Scalable Multitarget Tracking Using Multiple Sensors: A Belief Propagation Approach

Florian Meyer*, Paolo Braca†, Peter Willett‡, and Franz Hlawatsch*

*Institute of Telecommunications, Vienna University of Technology, 1040 Vienna, Austria ({fmeyer,fhlawats}@nt.tuwien.ac.at)
†NATO STO Centre for Maritime Research and Experimentation, La Spezia 19126, Italy (paolo.braca@cmre.nato.int)
‡Department of ECE, University of Connecticut, Storrs, CT 06269, USA (willett@engr.uconn.edu)

***Abstract*—We propose a method for multisensor-multitarget tracking with excellent scalability in the number of targets (which is assumed known), the number of sensors, and the number of measurements per sensor. Our method employs belief propagation based on a "detailed" factor graph that involves both target-related and measurement-related association variables. Using this approach, 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. We observed very low runtimes of the proposed method; e.g., our MATLAB simulation of a scenario of 30 targets and 10 sensors without gating required less than one second per time step. The performance of the proposed method in terms of mean optimal subpattern assignment (OSPA) error compares well with that of state-of-the-art methods whose complexity scales exponentially with the number of targets. In particular, we observed that our method can outperform the sequential multisensor joint probabilistic data association filter (JPDAF) and performs similar to the Monte Carlo JPDAF.**

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

I. Introduction

Multitarget tracking aims to estimate the states (positions, velocities, and possibly further parameters) of moving objects (targets) over time, based on measurements provided by remote sensing devices such as radar, sonar, or cameras [1]. This is complicated by a data association uncertainty (i.e., unknown association between measurements and targets) [1] and the fact that the number of targets is usually unknown [2]. "Classic" methods for multitarget tracking assume that the number of targets is known and estimate the target states jointly with the association variables. Two examples are the joint probabilistic data association filter (JPDAF) [1] and the multiple hypothesis density tracker (MHT) [3]. Extensions of these methods to multiple sensors include [4]–[7].

More recent methods are based on finite set statistics (FISST) [2], [8] and jointly estimate the number of targets and the target states without explicitly estimating the association variables. In particular, the probability hypothesis density (PHD) filter [2] is asymptotically efficient in the number of sensors [9], but the complexity of calculating the multisensor PHD is exponential in the number of sensors [8]. Approximations of the multisensor PHD are used in [10], [11]. Several multisensor-multitarget tracking methods, both classic and FISST-based, have been successfully applied to real-world data (e.g., [12], [13]). Unfortunately, most multisensor-multitarget tracking methods suffer from a poor scalability in the number of targets and number of sensors.

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. This advantage is achieved by combining the classic approach —under the assumption of a known number of targets— with belief propagation (BP) message passing [14]–[16]. We formulate a joint estimation problem involving the target states and association variables for all times, targets, and sensors. Following [17], [18], our formulation of the data association uncertainty involves both target-oriented and measurement-oriented association vectors. The problem is then described by a factor graph and solved using loopy BP, thereby exploiting independencies of variables for a drastic reduction of computational complexity. The "beliefs" obtained with loopy BP are only approximations of the true marginal posterior probability density functions (pdfs). However, these approximations are very accurate in many applications [14]–[16].

Message passing algorithms have been previously employed for data association (without multitarget tracking or with multitarget tracking outside the message passing scheme). In [19], the Viterbi algorithm is used to estimate the association variables. In [17], [18], loopy BP is used to obtain approximations of the marginal association probabilities for a single time step and a single sensor. In [20], again for a single time step and a single sensor, exact marginal association probabilities are calculated based on a tree representation of the possible associations variables; this exploits the conditional independence of certain target subsets after measurement gating. In [21], BP is used to obtain approximate association probabilities at a single time step for multiple sensors with overlapping regions of interest. This method scales poorly in the number of targets because at any given sensor, all target states are modeled as a joint state, and similarly for all association variables.

In [22], a BP algorithm has been previously proposed for multisensor-multitarget tracking (including data association). All target states and association variables at one time step are modeled as a joint state. Hence, the factor graph is tree-structured and BP message passing is exact but, again, this comes at the cost of poor scalability in the number of targets.

The BP-based multisensor-multitarget tracking method presented in this paper is different from [22] in that it is based on the BP formulation of data association introduced in [17] and a "detailed" factor graph in which every target state and association variable is modeled as an individual node. Because the factor graph involves only low-dimensional variables, the resulting BP algorithm does not perform any high-dimensional operations. As a consequence, the complexity of our method scales only quadratically in the number of targets and linearly in the number of sensors (assuming a fixed number of message passing iterations). Simulation results in a scenario with intersecting targets confirm that our method exhibits excellent scalability while, at the same time, its performance in terms of mean optimal subpattern assignment (OSPA) error compares well with that of state-of-the-art methods whose complexity is exponential in the number of targets. In particular, it can outperform the sequential multisensor JPDAF [6], [7] and performs similar to the Monte Carlo JPDAF (MC-JPDAF).

This paper is organized as follows. In Section II, we describe the system model and state the multisensor-multitarget tracking problem. To prepare the ground for our derivation of the proposed algorithm, a new BP interpretation of the JPDAF is given in Section III, and an efficient BP-based data-association scheme [17], [18], [23] is reviewed in Section IV. Section V presents the proposed BP algorithm for multisensor-multitarget tracking. Finally, simulation results in a scenario with intersecting targets are reported in Section VI.

II. System Model and Problem Formulation

We first describe the system model and formulate the multisensor-multitarget tracking problem. We also state our assumptions, which are identical to those underlying standard algorithms for tracking a known number of targets in the presence of data association uncertainty [1], [3]–[7].

A. Target States and Target Dynamics

We consider K targets $k \in \{1,\dots,K\}$ with time-varying states. The state $\mathbf{x}_{n,k}$ of target k at time n consists of the target's position and possibly further parameters. The number K of targets is assumed fixed and known. The joint target state at time n is denoted as $\mathbf{x}_n \triangleq \big[\mathbf{x}_{n,1}^{\mathrm{T}} \cdots \mathbf{x}_{n,K}^{\mathrm{T}}\big]^{\mathrm{T}}$. The target states are assumed to evolve independently according to Markovian dynamic models [1], [4]. Thus, the joint state-transition function at time n factors as

$$f(\mathbf{x}_n|\mathbf{x}_{n-1}) = \prod_{k=1}^{K} f(\mathbf{x}_{n,k}|\mathbf{x}_{n-1,k})\,. \qquad (1)$$

