method. We note that we had to restrict the number of actual targets to maximally 10, rather than 20 in Fig. 9, to avoid excessive simulation times.

D. Results for a Multistatic Sonar Tracking Experiment

To validate our method, we present results obtained for real measurements that were acquired by a multistatic sonar network during the Littoral Continuous Active Sonar 2015 (LCAS15) sea trial [59]. As shown in Fig. 11, the network consisted of an acoustic source and two receivers (sensors). In our experiment, an echo repeater towed by a research vessel (not shown in Fig. 11) served as a nominal target (replacing the target in Fig. 11). The source operated in continuous active sonar mode [60] and transmitted linearly frequency-modulated sweeps with a repetition period of 20 s and a duty cycle of almost 100 %. The receivers were hydrophone arrays towed by two autonomous underwater vehicles (AUVs); they computed measurements $\mathbf{z}_{n,m}^{(s)}$ from the output of the respective hydrophone array as described in [61]. The measurements follow a bistatic range-bearing measurement model with port-starboard ambiguity [62, Sec. II-B]. At each sensor $s \in \{1,2\}$, measurements $\mathbf{z}_{n}^{(s)}$ were available every $T=2$ s.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSP.2017.2688966, IEEE Transactions on Signal Processing

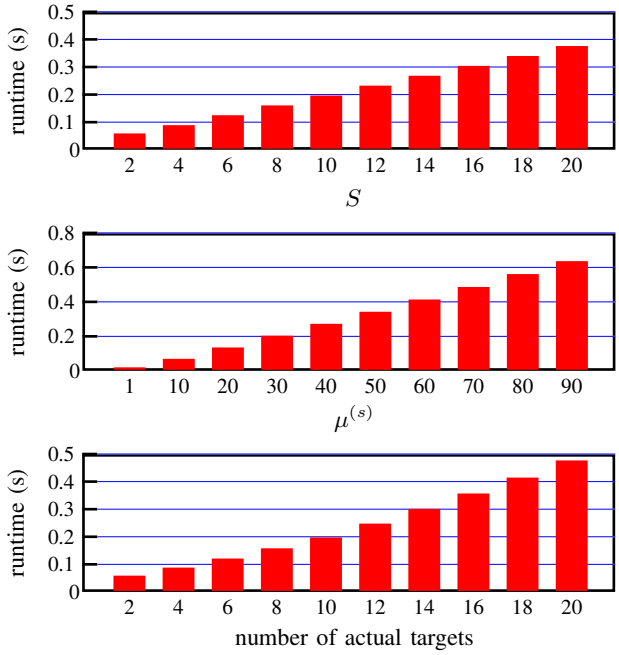

Fig. 9. Average runtime per time step of the proposed method. Top: versus S for five actual targets and $\mu^{(s)} = 10$; center: versus $\mu^{(s)}$ for $S = 3$ and five actual targets; bottom: versus the number of actual targets for $S = 3$, $\mu^{(s)} = 10$, and the number of PTs set to $K = 5, 7, \ldots, 23$.

