





# Multi-Scan Multi-Sensor Multi-Object State Estimation

Diluka Moratuwage , Ba-Ngu Vo , *Fellow, IEEE*, Ba-Tuong Vo , *Member, IEEE*, and Changbeom Shim 

**Abstract**—If computational tractability were not an issue, multi-object estimation should integrate all measurements from multiple sensors across multiple scans. In this article, we propose an efficient numerical solution to the multi-scan multi-sensor multi-object estimation problem by computing the (labeled) multi-sensor multi-object posterior density. Minimizing the  $L_1$ -norm error from the exact posterior density requires solving large-scale multi-dimensional assignment problems that are NP-hard. An efficient multi-dimensional assignment algorithm is developed based on Gibbs sampling, together with convergence analysis. The resulting multi-scan multi-sensor multi-object estimation algorithm can be applied either offline in one batch or recursively. The efficacy of the algorithm is demonstrated using numerical experiments with a simulated dataset.

**Index Terms**—State estimation, Smoothing, Random finite sets, Multi-sensor, Multi-dimensional assignment, Gibbs sampling.

## I. INTRODUCTION

INSTEAD of the state or trajectory of a single object, multi-object state estimation is concerned with the joint estimation of the number of objects and their trajectories [1], [2], [3]. This problem has a wide range of application areas from aerospace [4], computer vision [5], [6], [7], to cell biology [8], [9]. Multi-object state estimation is challenging because we have to address the unknown and time-varying number of objects, false negatives/positives, and data association uncertainty, which altogether incur a combinatorial complexity. The three main approaches to multi-object estimation in the literature are Multiple Hypothesis Tracking (MHT) [2], Joint Probabilistic Data Association (JPDA) [10], and Random Finite Set (RFS) [1], [3]. Apart from the number of objects and their trajectories, the RFS approach also provides statistical characterization of the entire ensemble of objects [11].

In general, employing multiple sensors in state estimation enhances detection capability and spatial coverage, reliability, observability, and reduces uncertainty [12], [13]. There are

many applications that fundamentally require multiple sensors because a single sensor is simply not adequate, see e.g., [14], [15]. Moreover, given the proliferation of inexpensive sensors, effective multiple sensor solutions are imperative for exploiting their intended capabilities. The benefits of multiple sensors are even more significant in multi-object estimation, where there is inherent additional uncertainty.

In multi-object estimation, the benefits of multiple sensors come with additional challenges due to the NP-hard multi-dimensional data association problem that arises from the matching of objects to measurements across multiple sensors [3], [13], [16], [17]. Several approximate multi-sensor multi-object estimation solutions have been developed. Centralized architecture solutions include the multi-sensor Probability Hypothesis Density (PHD) and Cardinalized PHD (CPHD) filters [3], [18] multi-sensor JPDA filter [19], multi-Sensor multi-Bernoulli filter [20], and multi-sensor Generalized Labeled Multi-Bernoulli (GLMB) filter [21]. The latter incurs a linear complexity in the total number of measurements across the sensors, without significantly compromising optimality. Decentralized solutions have also been developed for the PHD, CPHD filters [22], [23], [24] multi-Bernoulli filter [25], [26], [27], LMB and marginalized GLMB filters [28].

Whereas filtering only considers the current timestep, the multi-scan approach considers the history of the multi-object states over a time window, thereby allowing the correction of previous errors [2], [29], [30]. MHT forms hypotheses by associating measurements to tracks across a time window [31], and delays making estimates until further information becomes available. In [32], a fixed lag smoother was integrated to the JPDA filter [10], where it was shown that a significant improvement in tracking accuracy over filtering can be achieved with small window of sizes two and three. The first RFS multi-scan multi-object filter was developed in [30], where all statistical information pertinent to the multi-object system over the time window is encapsulated in the multi-object posterior, including statistics on the ensemble of trajectories. Computing this posterior, however, also requires solving NP-hard multi-dimensional ranked assignment problems. Recently, an efficient numerical algorithm with polynomial complexity was developed for propagating the so-called (multi-scan) GLMB posterior, where multi-dimensional ranked assignment problems are solved using Gibbs sampling [11].