B. Sensor Measurements

The target states are sensed by S sensors $s \in \{1,\dots,S\}$. At time n, sensor s produces $M_n^{(s)}$ "thresholded" measurements $\mathbf{z}_{n,m}^{(s)}$, $m \in \big\{1,\dots,M_n^{(s)}\big\}$. These measurements are the result of a detection process performed, e.g., by a radar or sonar device. We will also use the stacked measurement vectors $\mathbf{z}_n^{(s)} \triangleq \big[\mathbf{z}_{n,1}^{(s)\mathrm{T}} \cdots \mathbf{z}_{n,M_n^{(s)}}^{(s)\mathrm{T}}\big]^{\mathrm{T}}$ and $\mathbf{z}_n \triangleq \big[\mathbf{z}_n^{(1)\mathrm{T}} \cdots \mathbf{z}_n^{(S)\mathrm{T}}\big]^{\mathrm{T}}$. The measurements are subject to a data association uncertainty, i.e., it is not known which measurement $\mathbf{z}_{n,m}^{(s)}$ originated from which target k, and it is also possible that a measurement $\mathbf{z}_{n,m}^{(s)}$ did not originate from any target (false alarm, clutter) or that a target did not give rise to any measurement of sensor s (missed detection). We assume that at any time n, each target can generate at most one measurement at sensor s, and each measurement at sensor s can be generated by at most one target [1], [4]. The probability that a target is detected by sensor s is denoted by $P_{\mathrm{d}}^{(s)}$. The possible associations at sensor s and time n are described by the K-dimensional random vector $\mathbf{a}_n^{(s)} = \big[a_{n,1}^{(s)} \cdots a_{n,K}^{(s)}\big]^{\mathrm{T}}$, whose kth entry is defined as

$$a_{n,k}^{(s)} \triangleq \begin{cases} m \in \{1,\dots,M_n^{(s)}\}, & \text{at time } n \text{, target } k \text{ generates} \\ & \text{measurement } m \text{ at sensor } s \\ 0, & \text{at time } n \text{, target } k \text{ is not} \\ & \text{sensed by sensor } s. \end{cases}$$

We also define $\mathbf{a}_n \triangleq \big[\mathbf{a}_n^{(1)\mathrm{T}} \cdots \mathbf{a}_n^{(S)\mathrm{T}}\big]^{\mathrm{T}}$. The number of false alarms at sensor s is modeled by a Poisson distribution with mean value $\mu^{(s)}$; however, the proposed method can be easily generalized to an arbitrary distribution. The distribution of each false alarm measurement at sensor s is described by the pdf $f_{\mathrm{FA}}\big(\mathbf{z}_{n,m}^{(s)}\big)$ [1], [4].

The dependence of the measurement vector $\mathbf{z}_n$ on the state vector $\mathbf{x}_n$ and the association vector $\mathbf{a}_n$ is described by the *global likelihood function* $f(\mathbf{z}_n|\mathbf{x}_n,\mathbf{a}_n)$. Under commonly used assumptions about the statistics of the measurements [1], [4], the global likelihood function factors as

$$f(\mathbf{z}_n|\mathbf{x}_n,\mathbf{a}_n) = \prod_{s=1}^{S} \Bigg(\prod_{m=1}^{M_n^{(s)}} f_{\mathrm{FA}}\big(\mathbf{z}_{n,m}^{(s)}\big) \Bigg) \times \prod_{k \in \mathcal{K}_{\mathbf{a}_n}^{(s)}} \frac{f\Big(\mathbf{z}_{n,a_{n,k}^{(s)}}^{(s)} \Big| \mathbf{x}_{n,k}\Big)}{f_{\mathrm{FA}}\Big(\mathbf{z}_{n,a_{n,k}^{(s)}}^{(s)}\Big)}\,,$$

with $\mathcal{K}_{\mathbf{a}_n}^{(s)} \triangleq \big\{k \in \{1,\dots,K\} : a_{n,k}^{(s)} \neq 0\big\}$. For observed (and, thus, fixed) $\mathbf{z}_n$, this can be written as

$$f(\mathbf{z}_n|\mathbf{x}_n,\mathbf{a}_n) \propto \prod_{s=1}^{S} \prod_{k=1}^{K} g\big(\mathbf{x}_{n,k}, a_{n,k}^{(s)}; \mathbf{z}_n^{(s)}\big)\,, \qquad (2)$$

where $\propto$ denotes equality up to a normalization factor and

$$g\big(\mathbf{x}_{n,k}, a_{n,k}^{(s)}; \mathbf{z}_n^{(s)}\big) = \begin{cases} \dfrac{f\big(\mathbf{z}_{n,m}^{(s)}|\mathbf{x}_{n,k}\big)}{f_{\mathrm{FA}}\big(\mathbf{z}_{n,m}^{(s)}\big)}, & a_{n,k}^{(s)} = m \in \{1, \\ & \qquad \dots, M_n^{(s)}\} \\ 1, & a_{n,k}^{(s)} = 0\,. \end{cases}$$

C. Prior Distributions

We assume that for each target state $\mathbf{x}_{0,k}$ at time $n=0$, an informative initial prior pdf $f(\mathbf{x}_{0,k})$ is available. Furthermore, the prior probability mass function of the association vector $\mathbf{a}_n^{(s)}$ is modeled as [1]

$$p\big(\mathbf{a}_n^{(s)}\big) \,\propto\, \psi\big(\mathbf{a}_n^{(s)}\big) \prod_{k=1}^{K} \varphi\big(a_{n,k}^{(s)}\big) \,. \tag{3}$$

Here, $\psi\big(\mathbf{a}_n^{(s)}\big)$ expresses the exclusion assumptions stated in Section II-B, i.e., the fact that each target can generate at most one measurement at sensor s and each measurement at sensor s can be generated by at most one target [1], [4]:

$$\psi\big(\mathbf{a}_n^{(s)}\big) = \begin{cases} 0, & \exists k, k' \in \{1, \dots, K\} \text{ s.t. } a_{k,n}^{(s)} = a_{k',n}^{(s)} \neq 0 \\ 1, & \text{otherwise.} \end{cases} \tag{4}$$

The remaining factors in (3) are given by

$$\varphi\big(a_{n,k}^{(s)}\big) = \begin{cases} \frac{P_{\mathrm{d}}^{(s)}}{\mu^{(s)}}, & a_{n,k}^{(s)} \in \{1, \dots, M_n^{(s)}\} \\ \big(1 - P_{\mathrm{d}}^{(s)}\big), & a_{n,k}^{(s)} = 0 \,. \end{cases}$$

D. Measurement-oriented Association Vectors

Following [17], [18], we also introduce measurement-oriented association vectors $\mathbf{b}_n^{(s)} = \big[b_{n,1}^{(s)} \cdots b_{n,M_n^{(s)}}^{(s)}\big]^{\mathrm{T}}$ with

$$b_{n,m}^{(s)} \triangleq \begin{cases} k \in \{1, \dots, K\}, & \text{at time } n \text{, measurement } m \text{ at sensor } s \text{ is generated by target } k \\ 0, & \text{at time } n \text{, measurement } m \text{ at sensor } s \text{ is not generated by a target.} \end{cases}$$

We also define $\mathbf{b}_n \triangleq \big[\mathbf{b}_n^{(1)\mathrm{T}} \cdots \mathbf{b}_n^{(S)\mathrm{T}}\big]^{\mathrm{T}}$. Note that $\mathbf{b}_n^{(s)}$ can be constructed from $\mathbf{a}_n^{(s)}$ and vice versa. However, using $\mathbf{b}_n^{(s)}$, the exclusion indicator function $\psi\big(\mathbf{a}_n^{(s)}\big)$ in (4) can be formally replaced by the function $\psi\big(\mathbf{a}_n^{(s)}, \mathbf{b}_n^{(s)}\big)$, which can be written as the following product [17], [18], [23]:

$$\psi\big(\mathbf{a}_n^{(s)}, \mathbf{b}_n^{(s)}\big) = \prod_{k=1}^{K} \prod_{m=1}^{M_n^{(s)}} \Psi\big(a_{n,k}^{(s)}, b_{n,m}^{(s)}\big) \,, \tag{5}$$

with

$$\Psi\big(a_{n,k}^{(s)}, b_{n,m}^{(s)}\big) = \begin{cases} 0, & a_{n,k}^{(s)} = m,\, b_{n,m}^{(s)} \neq k \\ & \quad \text{or } b_{n,m}^{(s)} = k,\, a_{n,k}^{(s)} \neq m \\ 1, & \text{otherwise.} \end{cases} \tag{6}$$

Equations (5) and (6) are equivalent to the exclusion relation (4). The factorization (5) will play an important role in the development of our BP algorithm in Sections IV and V.

