

A BP Method for Track-Before-Detect

Mingchao Liang , *Student Member, IEEE*, Thomas Kropfreiter , *Member, IEEE*, and Florian Meyer , *Member, IEEE*

Abstract—Tracking an unknown number of low-observable objects is notoriously challenging. This letter proposes a sequential Bayesian estimation method based on the track-before-detect (TBD) approach. In TBD, raw sensor measurements are directly used by the tracking algorithm without any preprocessing. Our proposed method is based on a new statistical model that introduces a new object hypothesis for each data cell of the raw sensor measurements. It allows objects to interact and contribute to more than one data cell. Based on the factor graph representing our statistical model, we derive the message passing equations of the proposed belief propagation (BP) method for TBD. Approximations are applied to certain BP messages to reduce computational complexity and improve scalability. In a simulation experiment, our proposed BP-based TBD method outperforms two other state-of-the-art TBD methods.

Index Terms—Belief propagation, factor graph, multi-object tracking, track-before-detect.

I. INTRODUCTION

MULTI-OBJECT tracking (MOT) [1], [2], [3], [4], [5], [6] aims at estimating the number and states of a time-varying number of objects from noisy sensor measurements. Possible applications of MOT include applied ocean science [7], indoor localization [8], and autonomous driving [9]. In the conventional *detect-then-track* approach, a detection stage preprocesses the raw sensor data in order to reduce data flow and computational complexity. The resulting “point measurements” are then the input to the tracking stage. However, this preprocessing leads to a loss of relevant information and thus to a reduced tracking performance, especially in low signal-to-noise ratio (SNR) scenarios.

In *Track-before-detect* (TBD) methods, raw sensor data is passed directly to the tracking stage without any preprocessing. State-of-the-art TBD methods may be distinguished between batch processing approaches and sequential Bayesian estimation methods. Batch processing approaches include methods based on maximum likelihood estimation [10], the Hough transform [11], and dynamic programming [12]. On the other hand, sequential Bayesian estimation methods perform estimation of object states represented either by random vectors [13], [14], [15], [16], [17] or random finite sets (RFS) [18], [19], [20], [21], [22], [23], [24]. While vector-based approaches often rely

Manuscript received 21 May 2023; revised 3 July 2023; accepted 10 July 2023. Date of publication 19 July 2023; date of current version 30 August 2023. This work was supported by the National Science Foundation through CAREER under Grant 2146261. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Yue Deng. (*Corresponding author: Mingchao Liang.*)

The authors are with the University of California San Diego, La Jolla, CA 92093 USA (e-mail: m3liang@ucsd.edu; tkropfreiter@ucsd.edu; flmeyer@ucsd.edu).

Digital Object Identifier 10.1109/LSP.2023.3296874

on particle filtering methods [13], [14], [15], [16], set-based methods are mostly based on the Bernoulli filter for single object tracking [18], [19] or its generalizations for multi-object tracking [20], [21], [22], [23], [24]. Recently introduced TBD methods that are suitable for tracking an unknown number of objects (i) assume non-interacting objects, i.e., regions of measurements influenced by different objects do not overlap [20]; (ii) rely on heuristics to introduce newborn objects [22], [23], [24]; or (iii) assume that every object can contribute to at most one measurement [21].

Belief propagation (BP) [25], [26], [27], [28], also known as the sum-product algorithm, is a versatile and efficient method for performing inference in Bayesian estimation problems. More precisely, BP has already been used very successfully in detect-then-track MOT problems [4], [5]. Here, the statistical model underlying the MOT problem is represented by a so-called factor graph. By computing local messages and sending them along the edges through the graph, the structure of the MOT model can be exploited in order to reduce computational complexity and increase scalability [4], [5].

In this letter, we propose a BP method for TBD. More precisely, we propose a new statistical model for the TBD problem, develop the corresponding factor graph, and perform inference by applying BP on that graph. Our statistical model includes a new measurement model, in which interacting objects can contribute to more than one data cell. In fact, this new measurement model can be considered a generalization of other models used in existing TBD methods. Furthermore, a new object hypothesis, referred to as potential object (PO), is introduced for every cell measurement. To reduce computational complexity, certain BP messages are approximated by Gaussian probability density functions (PDFs) using moment matching. This approximation is similar to the one performed within the RFS-based TBD method with heuristic track initialization in [23]. The main contributions of this letter can be summarized as follows:

- We propose a new statistical model for TBD MOT consisting of a new measurement model for interacting objects and a new model for object birth.
- We derive a scalable BP inference method and demonstrate its improved performance compared to two other state-of-the-art TBD algorithms.

To the best of the author’s knowledge, the presented approach is the first TBD method based on BP.

II. SYSTEM MODEL

We model the multi-object state at discrete time k by N_k POs [4], [5], where the existence of each PO $n \in \{1, \dots, N_k\}$ is described by the binary random variable $r_{k,n} \in \{0, 1\}$. Here, $r_{k,n} = 1$ indicates that PO n exists. The kinematic state of PO n is modeled by the random vector $\mathbf{x}_{k,n}$ whose entries describe

the object's position, the object's intensity, and possibly further kinematic properties of the object. We define the state of PO n by $\mathbf{y}_{k,n} = [\mathbf{x}_{k,n}^T \ r_{k,n}]^T$ and the joint state of all POs by $\mathbf{y}_k = [\mathbf{y}_{k,1}^T \cdots \mathbf{y}_{k,N_k}^T]^T$.