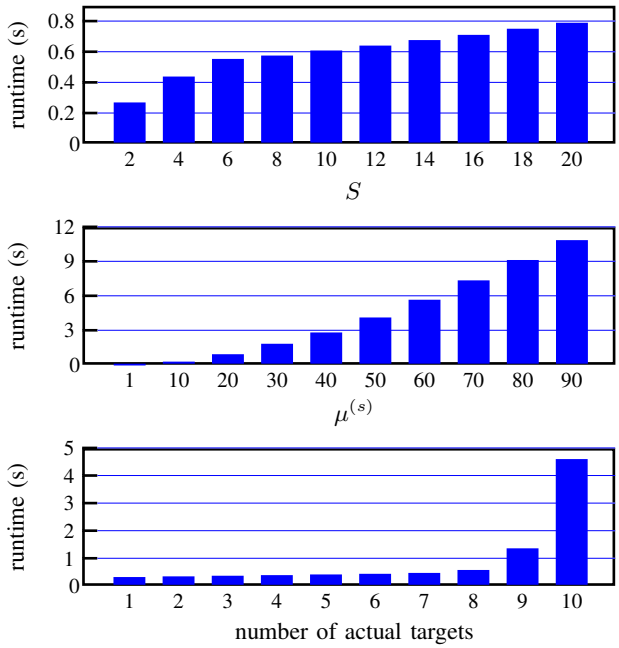

Fig. 10. Average runtime per time step of the IC-TOMB/P filter. Top: versus S for five actual targets and $\mu^{(s)} = 10$; center: versus $\mu^{(s)}$ for $S = 3$ and five actual targets; bottom: versus the number of actual targets for $S = 3$ and $\mu^{(s)} = 10$.

Our numerical study was based on the following assumptions and choices. The positions of the source and receivers are assumed known (since GPS position information was available). Although only one actual target is present, the number of PTs is set to $K = 8$ to accommodate additional targets simulated by features of the seafloor. The target states consist of 2D position and velocity. Target motion is modeled

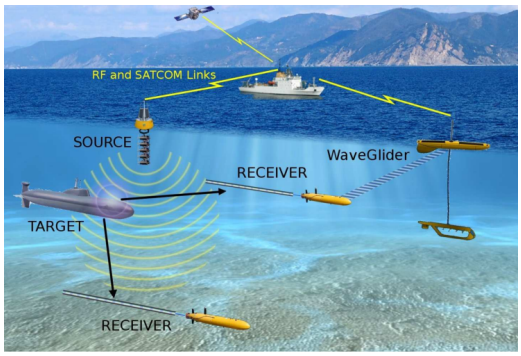

Fig. 11. Multistatic sonar network consisting of an acoustic source and two receivers (hydrophone arrays) towed by AUVs. A research vessel and a wave glider act as a communication relay.

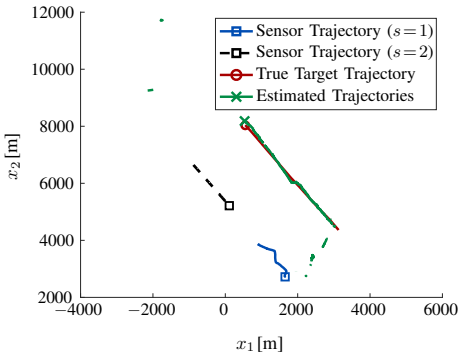

Fig. 12. Sensor trajectories, true target trajectory, and estimated trajectories. The squares, circle, and cross indicate the end point of the respective trajectory.

by the near constant velocity model (see Section IX-A) with driving noise variance $\sigma_u^2 = 0.001\,\mathrm{m}^2/\mathrm{s}^4$. The birth and survival probabilities are chosen as $p_{n,k}^{\mathrm{b}} = 0.8 \cdot 10^{-5}$ and $p_{n,k}^{\mathrm{s}} = 0.9995$, respectively. The covariance matrix of the measurement noise vectors used in the measurement model [62, Sec. II-B] is $\mathrm{diag}\{(20\mathrm{m})^2, (1^\circ)^2\}$. The false alarm pdf $f_{\mathrm{FA}}(\mathbf{z}_{n,m}^{(s)})$ is linearly increasing on $[0\mathrm{m}, 15000\mathrm{m}]$ and zero outside that interval with respect to the range component, and uniform on $[0^\circ, 360^\circ)$ with respect to the angle component. (In Cartesian coordinates, this is a uniform distribution on a disc of radius 15000m.) The mean number of false alarm measurements is $\mu^{(s)} = 18$. The detection threshold is $P_{\mathrm{th}} = 0.75$, and the reliability threshold is $R_{\mathrm{th}} = 10^{-3}$. We used $P = 20$ BP iterations for iterative data association, $J = 10000$ particles, and $I = 10000$ birth particles.

For an assumed detection probability of $P_{\mathrm{d}} = 0.5$, Fig. 12 shows the sensor trajectories, the true target trajectory (which was provided by GPS measurements), and the trajectories estimated by the proposed method from 35 min of measurement data. One can see that after a few seconds, the target is detected and reliably tracked with high accuracy. There are several false trajectories, which are probably related to features of the seafloor. The presence of such features motivates the use of a multitarget tracking method, because a single-target tracking method may track a feature rather than the target.