Ideally, if computational tractability were not an issue, to inherit all the benefits from both multi-sensor and multi-scan solutions we should make use of measurements from all sensors

Manuscript received 29 May 2022; revised 11 October 2022; accepted 20 October 2022. Date of publication 31 October 2022; date of current version 21 November 2022. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Yue M. Lu. This work was supported in part by Australian Research Council under Linkage under Project LP200301507 and in part by Future Fellowship under Grant FT210100506. (*Corresponding author: Diluka Moratuwage.*)

The authors are with the Faculty of Science and Engineering, Curtin University, Bentley, WA 6102, Australia (e-mail: diluka.moratuwage@curtin.edu.au; ba-ngu.vo@curtin.edu.au; ba-tuong.vo@curtin.edu.au; changbeom.shim@curtin.edu.au).

This article has supplementary downloadable material available at <https://doi.org/10.1109/TSP.2022.3218366>, provided by the authors.

Digital Object Identifier 10.1109/TSP.2022.3218366

across multiple scans. However, keeping in mind that the multi-scan problem itself requires solving NP-hard multi-dimensional ranked assignment problems, and ditto for the multi-sensor problem, the joint multi-scan multi-sensor problem is far more computationally demanding. Consequently, research on multi-scan multi-sensor multi-object estimation is very limited. To the best of our knowledge, only a two-scan two-sensor demonstration with four objects, using an Interacting Multiple Model (IMM) JPDA filter, has been reported [33].

This article addresses multi-scan multi-sensor multi-object estimation via the labeled RFS formulation, which translates to computing a labeled multi-object posterior consisting of an intractably large weighted sum of set functions. Approximating this so-called GLMB posterior by retaining the highest weighted terms minimizes the  $L_1$ -norm approximation error [11], and requires solving far more challenging multi-dimensional assignment problems than those for multi-scan (with single-sensor) or multi-sensor (with single-scan). Indeed, the posterior truncation problem for  $V$  sensors across  $K$  scans is a  $(KV)$ -dimensional assignment problem, and existing techniques are not adequate for  $K$  and  $V$  greater than two, even with only four objects.

We propose an efficient multi-dimensional assignment solution with polynomial complexity using Gibbs sampling, together with convergence analysis. This solution generalizes those for multi-sensor (with single-scan) and multi-scan (with single-sensor) in [11], [21]. The resultant multi-object estimation algorithm can be used offline in one batch or recursively at each measurement update, i.e., smoothing-while-filtering. The efficacy and utility of the proposed solution are demonstrated in numerical experiments. For the purposes of establishing the theoretical foundation and scalability for the algorithm, we focus on the (labeled) multi-object posterior, which involves a growing window of scans. While such a growing window results in a complexity per time step that grows with time, the algorithm can be easily adapted to a fixed-length moving window, so that the complexity per time step is fixed. Note that filtering is a special case of the moving window approach with a window length of one. A formulation that produces trajectories with a fixed complexity per time step is only possible with labels, see e.g., [34], [35] for further discussion. Without labels, the only way to obtain trajectories is by using a growing window which incur a complexity per time step that grows exponentially with time. Such an approach is impractical even for the simplest case of single object state estimation with a single sensor [36].

The remainder of this article is organized as follows. Section II summarizes relevant concepts in multi-object estimation, and Section III presents the implementation details of the multi-sensor GLMB posterior recursion. Section IV presents the numerical studies, and Section V concludes the paper. Mathematical proofs are given in the supplementary materials.

## II. BACKGROUND

As per the convention in [11], the symbols and notations used throughout the article are summarized in Table I.