A. Superpositional Measurement Model

Our measurement model consists of J data cells, where each cell j can be associated with a time-varying intensity measurement $\mathbf{z}_{k,j}$. We stack all measurements $\mathbf{z}_{k,j}$ into the joint measurement vector $\mathbf{z}_k = [\mathbf{z}_{k,1}^T \cdots \mathbf{z}_{k,J}^T]^T$. Note that the positional information of $\mathbf{z}_{k,j}$ is encoded by its index j . We now model $\mathbf{z}_{k,j}$, according to

$$\mathbf{z}_{k,j} = \sum_{n=1}^{N_k} r_{k,n} \mathbf{h}_{j,k,n} + \boldsymbol{\epsilon}_{k,j}. \quad (1)$$

Here, $\mathbf{h}_{j,k,n} \in \mathbb{R}^d$ is the contribution of PO n to measurement $\mathbf{z}_{k,j}$ if PO n exists, i.e., if $r_{k,n} = 1$. We model $\mathbf{h}_{j,k,n}$ by the Gaussian PDF $f(\mathbf{h}_{j,k,n} | \mathbf{x}_{k,n}) = \mathcal{N}(\mathbf{h}_{j,k,n}; \boldsymbol{\mu}_j(\mathbf{x}_{k,n}), \mathbf{C}_j(\mathbf{x}_{k,n}))$ whose mean $\boldsymbol{\mu}_j(\mathbf{x}_{k,n})$ and covariance matrix $\mathbf{C}_j(\mathbf{x}_{k,n})$ define a point spread function [19], [20], [21], [22]. We assume that $\mathbf{h}_{j,k,n}$ is statistically independent for all k, j , and n , and also independent of $\boldsymbol{\epsilon}_{k,j}$. Note that for $r_{k,n} = 0$, PO n does not exist and hence does not contribute to any measurement. Furthermore, the additive noise component $\boldsymbol{\epsilon}_{k,j}$ is modeled as Gaussian with zero mean and covariance \mathbf{C}_ϵ . It is further assumed statistically independent across all k and j .

From measurement (1), we can directly infer the conditional PDFs $f(\mathbf{z}_{k,j} | \mathbf{y}_k)$ by using the fact that the sum of statistically independent Gaussian variables, i.e., in our case all the $\mathbf{h}_{j,k,n}$ and $\boldsymbol{\epsilon}_{k,j}$, is again a Gaussian variable. Furthermore, since $\mathbf{h}_{j,k,n}$ and $\boldsymbol{\epsilon}_{k,j}$ are also conditionally independent for all j given \mathbf{y}_k , all the measurements $\mathbf{z}_{k,j}$ are in turn conditionally independent given \mathbf{y}_k . This leads to the joint likelihood function given by $f(\mathbf{z}_k | \mathbf{y}_k) = \prod_{j=1}^J f(\mathbf{z}_{k,j} | \mathbf{y}_k)$. Note that for some types of objects, e.g., objects with plane surfaces, the assumption of independent measurements $\mathbf{z}_{k,j}$ does not hold because the intensity values of neighboring data cells are usually correlated. However, most TBD algorithms rely on this independence assumption, which is required for efficient estimation [19], [20], [21], [22], [23].

Note that our superpositional model in (1) generalizes many TBD measurement models in the literature. In particular, by setting $\mathbf{C}_j(\mathbf{x}_{k,n}) = \mathbf{0}$, our model reduces to the model used in [20], [23], by setting $d = 2$, $\boldsymbol{\mu}_j(\mathbf{x}_{k,n}) = \mathbf{0}$, and $\mathbf{C}_\epsilon = \sigma_\epsilon^2 \mathbf{I}_d$, it is equivalent to the Rayleigh model in [21], and by additionally assuming that σ_ϵ^2 is Gamma distributed, it is equal to the model in [19], [22].

B. State-Transition and Birth Model

It is assumed that the legacy PO states $\mathbf{y}_{k-1,n}$, $n \in \{1, \dots, N_{k-1}\}$ evolve independently in time [1]. Thus, the joint state transition function can be factored according to [5, Sec. VIII-C] $f(\mathbf{y}_k | \mathbf{y}_{k-1}) = \prod_{n=1}^{N_{k-1}} f(\mathbf{y}_{k,n} | \mathbf{y}_{k-1,n})$, where we have introduced $\mathbf{y}_k = [\mathbf{y}_{k,1}^T \cdots \mathbf{y}_{k,N_{k-1}}^T]^T$.