E. Problem Formulation

The problem addressed in this paper is Bayesian estimation of the states $\mathbf{x}_{n,k}$, $k \in \{1, \dots, K\}$ from all the past and present measurements of all the sensors $s \in \{1, \dots, S\}$, i.e., from the total measurement vector $\mathbf{z} \triangleq \big[\mathbf{z}_1^{\mathrm{T}} \cdots \mathbf{z}_n^{\mathrm{T}}\big]^{\mathrm{T}}$. More specifically, we will develop an approximate calculation of the minimum mean-square error (MMSE) estimator [24]

$$\hat{\mathbf{x}}_{n,k}^{\mathrm{MMSE}} \,\triangleq\, \int \mathbf{x}_{n,k}\, f(\mathbf{x}_{n,k}|\mathbf{z})\, \mathrm{d}\mathbf{x}_{n,k} \,. \tag{7}$$

This calculation will be based on an approximate calculation of the marginal posterior pdfs $f(\mathbf{x}_{n,k}|\mathbf{z})$ involved in (7) by means of an efficient BP message passing algorithm.

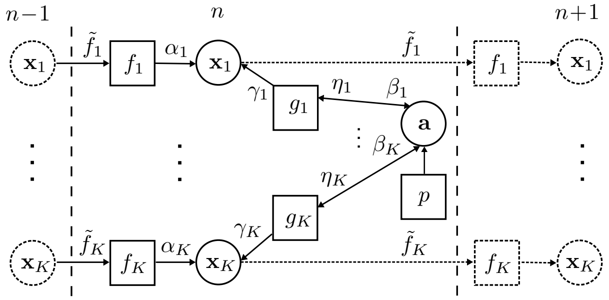

Fig. 1. Factor graph corresponding to the factorization (8) underlying the JPDAF, drawn for time n. The time index (subscript n) is omitted for simplicity. The short notations $f_k \triangleq f(\mathbf{x}_{n,k}|\mathbf{x}_{n-1,k})$, $g_k \triangleq g(\mathbf{x}_{n,k}, a_{n,k}; \mathbf{z}_n)$, $p \triangleq p(\mathbf{a}_n)$, $\tilde{f}_k \triangleq \tilde{f}(\mathbf{x}_{n,k})$, $\alpha_k \triangleq \alpha_{\rightarrow}(\mathbf{x}_{n,k})$, $\beta_k \triangleq \beta_{\rightarrow}(a_{n,k})$, $\eta_k \triangleq \eta_{\rightarrow}(a_{n,k})$, and $\gamma_k \triangleq \gamma_{\rightarrow}(\mathbf{x}_{n,k})$ are used.

III. BP Interpretation of the JPDAF

As a preparatory step for the derivation of our multisensor-multitarget tracking algorithm, we first present a new BP message passing interpretation of the JPDAF [1]. In Section IV, the complexity of this JPDAF BP scheme will be reduced through the data association formulation (5), (6). We temporarily consider only a single sensor and, thus, suppress the sensor index s. Multiple sensors will be considered in Section V.

The marginal posterior pdfs $f(\mathbf{x}_{n,k}|\mathbf{z})$ involved in (7) are marginals of the joint posterior pdf $f(\mathbf{x}, \mathbf{a}|\mathbf{z})$ of the total target vector $\mathbf{x} \triangleq \big[\mathbf{x}_1^{\mathrm{T}} \cdots \mathbf{x}_n^{\mathrm{T}}\big]^{\mathrm{T}}$ and the total association vector $\mathbf{a} \triangleq \big[\mathbf{a}_1^{\mathrm{T}} \cdots \mathbf{a}_n^{\mathrm{T}}\big]^{\mathrm{T}}$. However, straightforward marginalization of $f(\mathbf{x}, \mathbf{a}|\mathbf{z})$ is infeasible because it involves integration in spaces whose dimension grows with time n. A sequential approximate marginalization whose total complexity grows only linearly with n can be obtained by means of BP message passing. Using Bayes' rule and common independence assumptions [1], [4], $f(\mathbf{x}, \mathbf{a}|\mathbf{z})$ can be expressed up to an irrelevant factor as

$$\begin{aligned} f(\mathbf{x}, \mathbf{a}|\mathbf{z}) \,&\propto\, f(\mathbf{z}|\mathbf{x}, \mathbf{a})\, f(\mathbf{x}, \mathbf{a}) \\ &= f(\mathbf{x}_0) \prod_{n'=1}^{n} f(\mathbf{z}_{n'}|\mathbf{x}_{n'}, \mathbf{a}_{n'})\, f(\mathbf{x}_{n'}|\mathbf{x}_{n'-1})\, p(\mathbf{a}_{n'}) \\ &\propto \left(\prod_{k'=1}^{K} f(\mathbf{x}_{0,k'}) \right) \prod_{n'=1}^{n} p(\mathbf{a}_{n'}) \\ &\quad \times \prod_{k=1}^{K} g(\mathbf{x}_{n',k}, a_{n',k}; \mathbf{z}_{n'})\, f(\mathbf{x}_{n',k}|\mathbf{x}_{n'-1,k}) \,, \end{aligned} \tag{8}$$

where (1) and (2) have been used. The factorization structure in (8) is represented graphically by the loopy factor graph [14] depicted in Fig. 1. Running a BP algorithm [14], [16] on this factor graph provides approximations $\tilde{f}(\mathbf{x}_{n,k})$ of the marginal posterior pdfs $f(\mathbf{x}_{n,k}|\mathbf{z})$ (so-called *beliefs*) for all states $\mathbf{x}_{n,k}$. The approximate nature of the beliefs is due to the presence of loops (cycles) in the factor graph; these loops are not visible in Fig. 1 since, essentially, only the part of the factor graph corresponding to one time step n is shown.