TABLE I  
NOTATIONS

Notation	Description
$X_{m:n}$	$X_m, X_{m+1}, \dots, X_n$
$1_S(\cdot)$	Indicator function for a given set $S$
$\delta_X[Y]$	Kronecker- $\delta$ , $\delta_X[Y] = 1$ if $X = Y$ , 0 otherwise
$\mathcal{F}(S)$	Class of all finite subsets of a given set $S$
$\langle f, g \rangle$	Inner product, $\int f(x)g(x)dx$ of two functions $f$ and $g$
$ X $	Cardinality of a set $X$
$h^X$	Multi-object exponential, $\prod_{x \in X} h(x)$ , with $h^\emptyset = 1$
$1^V$	$V$ -dimensional vector $(1, \dots, 1)$
$\mathcal{L}(x)$	Label of a (labeled) state $x$ , $\mathcal{L}((x, \ell)) = \ell$
$\mathcal{L}(X)$	Labels of a (labeled) set $X$ , $\{\mathcal{L}(x) : x \in X\}$
$\Delta(X)$	Distinct label indicator $\delta_{ X }( \mathcal{L}(X) )$

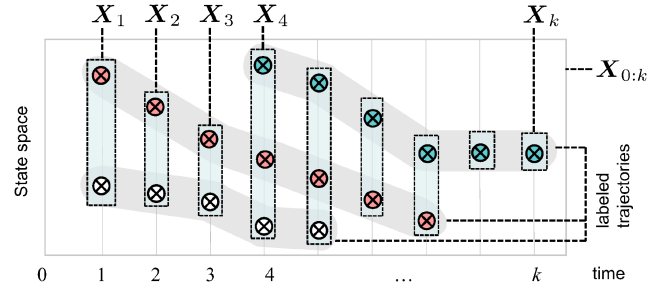


Fig. 1. Labeled multi-object trajectory and history. The two objects born at time 1 are given labels (1,1) and (1,2), colored red and white. The object born at time 4 is given label (4,1), colored blue. The multi-object history  $X_{0:k}$  (note  $X_0 = \emptyset$ ) is represented by two equivalent groupings according to: (a) time (vertical strips, i.e., multi-object states) or; (b) labels (strips containing states of the same color, i.e., trajectories).

### A. Multi-Object State and Multi-Object Trajectory

The labeled state of an object is represented by  $x = (x, \ell)$ , where the vector  $x \in \mathbb{X}$  is its kinematic state, and  $\ell = (s, \iota)$  is its (unique) label with  $s$  representing the time of birth and  $\iota$  is an index to distinguish objects born at the same time [37]. Let  $\mathbb{B}_s$  denote the label space of objects born at time  $s$ , then the space of all labels up to time  $k$  is the disjoint union  $\mathbb{L}_k = \uplus_{s=0}^k \mathbb{B}_s$  (equivalently  $\mathbb{L}_k = \mathbb{L}_{k-1} \uplus \mathbb{B}_k$ ). A sequence of consecutive labeled states (say, from time  $s$  to  $t$ )

$$\tau = [(x_s, \ell), (x_{s+1}, \ell), \dots, (x_t, \ell)], \quad (1)$$

with kinematic states  $x_s, x_{s+1}, \dots, x_t \in \mathbb{X}$ , and label  $\ell = (s, \iota)$ , is called a *trajectory*. The labeled state of trajectory  $\tau$  at time  $i$  is denoted by  $\tau(i)$ .

At time  $i$ , a *labeled multi-object state*  $X_i$  is a finite subset of  $\mathbb{X} \times \mathbb{L}_i$  with distinct labels. Let  $\mathcal{L}$  denote the projection defined by  $\mathcal{L}((x, \ell)) = \ell$ , and  $\mathcal{L}(X_i)$  denote the set of labels of  $X_i$ . Since a valid labeled multi-object state  $X_i$  must have distinct labels, we require the *distinct label indicator*  $\Delta(X_i) \triangleq \delta_{|X_i|}(|\mathcal{L}(X_i)|)$  to be 1. Given a set  $S$  of trajectories with distinct labels, the labeled multi-object state at time  $i$  is  $X_i = \{\tau(i) : \tau \in S\}$ , see Fig. 1 for illustration.