To account for newly appearing objects, we introduce, at each time k , J new POs with states $\mathbf{y}_{k,n}$, $n \in \{N_{k-1} + 1, \dots, N_k\}$, one new PO for each data cell j . Thus, $N_k = N_{k-1} + J$. We define $\bar{\mathbf{y}}_k = [\mathbf{y}_{k,N_{k-1}+1}^T \cdots \mathbf{y}_{k,N_k}^T]^T$ and assume that new PO

states are independent, i.e., $f(\bar{\mathbf{y}}_k) = \prod_{n=N_{k-1}+1}^{N_k} f(\mathbf{y}_{k,n})$. We furthermore assume that the statistics of $\mathbf{y}_{k,n}$ is based on a Poisson point process with mean μ_B and spatial PDF $f_B(\mathbf{x}_{k,n})$ [6]. More precisely, we define $f_{B,j}(\mathbf{x}_{k,n})$ being the birth PDF of new PO n in data cell j as equal to, up to a normalization constant, $f_B(\mathbf{x}_{k,n})$ if $\mathbf{x}_{k,n}$ is in cell j and zero otherwise. In order to define the birth probability, we first note that the expected number of new objects in cell j is $\mu_{B,j} = \mu_B \int_{\mathcal{X}_j} f_B(\mathbf{x}_{k,n}) d\mathbf{x}_{k,n}$ with \mathcal{X}_j being the volume of cell j . By assuming that there is at most one new object in cell j , we obtain $p_{B,j} = \mu_{B,j}/(\mu_{B,j} + 1)$. Finally, our model for $f(\mathbf{y}_{k,n})$ reads

$$f(\mathbf{y}_{k,n}) = f(\mathbf{x}_{k,n}, r_{k,n}) = \begin{cases} (1 - p_{B,j}) f_D(\mathbf{x}_{k,n}), & r_{k,n} = 0 \\ p_{B,j} f_{B,j}(\mathbf{x}_{k,n}), & r_{k,n} = 1 \end{cases}$$

where $f_D(\mathbf{x}_{k,n})$ is an arbitrary “dummy” PDF.

C. Object Declaration and State Estimation

For each time step k , our ultimate goal is (i) to declare whether PO $n \in \{1, \dots, N_k\}$ exists and (ii) to estimate the state of existing POs. In our Bayesian setting, this necessitates the computation of the posterior distributions $f(r_{k,n} = 1 | \mathbf{z}_{1:k})$ and $f(\mathbf{x}_{k,n} | r_{k,n} = 1, \mathbf{z}_{1:k})$. In fact, PO n is declared to exist if its existence probability $f(r_{k,n} = 1 | \mathbf{z}_{1:k})$ is larger than a chosen threshold T_{dec} [29, Ch. 2]. For existing POs, we perform minimum mean-square error (MMSE) state estimation according to $\hat{\mathbf{x}}_{k,n} = \int \mathbf{x}_{k,n} f(\mathbf{x}_{k,n} | r_{k,n} = 1, \mathbf{z}_{1:k}) d\mathbf{x}_{k,n}$. Both $f(r_{k,n} = 1 | \mathbf{z}_{1:k})$ and $f(\mathbf{x}_{k,n} | r_{k,n} = 1, \mathbf{z}_{1:k})$ can be obtained from the marginal posterior state PDFs $f(\mathbf{y}_{k,n} | \mathbf{z}_{1:k})$. Thus, the remaining problem is to find an efficient method for calculating $f(\mathbf{y}_{k,n} | \mathbf{z}_{1:k})$.

III. BELIEF PROPAGATION FOR TBD

Based on the system model introduced in Section II, common assumptions [5], and Bayes' rule, the joint posterior PDF $f(\mathbf{y}_{0:k} | \mathbf{z}_{1:k})$ can be factorized as

$$f(\mathbf{y}_{0:k} | \mathbf{z}_{1:k}) \propto \left(\prod_{n=1}^{N_0} f(\mathbf{y}_{0,n}) \right) \prod_{k'=1}^k \left(\prod_{n=1}^{N_{k'-1}} f(\mathbf{y}_{k',n} | \mathbf{y}_{k'-1,n}) \right) \times \left(\prod_{n'=N_{k'-1}+1}^{N_{k'}} f(\mathbf{y}_{k',n'}) \right) \prod_{j=1}^J f(\mathbf{z}_{k',j} | \mathbf{y}_{k'}). \quad (2)$$

Given factorization (2), a factor graph [25], [28] representing the joint posterior PDF $f(\mathbf{y}_{0:k} | \mathbf{z}_{1:k})$ can be constructed. Fig. 1 shows a single time step of this graph.

BP [25], [26], [27], [28] performs local operations on factor graphs to compute representations of marginal posterior PDFs, called “beliefs”. Since the factor graph in Fig. 1 has loops, the beliefs $\tilde{f}(\mathbf{y}_{k,n})$ provided by BP are approximations of the true marginal posteriors $f(\mathbf{y}_{k,n} | \mathbf{z}_{1:k})$, and there are many possible message passing orders [25]. We only send BP messages forward in time and perform iterative message passing at each time step individually [5]. Next, we will present the specific BP messages passed on the graph in Fig. 1.