We also compared the proposed algorithm with the distributed MHT proposed in [63], [64], which is a track-oriented MHT that was developed for the considered multistatic sonar

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSP.2017.2688966, IEEE Transactions on Signal Processing

	TOT [%]	FAR [$\mathrm{s}^{-1}\mathrm{km}^{-2}$]
BP ($P_\mathrm{d}=0.2$)	83	$5.1\cdot10^{-4}$
BP ($P_\mathrm{d}=0.3$)	94	$4.0\cdot10^{-4}$
BP ($P_\mathrm{d}=0.4$)	98	$3.3\cdot10^{-4}$
BP ($P_\mathrm{d}=0.5$)	97	$2.5\cdot10^{-4}$
BP ($P_\mathrm{d}=0.6$)	87	$1.9\cdot10^{-4}$
BP ($P_\mathrm{d}=0.7$)	47	$1.2\cdot10^{-4}$
MHT ($N_\mathrm{m}=8$)	84	$12.4\cdot10^{-4}$
MHT ($N_\mathrm{m}=7$)	81	$10.0\cdot10^{-4}$
MHT ($N_\mathrm{m}=6$)	74	$9.3\cdot10^{-4}$
MHT ($N_\mathrm{m}=5$)	67	$7.0\cdot10^{-4}$
MHT ($N_\mathrm{m}=4$)	63	$5.1\cdot10^{-4}$
MHT ($N_\mathrm{m}=3$)	52	$4.2\cdot10^{-4}$

TABLE I
TOT AND FAR FOR THE PROPOSED METHOD AND THE MHT.

tracking application. The MHT is based on the extended Kalman filter and thus uses a linearization of the measurement model. Therefore, it cannot explicitly model the port-starboard ambiguity of the measurement model. This leads to the formation of one "ghost" trajectory for each target and each sensor (see [62] for details). In the MHT, the validation threshold was set to 25, the depth of the filter to three scans, and the track confirmation logic of the filter to 5/3. Furthermore, we considered six different values of the maximum number of missed detections, N_m, ranging from 3 to 8 in steps of 1. All other parameters had the same values as in the proposed method. In the proposed method, we assumed six different values of P_d ranging from 0.2 to 0.7 in steps of 0.1. We note that the chosen values of N_m in the MHT and of P_d in the proposed method define comparable ranges of operating characteristics for the two methods. More specifically, decreasing P_d in the proposed method generally has the effect that more targets are detected and longer trajectories are estimated. A similar effect is obtained in the MHT by increasing N_m.

The performance metrics for this comparison were time-on-target (TOT) and false alarm rate (FAR) [2]. TOT is the fraction of time that the target is successfully detected, which was considered to be true if the target position estimate was within 300 m of the true target position. FAR is the normalized number of false trajectories (or false detections) generated in the surveillance region per unit of 2D space and time. In order to achieve a fair comparison and to minimize the impact of ghost trajectories on MHT performance, in the FAR calculation for the MHT, we removed $2\cdot\mathrm{TOT}$ false alarms at each time step. The TOT and FAR obtained with both methods are shown in Table I. It can be seen that the proposed method achieves a more attractive TOT/FAR tradeoff than the MHT.

X. Conclusion

We developed and demonstrated the application of the belief propagation (BP) scheme to the problem of tracking an unknown number of targets using multiple sensors. The proposed BP-based multitarget tracking method exhibits low complexity and excellent scaling properties with respect to all relevant system parameters. This is achieved through the use of "augmented target states" including binary target indicators and the establishment of an appropriate statistical model involving a redundant formulation of data association uncertainty [36]. The complexity of our method scales only quadratically in the number of targets, linearly in the number of sensors, and linearly in the number of measurements per sensors. Simulation results in a challenging scenario with intersecting target trajectories showed that the proposed method can outperform previously proposed methods, including methods with a less favorable scaling behavior. In particular, we observed significant improvements in OSPA performance relative to various multisensor extensions of the PHD, CPHD, and multi-Bernoulli filters. The application of the proposed method to real measurement data from a multistatic sonar tracking experiment demonstrated an improved tradeoff between time-on-target and false alarm rate compared to a track-oriented multi-hypothesis tracker. We note that an extension of our method to an unknown time-varying probability of detection is presented in [65].