In factor graphs with loops, the BP scheme is generally iterative, and there are many possible orders in which messages can be computed. These different computation orders may also lead to different beliefs. Here, we fix the order by sending messages only forward in time [25]. This results in a noniterative BP scheme, and furthermore is equivalent to assuming that the target states are mutually independent conditional on the past measurements, as is done in the derivation of the JPDAF [1]. Then, using the standard BP message passing rules [14], the beliefs $\tilde{f}(\mathbf{x}_{n,k})$ of all target states $\mathbf{x}_{n,k}$, $k \in \mathcal{K} \triangleq \{1, \ldots, K\}$ at time n are obtained by performing the following steps for each $k \in \mathcal{K}$:

1) Prediction:

$$\alpha_{\rightarrow}(\mathbf{x}_{n,k}) = \int f(\mathbf{x}_{n,k}|\mathbf{x}_{n-1,k})\, \tilde{f}(\mathbf{x}_{n-1,k})\, \mathrm{d}\mathbf{x}_{n-1,k} \,. \quad (9)$$

2) Measurement evaluation:

$$\beta_{\rightarrow}(a_{n,k}) = \int g(\mathbf{x}_{n,k}, a_{n,k}; \mathbf{z}_n)\, \alpha_{\rightarrow}(\mathbf{x}_{n,k})\, \mathrm{d}\mathbf{x}_{n,k} \,. \quad (10)$$

3) Data association:

$$\eta_{\rightarrow}(a_{n,k}) = \sum_{\sim a_{n,k}} p(\mathbf{a}_n) \prod_{k' \in \mathcal{K}\backslash\{k\}} \beta_{\rightarrow}(a_{n,k'}) \,. \quad (11)$$

4) Measurement update:

$$\gamma_{\rightarrow}(\mathbf{x}_{n,k}) = \sum_{a_{n,k}} g(\mathbf{x}_{n,k}, a_{n,k}; \mathbf{z}_n)\, \eta_{\rightarrow}(a_{n,k}) \,. \quad (12)$$

5) Calculation of target belief:

$$\tilde{f}(\mathbf{x}_{n,k}) \propto \alpha_{\rightarrow}(\mathbf{x}_{n,k})\, \gamma_{\rightarrow}(\mathbf{x}_{n,k}) \,. \quad (13)$$

Here, the notation $\sum_{\sim a_{n,k}}$ used in (11) indicates summation over all $a_{n,k'}$ with $k' \in \mathcal{K}\backslash\{k\}$. The messages and beliefs in (9)–(13) are visualized in Fig. 1.

Under the assumption of linear and Gaussian measurement and dynamic models, which was used in the original formulation of the JPDAF, closed-form expressions of prediction (9) and measurement evaluation (10) can be found [1]. In addition, the JPDAF uses a Gaussian approximation for the belief $\tilde{f}(\mathbf{x}_{n,k})$ in (13). In the nonlinear/non-Gaussian case, closed-form expressions do not exist but Monte Carlo computations of (9), (10), and (13) can be obtained. However, in either case, a problem is the summation $\sum_{\sim a_{n,k}}$ in the data association step (11), whose complexity is exponential in the number of targets K. This problem will be addressed next.

IV. Efficient BP Algorithm for Data Association

In this section, we review the efficient BP-based data association technique proposed in [17], [18], [23]. This technique is based on (5) and (6), i.e., on the formulation of the exclusion indicator function in terms of both the target-oriented association variables $a_{n,k}^{(s)}$ and the measurement-oriented association variables $b_{n,m}^{(s)}$. As in the previous section, we consider only a single sensor and suppress the sensor index s. The case of multiple sensors will be considered in Section V.

Because there is a one-to-one relation between $\mathbf{a}_n$ and $\mathbf{b}_n$, the prior probability mass function of $\mathbf{a}_n$, $p(\mathbf{a}_n)$, can be formally replaced by $p(\mathbf{a}_n, \mathbf{b}_n)$. Expression (3) then becomes

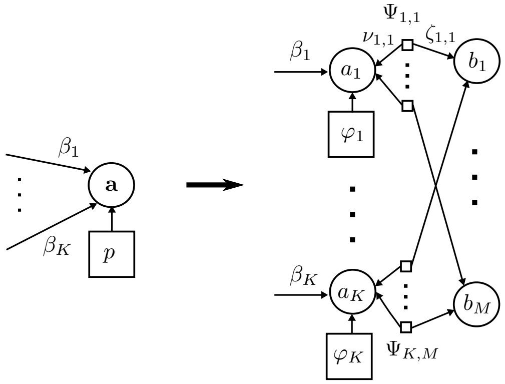

Fig. 2. Modification of the factor graph in Fig. 1 induced by the modified factorization (15). The time index n is omitted for simplicity. The short notations $p \triangleq p(\mathbf{a}_n)$, $\Psi_{k,m} \triangleq \Psi(a_{n,k}, b_{n,m})$, $\beta_k \triangleq \beta_{\rightarrow}(a_{n,k})$, $\varphi_k \triangleq \varphi(a_{n,k})$, $\nu_{m,k} \triangleq \nu_{m\rightarrow k}^{(p)}(a_{n,k})$, and $\zeta_{k,m} \triangleq \zeta_{k\rightarrow m}^{(p)}(b_{n,m})$ are used.

$$\begin{aligned} p(\mathbf{a}_n, \mathbf{b}_n) &\propto \psi(\mathbf{a}_n, \mathbf{b}_n) \prod_{k=1}^{K} \varphi(a_{n,k}) \\ &= \prod_{k=1}^{K} \varphi(a_{n,k}) \prod_{m=1}^{M_n} \Psi(a_{n,k}, b_{n,m}) \,, \end{aligned} \quad (14)$$

where (5) was used in the last step. Similarly, the joint posterior pdf $f(\mathbf{x},\mathbf{a}|\mathbf{z})$ can be replaced by $f(\mathbf{x},\mathbf{a},\mathbf{b}|\mathbf{z})$. (Note that $p(\mathbf{a}_n) = \sum_{\mathbf{b}_n} p(\mathbf{a}_n, \mathbf{b}_n)$ and $f(\mathbf{x},\mathbf{a}|\mathbf{z}) = \sum_{\mathbf{b}} f(\mathbf{x},\mathbf{a},\mathbf{b}|\mathbf{z})$.) As a consequence, in the factorization of $f(\mathbf{x},\mathbf{a}|\mathbf{z})$ in (8), the factors $p(\mathbf{a}_n)$ are replaced by the equivalent factors $p(\mathbf{a}_n, \mathbf{b}_n)$, which in turn factorize as given by (14). Thus, (8) becomes

$$\begin{aligned} f(\mathbf{x},\mathbf{a},\mathbf{b}|\mathbf{z}) \propto \left(\prod_{k'=1}^{K} f(\mathbf{x}_{0,k'}) \right) \prod_{n'=1}^{n} \prod_{k=1}^{K} \varphi(a_{n',k})\, g(\mathbf{x}_{n',k}, a_{n',k}; \mathbf{z}_{n'}) \\ \times f(\mathbf{x}_{n',k}|\mathbf{x}_{n'-1,k}) \prod_{m=1}^{M_{n'}} \Psi(a_{n',k}, b_{n',m}) \,. \end{aligned} \quad (15)$$

This factorization is represented graphically by a modification of the factor graph of Fig. 1 in which the element depicted in the left part of Fig. 2 (consisting of the variable node $\mathbf{a}_n$ and the function node $p(\mathbf{a}_n)$) is replaced as shown in the right part of Fig. 2.