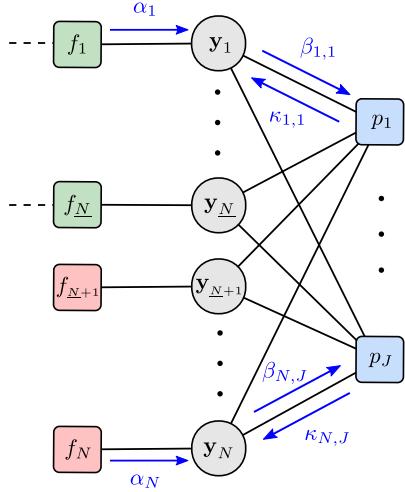


Fig. 1. Factor graph representing a single time step k of the joint posterior PDF in (2). We use the following short notation: $N = N_{k-1}$, $N = N_k$, $\mathbf{y}_n = \mathbf{y}_{k,n}$, $p_j = f(\mathbf{z}_{k,j}|\mathbf{y}_k)$, $f_n = f(\mathbf{y}_{k,n}|\mathbf{y}_{k-1,n})$ for $n \in \{1, \dots, N_{k-1}\}$, and $f_n = f(\mathbf{y}_{k,n})$ for $n \in \{N_{k-1}+1, \dots, N_k\}$. Furthermore, $\alpha_n = \alpha_{k,n}(\mathbf{y}_{k,n})$, $\beta_{n,j} = \beta_{k,n,j}^{(\ell)}(\mathbf{y}_{k,n})$, and $\kappa_{n,j} = \kappa_{k,n,j}^{(\ell)}(\mathbf{y}_{k,n}; \mathbf{z}_{k,j})$.

A. Prediction and Birth Messages

First, for each legacy PO $n \in \{1, \dots, N_{k-1}\}$, a prediction step is performed to compute the messages $\alpha_{k,n}(\mathbf{y}_{k,n}) = \alpha_n(\mathbf{x}_{k,n}, r_{k,n})$ that are passed from the factor nodes “ $f(\mathbf{y}_{k,n}|\mathbf{y}_{k-1,n})$ ” to the variable nodes “ $\mathbf{y}_{k,n}$ ”, i.e.,

$$\alpha_{k,n}(\mathbf{x}_{k,n}, 1) = \int p_s f(\mathbf{x}_{k,n}|\mathbf{x}_{k-1,n}) \tilde{f}(\mathbf{x}_{k-1,n}, 1) d\mathbf{x}_{k-1,n} \quad (3)$$

where p_s is the survival probability and $\alpha_{k,n}(\mathbf{x}_{k,n}, 0) = \alpha_{k,n} f_D(\mathbf{x}_{k,n})$. Here, $\alpha_{k,n}$ is the predicted probability of object non-existence [5]. For new POs $n \in \{N_{k-1}+1, \dots, N_k\}$, the messages from factor nodes “ $f(\mathbf{y}_{k,n})$ ” to variable nodes “ $\mathbf{y}_{k,n}$ ” are simply [25] $\alpha_{k,n}(\mathbf{y}_{k,n}) = f(\mathbf{y}_{k,n})$. In order to ease the notation, we omit the time index k in the following and, e.g., simply write \mathbf{y}_n instead of $\mathbf{y}_{k,n}$.

B. Iterative Message Passing and Belief Calculation

We now perform iterative message passing between the variable nodes “ \mathbf{y}_n ”, $n \in \{1, \dots, N\}$, and the factor nodes “ $f(\mathbf{z}_j|\mathbf{y})$ ”, $j \in \{1, \dots, J\}$. More precisely, at message passing iteration $\ell \in \{1, \dots, L\}$, the messages $\beta_{n,j}^{(\ell)}(\mathbf{y}_n)$ are passed from the variable nodes “ \mathbf{y}_n ” to the factor nodes “ $f(\mathbf{z}_j|\mathbf{y})$ ”. For $\ell > 1$, these messages can be computed according to

$$\beta_{n,j}^{(\ell)}(\mathbf{y}_n) = \frac{1}{C_{n,j}} \alpha_n(\mathbf{y}_n) \prod_{j'=1}^J \kappa_{n,j'}^{(\ell-1)}(\mathbf{y}_n; \mathbf{z}_{j'}), \quad (4)$$

and for $\ell = 1$, we set them to $\beta_{n,j}^{(1)}(\mathbf{y}_n) = \alpha_n(\mathbf{y}_n)$. The normalization factor $C_{n,j}$ ensures that $\sum_{\mathbf{y}_n} \beta_{n,j}^{(\ell)}(\mathbf{y}_n) = 1$. Note that $\sum_{\mathbf{y}_n}$ denotes integration for continuous random vector \mathbf{x}_n and summation for binary random variable r_n .

The messages $\kappa_{n,j}^{(\ell)}(\mathbf{y}_n; \mathbf{z}_j)$ passed from the factor nodes “ $f(\mathbf{z}_j|\mathbf{y})$ ” to the variable nodes “ \mathbf{y}_n ” are obtained for $\ell \in$

$\{1, \dots, L\}$ as

$$\kappa_{n,j}^{(\ell)}(\mathbf{y}_n; \mathbf{z}_j) = \sum_{\mathbf{y} \setminus \mathbf{y}_n} f(\mathbf{z}_j|\mathbf{y}) \prod_{n'=1}^N \beta_{n',j}^{(\ell)}(\mathbf{y}_{n'}). \quad (5)$$

Here, $\sum_{\mathbf{y} \setminus \mathbf{y}_n}$ denotes marginalization for all \mathbf{y} except \mathbf{y}_n . Note that our notation $\kappa_{n,j}^{(\ell)}(\mathbf{y}_n; \mathbf{z}_j)$ indicates that at this point the measurement \mathbf{z}_j is already observed and thus fixed.