Promising directions for future research include distributed variants of our method for use in decentralized wireless sensor networks with communication constraints [66] and FISST-based multisensor-multitarget tracking methods using BP for data association.

Acknowledgment

The real measurements used in Section IX-D were provided by the LCAS Multi-National Joint Research Project (MN-JRP). The authors would like to thank the anonymous reviewers, whose comments led to an improvement of this paper.

References

[1] W. Koch, *Tracking and Sensor Data Fusion: Methodological Framework and Selected Applications*. Berlin, Germany: Springer, 2014.
[2] Y. Bar-Shalom, P. K. Willett, and X. Tian, *Tracking and Data Fusion: A Handbook of Algorithms*. Storrs, CT, USA: Yaakov Bar-Shalom, 2011.
[3] R. Mahler, *Statistical Multisource-Multitarget Information Fusion*. Norwood, MA, USA: Artech House, 2007.
[4] D. B. Reid, "An algorithm for tracking multiple targets," *IEEE Trans. Autom. Control*, vol. 24, no. 6, pp. 843–854, Dec. 1979.
[5] L. Y. Pao and C. W. Frei, "A comparison of parallel and sequential implementations of a multisensor multitarget tracking algorithm," in *Proc. ACC-95*, vol. 3, Seattle, WA, USA, Jun. 1995, pp. 1683–1687.
[6] S. Deb, M. Yeddanapudi, K. Pattipati, and Y. Bar-Shalom, "A generalized S-D assignment algorithm for multisensor-multitarget state estimation," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 33, no. 2, pp. 523–538, Apr. 1997.
[7] J. Vermaak, S. J. Godsill, and P. Perez, "Monte Carlo filtering for multi target tracking and data association," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 41, no. 1, pp. 309–332, Jan. 2005.
[8] 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.
[9] 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.
[10] P. Horridge and S. Maskell, "Searching for, initiating and tracking multiple targets using existence probabilities," in *Proc. FUSION-09*, Seattle, WA, USA, Jul. 2009, pp. 611–617.
[11] R. Mahler, "Multitarget Bayes filtering via first-order multitarget moments," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 39, no. 4, pp. 1152–1178, Oct. 2003.
[12] B.-N. Vo, S. Singh, and A. Doucet, "Sequential Monte Carlo methods for multitarget filtering with random finite sets," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 41, no. 4, pp. 1224–1245, Oct. 2005.
[13] D. E. Clark and J. Bell, "Multi-target state estimation and track continuity for the particle PHD filter," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 43, no. 4, pp. 1441–1453, Oct. 2007.
[14] R. Mahler, "PHD filters of higher order in target number," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 43, no. 4, pp. 1523–1543, Oct. 2007.

[15] B.-T. Vo, B.-N. Vo, and A. Cantoni, "Analytic implementations of the cardinalized probability hypothesis density filter," *IEEE Trans. Signal Process.*, vol. 55, no. 7, pp. 3553–3567, Jul. 2007.

[16] 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.

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

[18] S. Reuter, B.-T. Vo, B.-N. Vo, and K. Dietmayer, "The labeled multi-Bernoulli filter," *IEEE Trans. Signal Process.*, vol. 62, no. 12, pp. 3246–3260, Jun. 2014.

[19] B.-N. Vo, B.-T. Vo, and D. Phung, "Labeled random finite sets and the Bayes multi-target tracking filter," *IEEE Trans. Signal Process.*, vol. 62, no. 24, pp. 6554–6567, Dec. 2014.

[20] 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.

[21] P. Braca, S. Marano, V. Matta, and P. Willett, "Asymptotic efficiency of the PHD in multitarget/multisensor estimation," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 3, pp. 553–564, Jun. 2013.

[22] R. Mahler, "The multisensor PHD filter: I. General solution via multi-target calculus," in *Proc. SPIE-09*, Orlando, FL, USA, Apr. 2009.