By performing loopy BP on this modified factor graph, the data association operation in (11) is replaced by an iterative scheme [17], [18], [23] which, at message passing iteration $p \in \{1, \ldots, P\}$, calculates

$$\nu_{m\rightarrow k}^{(p)}(a_{n,k}) = \sum_{b_{n,m}} \Psi(a_{n,k}, b_{n,m}) \prod_{k' \in \mathcal{K}\backslash\{k\}} \zeta_{k'\rightarrow m}^{(p-1)}(b_{n,m}) \quad (16)$$

and

$$\begin{aligned} \zeta_{k\rightarrow m}^{(p)}(b_{n,m}) = \sum_{a_{n,k}} \beta_{\rightarrow}(a_{n,k})\, \varphi(a_{n,k})\, \Psi(a_{n,k}, b_{n,m}) \\ \times \prod_{m' \in \mathcal{M}_n\backslash\{m\}} \nu_{m'\rightarrow k}^{(p)}(a_{n,k}) \,, \end{aligned} \quad (17)$$

for all $k \in \mathcal{K}$ and all $m \in \mathcal{M}_n \triangleq \{1, \ldots, M_n\}$. This scheme is initialized by

$$\zeta^{(0)}_{k \to m}(b_{n,m}) = \sum_{a_{n,k}} \beta_{\rightarrow}(a_{n,k})\, \varphi(a_{n,k})\, \Psi(a_{n,k}, b_{n,m})\,. \quad (18)$$

An efficient implementation of (16) and (17) can be found in [18], [23]. After P iterations, an approximation of the "data association message" $\eta_{\rightarrow}(a_{n,k})$ in (11) is obtained as

$$\tilde{\eta}_{\rightarrow}(a_{n,k}) = \varphi(a_{n,k}) \prod_{m \in \mathcal{M}_n} \nu^{(P)}_{m \to k}(a_{n,k})\,. \quad (19)$$

While iterative loopy BP is not guaranteed to converge in general, convergence of the iterative scheme in (16)–(18) has been proven in [18], [23]. Therefore, instead of performing a fixed number P of iterations, a convergence-related stopping criterion can be used. The complexity of the iterative scheme in (16), (17) (with P fixed) scales as $\mathcal{O}(KM_n)$ [18]. Thus, the main limitation of the original formulation of the JPDAF (and of its BP message passing reformulation in Section III)—the exponential scaling of complexity with respect to K—has been mitigated significantly.

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

We now develop the proposed BP-based multisensor-multitarget tracking algorithm by extending the scheme described in Sections III and IV to multiple sensors. As in Section II, we assume that there are S sensors $s \in \{1, \ldots, S\}$.

A. Factor Graph

Our goal is to compute the marginal posterior pdfs $f(\mathbf{x}_{n,k}|\mathbf{z})$ occurring in (7). These pdfs are marginals of the joint posterior pdf $f(\mathbf{x}, \mathbf{a}, \mathbf{b}|\mathbf{z})$, which involves all the time steps, all the target states, all the target-oriented and measurement-oriented association variables, and all the measurements of all the sensors. Using Bayes' rule and common independence assumptions [1], [4], the joint posterior pdf can be expressed up to an irrelevant factor as

$$\begin{aligned} f(\mathbf{x}, \mathbf{a}, \mathbf{b}|\mathbf{z}) &\propto f(\mathbf{z}|\mathbf{x}, \mathbf{a}, \mathbf{b})\, f(\mathbf{x}, \mathbf{a}, \mathbf{b}) \\ &= f(\mathbf{z}|\mathbf{x}, \mathbf{a})\, f(\mathbf{x})\, p(\mathbf{a}, \mathbf{b}) \\ &= f(\mathbf{x}_0) \prod_{n'=1}^{n} f(\mathbf{z}_{n'}|\mathbf{x}_{n'}, \mathbf{a}_{n'})\, f(\mathbf{x}_{n'}|\mathbf{x}_{n'-1}) \\ &\qquad\qquad \times\, p(\mathbf{a}_{n'}, \mathbf{b}_{n'}) \\ &\propto \left(\prod_{k'=1}^{K} f(\mathbf{x}_{0,k'}) \right) \prod_{n'=1}^{n} \prod_{k=1}^{K} f(\mathbf{x}_{n',k}|\mathbf{x}_{n'-1,k}) \\ &\qquad \times \prod_{s=1}^{S} \varphi\big(a^{(s)}_{n',k}\big)\, g\big(\mathbf{x}_{n',k}, a^{(s)}_{n',k}; \mathbf{z}^{(s)}_{n'}\big) \\ &\qquad \times \prod_{m=1}^{M^{(s)}_{n'}} \Psi\big(a^{(s)}_{n',k}, b^{(s)}_{n',m}\big)\,, \end{aligned} \quad (20)$$

where (1), (2), and (14) have been used. The factor graph corresponding to this factorization is shown for one time step in Fig. 3. Note the relations to Figs. 1 and 2.

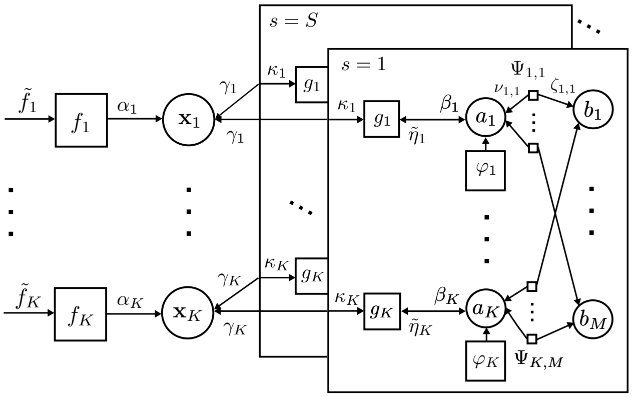

Fig. 3. Factor graph corresponding to the factorization (20) underlying the proposed algorithm, drawn for one time step. The time index n and sensor index s are omitted for simplicity. The short notations $f_k \triangleq f(\mathbf{x}_{n,k}|\mathbf{x}_{n-1,k})$, $g_k \triangleq g\big(\mathbf{x}_{n,k}, a^{(s)}_{n,k}; \mathbf{z}^{(s)}_n\big)$, $\tilde{f}_k \triangleq \tilde{f}(\mathbf{x}_{n,k})$, $\Psi_{k,m} \triangleq \Psi\big(a^{(s)}_{n,k}, b^{(s)}_{n,m}\big)$, $\alpha_k \triangleq \alpha_{\rightarrow}(\mathbf{x}_{n,k})$, $\beta_k \triangleq \beta^{(q)}_{\rightarrow}\big(a^{(s)}_{n,k}\big)$, $\varphi_k \triangleq \varphi\big(a^{(s)}_{n,k}\big)$, $\tilde{\eta}_k \triangleq \tilde{\eta}^{(q)}_{\rightarrow}\big(a^{(s)}_{n,k}\big)$, $\gamma_k \triangleq \gamma^{(q)}_{s\rightarrow}(\mathbf{x}_{n,k})$, $\kappa_k \triangleq \kappa^{(q)}_{\rightarrow s}(\mathbf{x}_{n,k})$, $\nu_{m,k} \triangleq \nu^{(q,p)}_{m \to k}\big(a^{(s)}_{n,k}\big)$, and $\zeta_{k,m} \triangleq \zeta^{(q,p)}_{k \to m}\big(b^{(s)}_{n,m}\big)$ are used.