After the last iteration $\ell = L$, the beliefs $\tilde{f}(\mathbf{y}_n)$ for all POs can be calculated as the normalized product of all incoming messages according to

$$\tilde{f}(\mathbf{y}_n) = \frac{1}{C_n} \alpha_n(\mathbf{y}_n) \prod_{j=1}^J \kappa_{n,j}^{(L)}(\mathbf{y}_n; \mathbf{z}_j). \quad (6)$$

Here, C_n again ensures $\sum_{\mathbf{y}_n} \tilde{f}(\mathbf{y}_n) = 1$. The obtained beliefs can then be used for object declaration and state estimation as discussed in Section II-C.

C. Approximate Computation of $\kappa_{n,j}^{(\ell)}(\mathbf{y}_n; \mathbf{z}_j)$ and Complexity

The computation of $\kappa_{n,j}^{(\ell)}(\mathbf{y}_n; \mathbf{z}_j)$ in (5), relies on a high-dimensional marginalization whose complexity scales exponentially with the number of POs N . To improve this complexity scaling, we approximate $\kappa_{n,j}^{(\ell)}(\mathbf{y}_n; \mathbf{z}_j)$ as follows. We interpret $\kappa_{n,j}^{(\ell)}(\mathbf{y}_n; \mathbf{z}_j)$ as the PDF of \mathbf{z}_j and approximate it by a Gaussian PDF via moment matching [23], i.e., $\kappa_{n,j}^{(\ell)}(\mathbf{y}_n; \mathbf{z}_j) \approx \tilde{\kappa}_{n,j}^{(\ell)}(\mathbf{y}_n; \mathbf{z}_j)$, where $\tilde{\kappa}_{n,j}^{(\ell)}(\mathbf{y}_n; \mathbf{z}_j) = \mathcal{N}(\mathbf{z}_j; \boldsymbol{\mu}_{\kappa,j}^{(\ell)}(\mathbf{y}_n), \mathbf{C}_{\kappa,j}^{(\ell)}(\mathbf{y}_n))$. Here, $\boldsymbol{\mu}_{\kappa,j}^{(\ell)}(\mathbf{y}_n)$ and $\mathbf{C}_{\kappa,j}^{(\ell)}(\mathbf{y}_n)$ are the matched mean and covariance matrix that are equal to the mean and covariance matrix of $\kappa_{n,j}^{(\ell)}(\mathbf{y}_n; \mathbf{z}_j)$. As derived in [30], $\boldsymbol{\mu}_{\kappa,j}^{(\ell)}(\mathbf{y}_n)$ and $\mathbf{C}_{\kappa,j}^{(\ell)}(\mathbf{y}_n)$ are given by

$$\boldsymbol{\mu}_{\kappa,j}^{(\ell)}(\mathbf{y}_n) = r_n \boldsymbol{\mu}_j(\mathbf{x}_n) + \sum_{\substack{n'=1 \\ n' \neq n}}^N \boldsymbol{\mu}_{n',j}^{(\ell)} \quad (7)$$

$$\mathbf{C}_{\kappa,j}^{(\ell)}(\mathbf{y}_n) = r_n \mathbf{C}_j(\mathbf{x}_n) + \mathbf{C}_\epsilon + \sum_{\substack{n'=1 \\ n' \neq n}}^N \left(\mathbf{R}_{n',j}^{(\ell)} - \boldsymbol{\mu}_{n',j}^{(\ell)} \boldsymbol{\mu}_{n',j}^{(\ell)\top} \right) \quad (8)$$

where we have introduced $\boldsymbol{\mu}_{n,j}^{(\ell)} \triangleq \int \boldsymbol{\mu}_j(\mathbf{x}_n) \beta_{n,j}^{(\ell)}(\mathbf{x}_n, 1) d\mathbf{x}_n$ and $\mathbf{R}_{n,j}^{(\ell)} \triangleq \int (\mathbf{C}_j(\mathbf{x}_n) + \boldsymbol{\mu}_j(\mathbf{x}_n) \boldsymbol{\mu}_j^\top(\mathbf{x}_n)) \beta_{n,j}^{(\ell)}(\mathbf{x}_n, 1) d\mathbf{x}_n$.

In summary, our BP-based TBD algorithm consists of executing the following steps for each time step: First, we perform state prediction according to (3) for $n \in \{1, \dots, N\}$. We then run the iterative message passing scheme by iteratively computing (4) and (7)–(8) for $\ell \in \{1, \dots, L\}$. Finally, after computing the PO beliefs according to (6), object declaration and state estimation are performed as described in Section II-C. An inspection of all these computations shows that the operations with the highest complexity are the summations in (7) and (8), whose complexity scales according to $\mathcal{O}(NJ)$. We can therefore conclude that the computational complexity of the proposed algorithm also exhibits this scaling behavior.