[23] S. Nannuru, M. Coates, M. Rabbat, and S. Blouin, "General solution and approximate implementation of the multisensor multitarget CPHD filter," in *Proc. IEEE ICASSP-15*, Brisbane, Australia, Apr. 2015, pp. 4055–4059.

[24] S. Nannuru, S. Blouin, M. Coates, and M. Rabbat, "Multisensor CPHD filter," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 52, no. 4, pp. 1834–1854, Aug. 2016.

[25] E. Delande, E. Duflos, P. Vanheeghe, and D. Heurguier, "Multi-sensor PHD: Construction and implementation by space partitioning," in *Proc. IEEE ICASSP-11*, Prague, Czech Republic, May 2011, pp. 3632–3635.

[26] S. Nagappa and D. Clark, "On the ordering of the sensors in the iterated-corrector probability hypothesis density (PHD) filter," in *Proc. SPIE-11*, Orlando, FL, USA, Apr. 2011, pp. 26–28.

[27] R. Mahler, "Approximate multisensor CPHD and PHD filters," in *Proc. FUSION-10*, Edinburgh, UK, Jul. 2010, pp. 26–29.

[28] M. Tobias and A. D. Lanterman, "Multitarget tracking using multiple bistatic range measurements with probability hypothesis densities," in *Proc. SPIE-04*, Orlando, FL, USA, Apr. 2004, pp. 296–305.

[29] G. Battistelli, L. Chisci, S. Morrocchi, F. Papi, A. Farina, and A. Graziano, "Robust multisensor multitarget tracker with application to passive multistatic radar tracking," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 48, no. 4, pp. 3450–3472, Oct. 2012.

[30] C. Fantacci and F. Papi, "Scalable multisensor multitarget tracking using the marginalized δ-GLMB density," *IEEE Signal Process. Lett.*, vol. 23, no. 6, pp. 863–867, Jun. 2016.

[31] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.

[32] H. Wymeersch, *Iterative Receiver Design.* New York, NY, USA: Cambridge University Press, 2007.

[33] M. J. Wainwright and M. I. Jordan, "Graphical models, exponential families, and variational inference," *Found. Trends Mach. Learn.*, vol. 1, Jan. 2008.

[34] H.-A. Loeliger, "An introduction to factor graphs," *IEEE Signal Process. Mag.*, vol. 21, no. 1, pp. 28–41, Jan. 2004.

[35] M. Chertkov, L. Kroc, F. Krzakala, M. Vergassola, and L. Zdeborová, "Inference in particle tracking experiments by passing messages between images," *PNAS*, vol. 107, no. 17, pp. 7663–7668, Apr. 2010.

[36] J. L. Williams and R. Lau, "Approximate evaluation of marginal association probabilities with belief propagation," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 50, no. 4, pp. 2942–2959, Oct. 2014.

[37] Z. Chen, L. Chen, M. Cetin, and A. S. Willsky, "An efficient message passing algorithm for multi-target tracking." in *Proc. FUSION-09*, Seattle, WA, USA, Jul. 2009, pp. 826–833.

[38] F. Meyer, P. Braca, P. Willett, and F. Hlawatsch, "Scalable multitarget tracking using multiple sensors: A belief propagation approach," in *Proc. FUSION-15*, Washington D.C., USA, Jul. 2015, pp. 1778–1785.

[39] P. Horridge and S. Maskell, "Real-time tracking of hundreds of targets with efficient exact JPDAF implementation," in *Proc. FUSION-06*, Florence, Italy, Jul. 2006.

[40] L. Chen, M. J. Wainwright, M. Cetin, and A. S. Willsky, "Data association based on optimization in graphical models with application to sensor networks," *Math. Comp. Model.*, vol. 43, no. 9–10, pp. 1114–1135, 2006.

[41] F. Meyer, P. Braca, P. Willett, and F. Hlawatsch, "Tracking an unknown number of targets using multiple sensors: A belief propagation method," in *Proc. FUSION-16*, Heidelberg, Germany, Jul. 2016, pp. 719–726.