B. BP Message Passing Algorithm

To obtain an approximation of the marginal posterior pdfs $f(\mathbf{x}_{n,k}|\mathbf{z})$, we propose to run the following iterative BP message passing algorithm on the factor graph in Fig. 3. The algorithm consists of an outer iteration loop for fusing the information from different sensors (obtained using (10) and (12)) and an inner iteration loop for iterative data association using (16), (17), and (19). At time n, first the prediction step (9) is performed for all targets $k \in \mathcal{K}$,

$$\alpha_{\rightarrow}(\mathbf{x}_{n,k}) = \int f(\mathbf{x}_{n,k}|\mathbf{x}_{n-1,k})\, \tilde{f}^{(Q)}(\mathbf{x}_{n-1,k})\, \mathrm{d}\mathbf{x}_{n-1,k}\,, \quad (21)$$

where $\tilde{f}^{(Q)}(\mathbf{x}_{n-1,k})$ is the final result of the outer iteration loop at the previous time $n-1$. Then, the outer BP message passing iteration loop is executed. Specifically, in outer iteration $q \in \{1, \ldots, Q\}$, the following steps are performed for all targets $k \in \mathcal{K}$ and all sensors $s \in \mathcal{S} \triangleq \{1, \ldots, S\}$ in parallel:

1) Measurement evaluation (cf. (10)):

$$\beta^{(q)}_{\rightarrow}\big(a^{(s)}_{n,k}\big) = \int g\big(\mathbf{x}_{n,k}, a^{(s)}_{n,k}; \mathbf{z}^{(s)}_n\big)\, \kappa^{(q-1)}_{\rightarrow s}(\mathbf{x}_{n,k})\, \mathrm{d}\mathbf{x}_{n,k}\,. \quad (22)$$

2) Iterative data association (cf. (19)):

$$\tilde{\eta}^{(q)}_{\rightarrow}\big(a^{(s)}_{n,k}\big) = \varphi\big(a^{(s)}_{n,k}\big) \prod_{m \in \mathcal{M}^{(s)}_n} \nu^{(q,P)}_{m \to k}\big(a^{(s)}_{n,k}\big)\,, \quad (23)$$

where $\nu^{(q,P)}_{m \to k}\big(a^{(s)}_{n,k}\big)$ is the result of the inner BP message passing iteration loop. Specifically, in inner iteration $p \in \{1, \ldots, P\}$, the following steps are performed for all measurements $m \in \mathcal{M}^{(s)}_n$ (cf. (16), (17)):

$$\nu^{(q,p)}_{m \to k}\big(a^{(s)}_{n,k}\big) = \sum_{b^{(s)}_{n,m}} \Psi\big(a^{(s)}_{n,k}, b^{(s)}_{n,m}\big) \prod_{k' \in \mathcal{K} \setminus \{k\}} \zeta^{(q,p-1)}_{k' \to m}\big(b^{(s)}_{n,m}\big) \quad (24)$$

and

$$\zeta^{(q,p)}_{k\to m}\big(b^{(s)}_{n,m}\big) \,=\, \sum_{a^{(s)}_{n,k}} \beta^{(q)}_{\to}\big(a^{(s)}_{n,k}\big)\,\varphi\big(a^{(s)}_{n,k}\big)\,\Psi\big(a^{(s)}_{n,k},b^{(s)}_{n,m}\big) \times \prod_{m'\in\mathcal{M}^{(s)}_n\setminus\{m\}} \nu^{(q,p)}_{m'\to k}\big(a^{(s)}_{n,k}\big). \quad (25)$$

This inner iteration loop is initialized by (cf. (18))

$$\zeta^{(q,0)}_{k\to m}\big(b^{(s)}_{n,m}\big) \,=\, \sum_{a^{(s)}_{n,k}} \beta^{(q)}_{\to}\big(a^{(s)}_{n,k}\big)\,\varphi\big(a^{(s)}_{n,k}\big)\,\Psi\big(a^{(s)}_{n,k},b^{(s)}_{n,m}\big).$$

3) Measurement update (cf. (12)):

$$\gamma^{(q)}_{s\to}(\mathbf{x}_{n,k}) \,=\, \sum_{a^{(s)}_{n,k}} g\big(\mathbf{x}_{n,k},a^{(s)}_{n,k};\mathbf{z}^{(s)}_n\big)\,\tilde{\eta}^{(q)}_{\to}\big(a^{(s)}_{n,k}\big). \quad (26)$$

4) Extrinsic information:

$$\kappa^{(q)}_{\to s}(\mathbf{x}_{n,k}) \,=\, \alpha_{\to}(\mathbf{x}_{n,k}) \prod_{s'\in\mathcal{S}\setminus\{s\}} \gamma^{(q)}_{s'\to}(\mathbf{x}_{n,k}). \quad (27)$$

Note that (27) need not be calculated at the last iteration $q=Q$. The outer iteration loop is initialized by

$$\kappa^{(0)}_{\to s}(\mathbf{x}_{n,k}) \,=\, \alpha_{\to}(\mathbf{x}_{n,k}),$$

which was calculated in the prediction step (21). After termination of the outer iteration loop ($q\!=\!Q$), the beliefs $\tilde{f}^{(Q)}(\mathbf{x}_{n,k})$ approximating the marginal posterior pdfs $f(\mathbf{x}_{n,k}|\mathbf{z})$ are finally obtained up to a normalization as (cf. (13))

$$\tilde{f}^{(Q)}(\mathbf{x}_{n,k}) \,\propto\, \alpha_{\to}(\mathbf{x}_{n,k}) \prod_{s\in\mathcal{S}} \gamma^{(Q)}_{s\to}(\mathbf{x}_{n,k}). \quad (28)$$

C. Discussion

The inner "data association" iteration loop (23)–(25) involves solely messages related to discrete random variables. Since it is based on loopy BP [14], [16], it does not perform an exact calculation of the marginalization (11). However, its high accuracy has been demonstrated numerically [17], [18] (see also Section VI), and furthermore its convergence has been proven [18], [23]. The outer "sensor fusion" iteration loop involves messages related to discrete or continuous random variables. Because one complete inner iteration loop has to be executed for every outer iteration, a small number Q of outer iterations is desirable. In the first outer iteration $q\!=\!1$, each sensor uses only the predicted information about the target states to perform local data association. This simple fusion strategy is equivalent to that employed by the parallel multisensor JPDAF [7] and the multisensor MC-JPDAF [4]. In all subsequent outer iterations $q=2,\ldots,Q$, the local data association performed at each sensor also uses information about the target states that is provided by the measurements of all the other sensors (see (27)).