IV. NUMERICAL RESULTS

We consider a two-dimensional (2D) simulation scenario with a region of interest (ROI) of $[0 \text{ m}, 32 \text{ m}] \times [0 \text{ m}, 32 \text{ m}]$. We simulated five objects and 50 time steps. The object states are modeled by random vectors $\mathbf{x}_{k,n} = [\mathbf{p}_{k,n}^T \mathbf{v}_{k,n}^T \gamma_{k,n}]^T$ consisting of 2D position $\mathbf{p}_{k,n}$, 2D velocity $\mathbf{v}_{k,n}$, and the object's intensity $\gamma_{k,n}$. The objects appear at time steps $k \in \{1, 5, 10, 15, 20\}$ at positions randomly chosen in the region $[8 \text{ m}, 24 \text{ m}] \times [8 \text{ m}, 24 \text{ m}]$. The object's initial velocity is drawn from $\mathcal{N}(\mathbf{v}_{\cdot,n}; \mathbf{0}, 10^{-2} \mathbf{I}_2)$ and the object's initial intensity is γ_0 . The object's position $\mathbf{p}_{k,n}$ and velocity $\mathbf{v}_{k,n}$ evolve according to a constant velocity model [31, Ch. 4] with independent and identically distributed (IID) zero-mean Gaussian noise with variance 10^{-3} [31, Ch. 4]. The object's intensity $\gamma_{k,n}$ evolves according to a random walk model with IID zero-mean Gaussian noise with variance 10^{-4} . The objects disappear at $k \in \{31, 36, 41, 46, +\infty\}$ or when they leave the ROI.

The measurement \mathbf{z}_k is an image of 32×32 pixels or bins, i.e., $J = 1024$. Each bin has a square size with 1 m length, covering the total ROI. Pixel $j \in \{1, \dots, J\}$ is represented by the 2D vector $\mathbf{z}_{k,j}$ with center position \mathbf{p}_j^z . We use the measurement model defined by (1) and set the mean and the covariance of the Gaussian random vector $\mathbf{h}_{j,k,n}$ to $\mu_j(\mathbf{x}_{k,n}) = \mathbf{0}$ and

$$\mathbf{C}_j(\mathbf{x}_{k,n}) = \frac{\gamma_{k,n}}{2\pi\sigma_S^2} \exp\left(-\frac{\|\mathbf{p}_{k,n} - \mathbf{p}_j^z\|^2}{2\sigma_S^2}\right) \mathbf{I}_2, \quad (9)$$

respectively. Note that the variance σ_S^2 defines the shape of $f(\mathbf{h}_{j,k,n} | \mathbf{x}_{k,n})$ and thus the number of pixels illuminated by object n . Furthermore, the covariance of the noise vector $\epsilon_{k,j}$ in (1) is set to $\mathbf{C}_\epsilon = \sigma_\epsilon^2 \mathbf{I}_2$ with $\sigma_\epsilon^2 = 1$. These settings lead to a measurement process in which the contribution of object n on pixel j is large if the intensity $\gamma_{k,n}$ of object n is large and the position $\mathbf{p}_{k,n}$ of object n is close to pixel j .

We employ a particle implementation of our proposed BP-based TBD algorithm denoted by TBD-BP [4], [32]. The spatial PDF of each PO state's belief is represented by 300 particles. The generation of new POs is based on the measurement \mathbf{z}_k , in particular on $\|\mathbf{z}_{k,j}\|$. To keep the computational complexity low, we initialize a new PO $n \in \mathcal{N}_n \subseteq \{N_{k-1} + 1, \dots, N_k\}$ only for those pixels whose intensity $\|\mathbf{z}_{k,j}\|$ is larger than the predefined threshold $1.5\sqrt{\gamma_0/(2\pi\sigma_S^2)} + \sigma_\epsilon^2$. The spatial PDF of new PO $n \in \mathcal{N}_n$ is modeled by $f_{B,j}(\mathbf{x}_{k,n}) = f_j(\mathbf{p}_{k,n})f(\mathbf{v}_{k,n})f(\gamma_{k,n})$. Here, $f_j(\mathbf{p}_{k,n})$ is uniform over the area of pixel j , $f(\mathbf{v}_{k,n})$ is $\mathcal{N}(\mathbf{v}_{k,n}; \mathbf{0}, 10^{-2} \mathbf{I}_2)$, and $f(\gamma_{k,n})$ is uniform from 0 to $\gamma_{\max} = 2\gamma_0$. The birth probability of each new PO is set to $p_{B,j} = 10^{-5}$. An object is declared to exist if its existence probability is larger than $T_{\text{dec}} = 0.5$. We prune POs whose existence probability is below $T_{\text{pru}} = 10^{-3}$ [5]. The survival probability is set to $p_s = 0.999$.

To evaluate the performance of our proposed algorithm, we compute the Euclidean distance based generalized optimal sub-pattern assignment (GOSPA) metric [33] averaged over 400 simulation runs, with cutoff parameter $c = 1$, order $p = 2$, and $\alpha = 2$. In our first experiment, we compare the proposed TBD-BP with a particle implementation of the multi-Bernoulli (MB) filter in [20], referred to as TBD-MB, and the information exchange multi-Bernoulli (IEMB) filter in [23]. The TBD-IEMB models spatial distributions by Gaussian PDFs. The GOSPA results are displayed in Fig. 2. As the figure shows, TBD-BP with $L = 2$ message passing iterations performs slightly better than

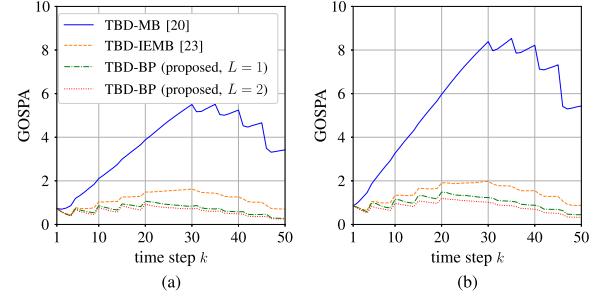


Fig. 2. GOSPA error of TBD-MB, TBD-IEMB and the proposed TBD-BP versus time k for $\gamma_0 = 60$, (a) $\sigma_S^2 = 0.5$ and (b) $\sigma_S^2 = 1$.