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

[43] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Upper Saddle River, NJ, USA: Prentice-Hall, 1993.

[44] R. Kindermann and J. L. Snell, *Markov Random Fields and Their Applications*. Providence, RI, USA: AMS, 1980.

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

[46] T. Sathyan and M. Hedley, "Fast and accurate cooperative tracking in wireless networks," *IEEE Trans. Mobile Comput.*, vol. 12, no. 9, pp. 1801–1813, Sep. 2013.

[47] 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.

[48] P. O. Vontobel, "The Bethe permanent of a nonnegative matrix," *IEEE Trans. Inf. Theory*, vol. 59, no. 3, pp. 1866–1901, Mar. 2013.

[49] J. L. Williams and R. A. Lau, "Convergence of loopy belief propagation for data association," in *Proc. ISSNIP-10*, Brisbane, Australia, Dec. 2010, pp. 175–180.

[50] 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.

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

[52] B. Ristic and S. Arulampalam, "Bernoulli particle filter with observer control for bearings-only tracking in clutter," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 48, no. 3, pp. 2405–2415, Jul. 2012.

[53] T. Kropfreiter, F. Meyer, and F. Hlawatsch, "Sequential Monte Carlo implementation of the track-oriented marginal multi-Bernoulli/Poisson filter," in *Proc. FUSION-16*, Heidelberg, Germany, Jul. 2016, pp. 972–979.

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

[55] R. Mahler, "The multisensor PHD filter: II. Erroneous solution via Poisson magic," in *Proc. SPIE-09*, Orlando, FL, USA, Apr. 2009.

[56] D. Schuhmacher, B.-T. Vo, and B.-N. Vo, "A consistent metric for performance evaluation of multi-object filters," *IEEE Trans. Signal Process.*, vol. 56, no. 8, pp. 3447–3457, Aug. 2008.

[57] G. Gan, C. Ma, and J. Wu, *Data Clustering: Theory, Algorithms, and Applications*. Philadelphia, PA, USA: SIAM, 2007.

[58] D. Franken, M. Schmidt, and M. Ulmke, ""Spooky action at a distance" in the cardinalized probability hypothesis density filter," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 45, no. 4, pp. 1657–1664, Oct. 2009.

[59] A. Munafò, J. R. Bates, F. Meyer, and G. Canepa, "CAS vs. PAS performance assessment from LCAS15," Centre for Maritime Research and Experimentation (CMRE), Tech. Rep. CMRE-FR-2016-003, Jul. 2016.

[60] D. Grimmett and C. Wakayama, "Multistatic tracking for continuous active sonar using Doppler-bearing measurements," in *Proc. FUSION-13*, Istanbul, Turkey, Jul. 2013, pp. 258–265.

[61] A. Munafò, G. Canepa, and J. R. Bates, "CAS processing results from LCAS15," Centre for Maritime Research and Experimentation (CMRE), Tech. Rep. CMRE-TPR-NU-601-03-0301-Q1, Apr. 2016.

[62] P. Braca, P. Willett, K. LePage, S. Marano, and V. Matta, "Bayesian tracking in underwater wireless sensor networks with port-starboard ambiguity," *IEEE Trans. Signal Process.*, vol. 62, no. 7, pp. 1864–1878, Apr. 2014.

[63] S. Coraluppi and D. Grimmett, "Multistatic sonar tracking," in *Proc. SPIE-03*, vol. 5096, Orlando, FL, USA, Oct. 2003, pp. 399–410.

[64] S. Coraluppi and C. Carthel, "Distributed tracking in multistatic sonar," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 41, no. 3, pp. 1138–1147, Jul. 2005.

[65] F. Meyer, P. Braca, F. Hlawatsch, M. Micheli, and K. LePage, "Scalable adaptive multitarget tracking using multiple sensors," in *Proc. IEEE GLOBECOM-16*, Washington D.C., USA, Dec. 2016.

[66] K. Sohraby, D. Minoli, and T. Znati, *Wireless Sensor Networks: Technology, Protocols, and Applications*. Hoboken, NJ, USA: Wiley, 2007.