The main advantage of the proposed algorithm is its scaling characteristic. For fixed numbers P and Q of inner and outer iterations, the computational complexity of calculating the (approximate) marginal posterior pdfs of all the target states is linear in the number of sensors S (see (27) and (28)). Furthermore, the complexity of the operations (22)–(26) performed for each sensor $s\in\mathcal{S}$ scales as $\mathcal{O}(KM^{(s)}_n)$, where $M^{(s)}_n$ increases linearly with K and with the number of false alarms. This means that the overall complexity of our algorithm scales linearly in the number of sensors and quadratically in the number of targets. This is a consequence of the "detailed" factorization (20) in terms of both target-related and measurement-related association variables. Using this factorization, an increase in the number of targets, sensors, or measurements leads to additional variable nodes in the factor graph (see Fig. 3) but not to a higher dimension of the messages passed between the nodes.

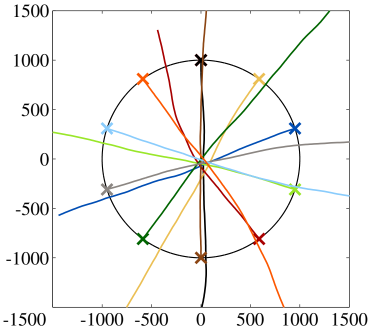

Fig. 4. Example trajectories for $K=10$ targets. The initial target positions are indicated by crosses.

VI. Simulation Results

We compare the performance of our method with that of two state-of-the-art methods for multisensor-multitarget tracking, and we study how the runtime of our method depends on the number of sensors and on the number of targets.

A. Simulation Setup

The target states consist of the targets' two-dimensional (2D) position and 2D 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 region of interest (ROI) is $[-3000, 3000] \times [-3000, 3000]$. All targets move according to the constant-velocity motion model, i.e.,

$$\mathbf{x}_{n,k} \,=\, \mathbf{A}\,\mathbf{x}_{n-1,k} + \mathbf{W}\mathbf{u}_{n,k}\,,$$

where the matrices $\mathbf{A}\in\mathbb{R}^{4\times4}$ and $\mathbf{W}\in\mathbb{R}^{4\times2}$ are chosen as in [26] and $\mathbf{u}_{n,k}\sim\mathcal{N}(\mathbf{0},\sigma_u^2\mathbf{I}_2)$ is an independent and identically distributed (iid) 2D Gaussian random vector with component variance $\sigma_u^2 = 0.1$. To show the benefits of the proposed method and of the use of multiple sensors, we simulated a challenging scenario where all the target trajectories intersect at the ROI center. To obtain this behavior, the targets start at positions (at time $n=0$) that are uniformly placed on a circle of radius 1000, and then move with an initial speed of 20 toward the ROI center. Example trajectories for $K=10$ targets are shown in Fig. 4. We note that the case of intersecting targets is not only difficult in terms of performance but also a worst case in terms of computational complexity, because gating has no effect at the time points where the targets intersect.

The target-generated measurements $\mathbf{z}_{n,m}^{(s)}$ are distributed according to an additive-Gaussian model. That is, for $b_{n,m}^{(s)} = k \neq 0$, we have

$$\mathbf{z}_{n,m}^{(s)} = [x_{1,n,k} \;\; x_{2,n,k}]^{\mathrm{T}} + \mathbf{v}_{n,m}^{(s)} \,,$$

where $\mathbf{v}_{n,m}^{(s)} \sim \mathcal{N}(\mathbf{0}, \sigma_v^2 \mathbf{I}_2)$ is iid 2D Gaussian measurement noise with component standard deviation $\sigma_v = 75$. The false alarm pdf $f_{\mathrm{FA}}\big(\mathbf{z}_{n,m}^{(s)}\big)$ (for $b_{n,m}^{(s)} = 0$) is uniform on the ROI, and the mean number of false alarms is $\mu^{(s)} = \mu = 50$ unless noted otherwise. The detection probability is chosen as $P_{\mathrm{d}}^{(s)} = P_{\mathrm{d}} = 0.3$. Unless noted otherwise, each sensor performs gating with gate threshold 9.2 [1].

In the proposed BP scheme, we use a particle-based implementation (similar to [4]) of the integrations in (21) and (22) and of the message multiplications in (27) and (28). The numbers of inner and outer iterations are $P = 20$ and[1] $Q = 1$, respectively. At time step n, the result of this particle-based BP scheme are K sets of J particles and weights $\big\{\big(\mathbf{x}_{n,k}^{(j)}, w_{n,k}^{(j)}\big)\big\}_{j=1}^{J}$, $k \in \{1, \ldots, K\}$ representing the beliefs $\tilde{f}(\mathbf{x}_{n,k})$ of the targets k. From these particles and weights, approximations of the MMSE estimates in (7) are obtained as

$$\hat{\mathbf{x}}_{n,k} = \sum_{j=1}^{J} w_{n,k}^{(j)} \mathbf{x}_{n,k}^{(j)} \,.$$

We simulated 100 time steps and performed 100 simulation runs.

B. Performance Comparison

For a scenario with $K = 5$ targets and $S = 10$ sensors, we compare the proposed particle-based algorithm with the sequential multisensor JPDAF [6], [7] (which is based on a Gaussian approximation) and with the multisensor MC-JPDAF [4] (which is particle-based). Note that both reference methods are infeasible for more than about $K = 7$ intersecting targets due to their exponential scaling behavior. Both particle-based algorithms use $J = 10.000$ particles. The algorithms are initialized with the initial prior pdf $f(\mathbf{x}_{0,k}) = \mathcal{N}(\boldsymbol{\mu}_{0,k}, \mathbf{C}_{0,k})$. Here, $\mathbf{C}_{0,k} = \mathrm{diag}\{150^2, 150^2, 0.1^2, 0.1^2\}$ represents the uncertainty in knowing $\mathbf{x}_{0,k}$, and $\boldsymbol{\mu}_{0,k}$ is a random hyperparameter that is sampled for each simulation run from $\mathcal{N}(\mathbf{x}_{0,k}, \mathbf{C}_{0,k})$, where $\mathbf{x}_{0,k}$ is the true initial state used for generating the target trajectories. The performance of the three methods is evaluated in terms of the OSPA metric [27], whose calculation is based on the Euclidean distance. Fig. 5 shows the mean OSPA error (MOSPA) of the methods versus time n. The increase of the MOSPA around time $n = 55$ is due to the fact that, at approximately $n = 50$, all targets intersect at the ROI center. It can be seen that the proposed method consistently outperforms the sequential multisensor JPDAF. Furthermore, it performs similar to the multisensor MC-JPDAF, with only a slight performance loss around the intersection time. Thus, the good scaling properties of our method come at the cost of only a very small reduction of tracking performance relative to the multisensor MC-JPDAF, whose complexity is exponential in the number of intersecting targets.

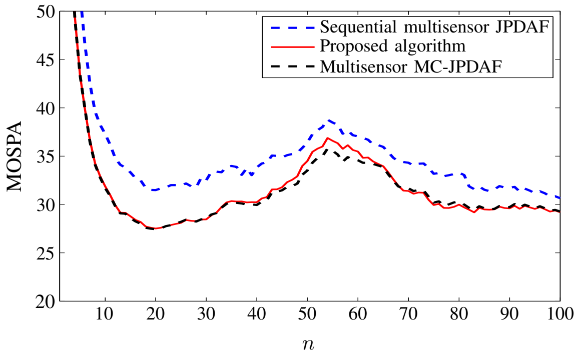

Fig. 5. MOSPA of three multisensor-multitarget tracking algorithms versus time.