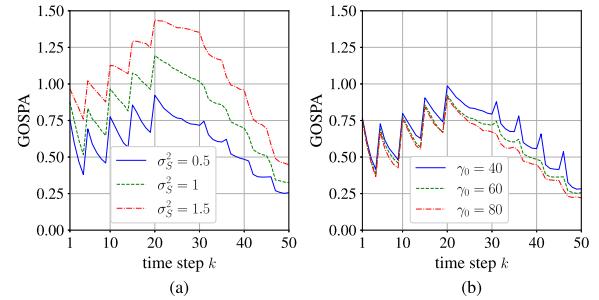


Fig. 3. GOSPA error of the proposed TBD-BP ($L = 2$) with (a) $\gamma_0 = 60$ and different σ_S^2 ; (b) $\sigma_S^2 = 0.5$ and different γ_0 .

TBD-BP with $L = 1$, followed by TBD-IEMB, and significantly better than TBD-MB. The lower performance of TBD-MB is due to the fact that it does not model the interaction of objects and tracks them independently. Furthermore, the track initialization scheme of TBD-MB leads to a high number of false tracks resulting in an almost linear increase in GOSPA for $k < 30$.

In the second experiment, we investigated the GOSPA performance of TBD-BP for different intensity values γ_0 and variances σ_S^2 . As Fig. 3(a) shows, the GOSPA error increases for larger σ_S^2 . This is due to the fact that for larger σ_S^2 , an object illuminates a larger number of neighboring pixels, which in turn leads to more false tracks caused by the now larger number of pixels of higher intensity. Fig. 3(b) additionally shows that the GOSPA error decreases as γ_0 increases.

V. CONCLUSION

In this letter, we propose a BP method for tracking an unknown number of low-observable objects based on the TBD approach. We introduced a new object birth model and a new measurement model that allows interacting objects to contribute to more than one data cell. To reduce computational complexity and improve scalability, certain BP messages are approximated by Gaussian distributions. Experiments conducted on image data show that the proposed TBD-BP method outperforms two state-of-the-art TBD-MB filtering methods [20], [23]. However, the proposed method is limited to measurements that are independent conditioned on the object states. An interesting possibility for future research is the design of a more general measurement model and an application to real data. Other promising directions for future research are the development of TBD approaches for extended object tracking [9] or based on particle flow [34], [35], [36], [37] and an extension to hybrid model-based and data-driven TBD [38].