Given a sequence  $X_{j:k}$  of labeled multi-object states (from a set of labeled trajectories) over the interval  $\{j : k\}$ , the trajectory of (the object with) label  $\ell \in \cup_{i=j}^k \mathcal{L}(X_i)$  is

$$x_{s(\ell):t(\ell)}^{(\ell)} \triangleq [(x_{s(\ell)}^{(\ell)}, \ell), \dots, (x_{t(\ell)}^{(\ell)}, \ell)], \quad (2)$$

where:  $s(\ell)$  and  $t(\ell)$  are, respectively, the earliest and latest times on  $\{j : k\}$  such that the label  $\ell$  exists; and  $(x_i^{(\ell)}, \ell)$  denotes the element of  $\mathbf{X}_i$  with label  $\ell$  and unlabeled state  $x_i^{(\ell)}$ . Hence, the sequence  $\mathbf{X}_{j:k}$  can be equivalently represented by the trajectories of all labels in  $\cup_{i=j}^k \mathcal{L}(\mathbf{X}_i)$ , i.e.,

$$\mathbf{X}_{j:k} \equiv \left\{ \mathbf{x}_{s(\ell):t(\ell)}^{(\ell)} : \ell \in \cup_{i=j}^k \mathcal{L}(\mathbf{X}_i) \right\}. \quad (3)$$

This equivalence is illustrated in Fig. 1.

Analogous to a single-object system where the state trajectory is represented by the state history  $x_{0:k}$ , in a multi-object system the *multi-object trajectory* is represented by the labeled multi-object (state) history  $\mathbf{X}_{0:k}$ . Further, the equivalence (3) enables the following extension of the multi-object exponential to multiple scans. For any non-negative integer  $n$  and  $i_1 < i_2 < \dots < i_n$ , let  $\mathbb{T}_{\{i_1, i_2, \dots, i_n\}} \triangleq (\mathbb{X} \times \mathbb{I}_{i_1}) \times \dots \times (\mathbb{X} \times \mathbb{I}_{i_n})$ , with  $\mathbb{T}_\emptyset = \emptyset$ . For any function  $h : \cup_{I \subseteq \{j:k\}} \mathbb{T}_I \rightarrow [0, \infty)$ , the *multi-scan exponential* [11] is defined as

$$[h]^{\mathbf{X}_{j:k}} \triangleq [h]^{\{\mathbf{x}_{s(\ell):t(\ell)}^{(\ell)} : \ell \in \mathcal{L}(\mathbf{X}_{j:k})\}} = \prod_{\ell \in \mathcal{L}(\mathbf{X}_{j:k})} h\left(\mathbf{x}_{s(\ell):t(\ell)}^{(\ell)}\right).$$

Note that when  $j = k$ , this reduces to the single-scan multi-object exponential  $h^{\mathbf{X}_j}$  in Table 1.

Hereon, single object states are represented by lower case letters (i.e.,  $x$  and  $\mathbf{x}$ ), and multi-object states are represented by upper case letters (i.e.,  $X$  and  $\mathbf{X}$ ), where the symbols for labeled states and their distributions are bolded (i.e.,  $\mathbf{x}$ ,  $\mathbf{X}$ ,  $\boldsymbol{\pi}$ , etc.) to distinguish them from unlabeled states.

In Bayesian estimation, the states and measurements are modeled as random variables. Hence, in a multi-object system, the multi-object state is modeled as an RFS, and the system model is described by the multi-object state transition kernel and measurement likelihood function, presented respectively in Subsection II-B and II-C. The multi-object posterior recursion is presented in Subsection II-D.