C. Scalability

Finally, we investigate how the runtime of a MATLAB implementation of our algorithm on a single core of an Intel Xeon X5650 CPU scales with the number of sensors S and the number of targets K. We also study the dependence of the MOSPA on S. To ensure that the initial prior pdfs of the targets are well separated even for $K = 30$, we now use smaller initial position variances, i.e., we set $\mathbf{C}_{0,k} = \mathrm{diag}\{10^2, 10^2, 0.1^2, 0.1^2\}$. Both the runtimes and the OSPA errors are averaged over 100 time steps and 100 simulation runs. The number of particles is now only $J = 1000$.

For $K = 5$ intersecting targets, Fig. 6 shows the runtime per time step n and the MOSPA versus $S \in \{2, 4, \ldots, 20\}$. These results confirm the linear scaling of the runtime with S and demonstrate the improvement in tracking performance that results from using more sensors.

For $S = 10$ sensors, Fig. 7 shows the runtime per time step n versus $K \in \{2, 4, \ldots, 30\}$. To reduce the simulation time, the mean number of false alarms is now chosen as

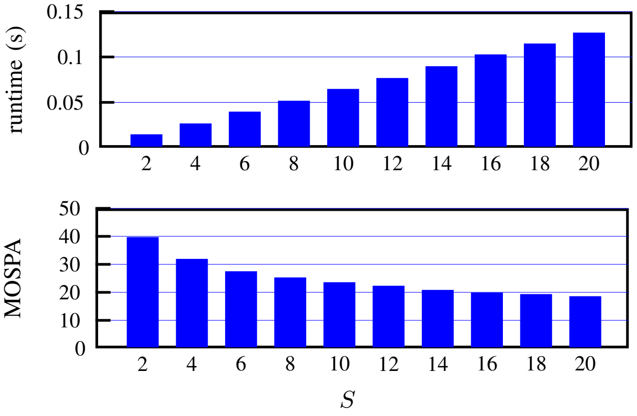

Fig. 6. Runtime per time step (top) and MOSPA (bottom) of the proposed algorithm versus number of sensors for $K = 5$ intersecting targets.

[1] In the scenarios simulated in this paper, we did not observe any performance benefits for $Q > 1$. However, this may be different when the sensor characteristics vary across the sensors.

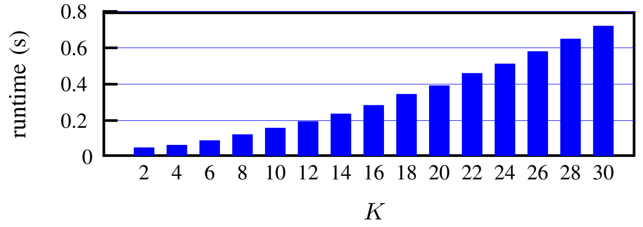

Fig. 7. Runtime per time step of the proposed algorithm versus number of targets for $S = 10$ sensors.

$\mu = 5$. Furthermore, in order to demonstrate a worst-case complexity, we do not perform gating. As can be seen in Fig. 7, this worst-case scaling behavior in K is roughly quadratic. The low complexity of the proposed method is evidenced by the fact that for $K = 30$ targets and $S = 10$ sensors, the computations per time step n require less than one second. For $K \in \{2, 4, 6, 8\}$, the MOSPA was observed to be approximately 20. For $K > 8$, calculating the MOSPA is infeasible. However, for $K \leq 16$, we did not observe any lost tracks, and for $K = 30$, the lost tracks still amounted to less than 5% of the total number of tracks.

VII. Conclusion

We proposed a belief propagation based framework and algorithm for multisensor-multitarget tracking when the number of targets is known. Excellent scalability in the number of targets, number of sensors, and number of measurements per sensor is achieved by using a "detailed" factor graph that involves both target-related and measurement-related association variables. Despite its low complexity even for a large number of targets and sensors, the proposed algorithm can outperform the sequential multisensor JPDAF, whose complexity is exponential in the number of intersecting targets.

Possible directions for future research include an extension of the proposed algorithm to an unknown number of targets via the introduction of Bernoulli random variables modeling target existence, and the development of a distributed version of the algorithm using consensus techniques (cf. [28]). It would also be interesting to study the performance of the algorithm for sensors with different characteristics.

Acknowledgments

F. Meyer and F. Hlawatsch were supported by the FWF via grant P27370-N30. P. Braca thanks the NATO Allied Command Transformation (NATO-ACT) for supporting the CMRE project Data Knowledge Operational Effectiveness. P. Willett was supported by the NPS via grant ONR N00244-14-1-0033, and by the ONR directly via grant N00014-13-1-0231.

References

[1] Y. Bar-Shalom and X.-R. Li, *Multitarget-Multisensor Tracking: Principles and Techniques*. Storrs, CT: Yaakov Bar-Shalom, 1995.

[2] R. Mahler, *Statistical Multisource-Multitarget Information Fusion*. Norwood, MA: Artech House, 2007.

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

[4] J. Vermaak, S. J. Godsill, and P. Perez, "Monte Carlo filtering for multitarget tracking and data association," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 41, no. 1, pp. 309–332, Jan. 2005.

[5] 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.

[6] M. Kalandros and L. Y. Pao, "Multisensor covariance control strategies for reducing bias effects in interacting target scenarios," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 41, no. 1, pp. 153–173, Jan. 2005.

[7] C. Frei and L. Y. Pao, "Alternatives to Monte-Carlo simulation evaluations of two multisensor fusion algorithms," *Automatica*, vol. 34, no. 1, pp. 103–110, 1998.

[8] R. Mahler, "Statistics 102 for multisource-multitarget detection and tracking," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 3, pp. 376–389, Jun. 2013.

[9] 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.

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

[11] 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, Apr. 2011, pp. 26–28.

[12] S. Maresca, P. Braca, J. Horstmann, and R. Grasso, "Maritime surveillance using multiple high-frequency surface-wave radars," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 8, pp. 5056–5071, Aug. 2014.

[13] G. Papa, P. Braca, S. Horn, S. Marano, V. Matta, and P. Willett, "Adaptive Bayesian tracking under time-varying target detection capability of sensor networks with measurement origin uncertainty," *IEEE Trans. Signal Process.*, 2015, submitted.

[14] 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.

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

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

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

[18] J. 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.

[19] G. W. Pulford and B. F. La Scala, "Multihypothesis Viterbi data association: Algorithm development and assessment," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 46, no. 2, pp. 583–609, Apr. 2010.

[20] 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, pp. 1–8.

[21] 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.

[22] 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, Jul. 2009, pp. 826–833.

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

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

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

[26] J. H. Kotecha and P. M. Djuric, "Gaussian particle filtering," *IEEE Trans. Signal Process.*, vol. 51, no. 10, pp. 2592–2601, Oct 2003.

[27] 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.

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