REFERENCES

- [1] Y. Bar-Shalom, P. K. Willett, and X. Tian, *Tracking and Data Fusion: A Handbook of Algorithms*. Storrs, CT, USA: Yaakov Bar-Shalom, 2011.
- [2] R. Mahler, *Statistical Multisource-Multitarget Information Fusion*. Norwood, MA, USA: Artech House, 2007.
- [3] S. Challa, M. R. Morelande, D. Mušicki, and R. J. Evans, *Fundamentals of Object Tracking*. Cambridge, U.K.: Cambridge Univ. Press, 2011.
- [4] F. Meyer, P. Braca, P. Willett, and F. Hlawatsch, “A scalable algorithm for tracking an unknown number of targets using multiple sensors,” *IEEE Trans. Signal Process.*, vol. 65, no. 13, pp. 3478–3493, Jul. 2017.
- [5] F. Meyer et al., “Message passing algorithms for scalable multitarget tracking,” *Proc. IEEE*, vol. 106, no. 2, pp. 221–259, Feb. 2018.
- [6] 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.
- [7] G. Ferri et al., “Cooperative robotic networks for underwater surveillance: An overview,” *IET Radar Sonar Navigation*, vol. 11, no. 12, pp. 1740–1761, Dec. 2017.
- [8] K. Witrisal et al., “High-accuracy localization for assisted living: 5G systems will turn multipath channels from foe to friend,” *IEEE Signal Process. Mag.*, vol. 33, no. 2, pp. 59–70, Mar. 2016.
- [9] F. Meyer and J. L. Williams, “Scalable detection and tracking of geometric extended objects,” *IEEE Trans. Signal Process.*, vol. 69, pp. 6283–6298, 2021.
- [10] S. Tonissen and Y. Bar-Shalom, “Maximum likelihood track-before-detect with fluctuating target amplitude,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 34, no. 3, pp. 796–809, Jul. 1998.
- [11] L. R. Moyer, J. Spak, and P. Lamanna, “A multi-dimensional Hough transform-based track-before-detect technique for detecting weak targets in strong clutter backgrounds,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 47, no. 4, pp. 3062–3068, Oct. 2011.
- [12] Y. Barniv, “Dynamic programming solution for detecting dim moving targets,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 21, no. 1, pp. 144–156, Jan. 1995.
- [13] D. Salmond and H. Birch, “A particle filter for track-before-detect,” in *Proc. IEEE Amer. Control Conf.*, 2001, pp. 3755–3760.
- [14] M. Orton and W. Fitzgerald, “A Bayesian approach to tracking multiple targets using sensor arrays and particle filters,” *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 216–223, Feb. 2002.
- [15] Y. Boers, J. N. Driessens, F. Verschure, W. P. M. H. Heemels, and A. Juloski, “A multi target track before detect application,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshop*, 2003, Art. no. 104.
- [16] N. Ito and S. Godsill, “A multi-target track-before-detect particle filter using superpositional data in non-Gaussian noise,” *IEEE Trans. Signal Process.*, vol. 27, pp. 1075–1079, 2020.
- [17] S. J. Davey and H. X. Gaetjens, *Track-Before-Detect Using Expectation Maximisation*. Singapore: Springer, 2018.
- [18] 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.
- [19] B. Ristic, L. Rosenberg, D. Y. Kim, X. Wang, and J. Williams, “Bernoulli track-before-detect filter for maritime radar,” *IET Radar, Sonar Navigation*, vol. 14, no. 3, pp. 356–363, Mar. 2020.
- [20] B.-N. Vo, B.-T. Vo, N.-T. Pham, and D. Suter, “Joint detection and estimation of multiple objects from image observations,” *IEEE Trans. Signal Process.*, vol. 58, no. 10, pp. 5129–5141, Oct. 2010.
- [21] T. Kropfleiter, J. L. Williams, and F. Meyer, “A scalable track-before-detect method with poisson/multi-Bernoulli model,” in *Proc. IEEE 24th Int. Conf. Inf. Fusion*, 2021, pp. 1–8.
- [22] D. Y. Kim, B. Ristic, R. Guan, and L. Rosenberg, “A Bernoulli track-before-detect filter for interacting targets in maritime radar,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 57, no. 3, pp. 1981–1991, Jun. 2021.
- [23] E. S. Davies and Á. F. García-Fernández, “A multi-Bernoulli Gaussian filter for track-before-detect with superpositional sensors,” in *Proc. IEEE 25th Int. Conf. Inf. Fusion*, 2022, pp. 1–8.
- [24] C. Cao and Y. Zhao, “A generalized labeled multi-Bernoulli filter based on track-before-detect measurement model for multiple-weak-target state estimate using belief propagation,” *Remote Sens.*, vol. 14, no. 17, pp. 1–18, Aug. 2022.
- [25] 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.
- [26] J. Yedidia, W. Freeman, and Y. Weiss, “Constructing free-energy approximations and generalized belief propagation algorithms,” *IEEE Trans. Inf. Theory*, vol. 51, no. 7, pp. 2282–2312, Jul. 2005.
- [27] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques*. Cambridge, MA, USA: MIT Press, 2009.
- [28] H.-A. Loeliger, “An introduction to factor graphs,” *IEEE Signal Process. Mag.*, vol. 21, no. 1, pp. 28–41, Jan. 2004.
- [29] H. V. Poor, *An Introduction to Signal Detection and Estimation*, 2nd ed. New York, NY, USA: Springer, 1994.
- [30] M. Liang, T. Kropfleiter, and F. Meyer, “A BP method for track-before-detect: Supporting derivations,” 2023. [Online]. Available: <https://fmeyer.ucsd.edu/SPL-2023-SD.pdf>
- [31] Y. Bar-Shalom, T. Kirubarajan, and X.-R. Li, *Estimation With Applications to Tracking and Navigation*. New York, NY, USA: Wiley, 2002.
- [32] A. Ihler and D. McAllester, “Particle belief propagation,” in *Proc. Conf. Artif. Intell. Statist.*, 2009, vol. 5, pp. 256–263.
- [33] A. S. Rahmathullah, Á. F. García-Fernández, and L. Svensson, “Generalized optimal sub-pattern assignment metric,” in *Proc. IEEE 20th Int. Conf. Inf. Fusion*, 2017, pp. 1–8.
- [34] F. Daum and J. Huang, “Nonlinear filters with log-homotopy,” in *Proc. SPIE Conf. Signal Data Process. Small Targets*, 2007, pp. 423–437.
- [35] J. Jang, F. Meyer, E. R. Snyder, S. M. Wiggins, S. Baumann-Pickering, and J. A. Hildebrand, “Bayesian detection and tracking of odontocetes in 3-D from their echolocation clicks,” *J. Acoust. Soc. Amer.*, vol. 153, no. 5, pp. 2690–2705, May 2023.
- [36] L. Dai and F. Daum, “On the design of stochastic particle flow filters,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 59, no. 3, pp. 2439–2450, Jun. 2023.
- [37] W. Zhang and F. Meyer, “Multisensor multiobject tracking with high-dimensional object states,” 2023. [Online]. Available: <https://arxiv.org/abs/2212.14556>
- [38] M. Liang and F. Meyer, “Neural enhanced belief propagation for multiobject tracking,” 2023. [Online]. Available: <https://arxiv.org/abs/2212.08340>