## B. Multi-Object Dynamic Model

Given the multi-object state  $\mathbf{X}_{k-1}$  at time  $k-1$ , each  $\mathbf{x}_{k-1} = (x_{k-1}, \ell_{k-1}) \in \mathbf{X}_{k-1}$  either survives with probability  $P_{S,k-1}(\mathbf{x}_{k-1})$  and moves to a new state  $\mathbf{x}_k = (x_k, \ell_k)$  with transition density  $f_{S,k|k-1}(x_k|x_{k-1}, \ell_k)\delta_{\ell_{k-1}}[\ell_k]$  at time  $k$ , or dies with probability  $Q_{S,k-1}(\mathbf{x}_{k-1}) = 1 - P_{S,k-1}(\mathbf{x}_{k-1})$ . An object (with label)  $\ell_k \in \mathbb{B}_k$ , is either born at time  $k$  with probability  $P_{B,k}(\ell_k)$  and state  $x_k$  with probability density  $f_{B,k}(x_k, \ell_k)$ , or not born with probability  $Q_{B,k}(\ell_k) = 1 - P_{B,k}(\ell_k)$ . The multi-object state  $\mathbf{X}_k$  at time  $k$ , is the superposition of surviving and new birth states. In a standard multi-object dynamic model, the birth and survival sets are independent of each other, and each object moves and dies independently of each other. The multi-object dynamic model is encapsulated in the multi-object transition density  $\mathbf{f}_{k|k-1}(\mathbf{X}_k|\mathbf{X}_{k-1})$ , see [11], [37] for the actual expressions.

## C. Multi-Object Measurement Model

Consider the multi-object state  $\mathbf{X}_k$  at time  $k$ , and  $V$  sensors, each produces a set  $Z_k^{(v)}$  of measurements,  $v \in \{1 : V\}$ . Each

$\mathbf{x} \in \mathbf{X}_k$  is either detected by sensor  $v$  with probability  $P_{D,k}^{(v)}(\mathbf{x})$  and generates a detection  $z \in Z_k^{(v)}$  on the observation space  $\mathbb{Z}^{(v)}$ , with likelihood  $g_k^{(v)}(z|\mathbf{x})$  or miss-detected with probability  $Q_{D,k}^{(v)}(\mathbf{x}) = 1 - P_{D,k}^{(v)}(\mathbf{x})$ . Sensor  $v$  also receives false alarms (clutter), modeled by a Poisson RFS with intensity function  $\kappa_k^{(v)}$  on  $\mathbb{Z}^{(v)}$ . The multi-object observation  $Z_k^{(v)}$  is the superposition of detections and false alarms, which are assumed independent, conditional on  $\mathbf{X}_k$ .

An *association map*  $\gamma_k^{(v)} : \mathbb{L}_k \rightarrow \{-1 : |Z_k^{(v)}|\}$ , of sensor  $v$ , is a positive 1-1 map (i.e., no two distinct labels are mapped to the same positive value). Here  $\gamma_k^{(v)}(\ell) > 0$  means  $\ell$  generates the  $\gamma_k^{(v)}(\ell)$ -th measurement at sensor  $v$ ,  $\gamma_k^{(v)}(\ell) = 0$  means  $\ell$  is misdetected by sensor  $v$ , and  $\gamma_k^{(v)}(\ell) = -1$  means  $\ell$  does not exist. Let  $\mathcal{L}(\gamma_k^{(v)}) \triangleq \{\ell \in \mathbb{L}_k : \gamma_k^{(v)}(\ell) \geq 0\}$  be the set of *live labels*<sup>1</sup> of  $\gamma_k^{(v)}$ , and  $\Gamma_k^{(v)}$  be the space of all association maps for sensor  $v$ , then the multi-object likelihood is [21]

$$g_k^{(v)}(Z_k^{(v)}|\mathbf{X}_k) \propto \sum_{\gamma_k^{(v)} \in \Gamma_k^{(v)}} \delta_{\mathcal{L}(\gamma_k^{(v)})}[\mathcal{L}(\mathbf{X}_k)] \left[ \psi_{k, Z_k^{(v)}}^{(v, \gamma_k^{(v)} \circ \mathcal{L}(\cdot))}(\cdot) \right]^{\mathbf{X}_k} \quad (4)$$

where  $\gamma_k^{(v)} \circ \mathcal{L}(\cdot) = \gamma_k^{(v)}(\mathcal{L}(\cdot))$ , and

$$\psi_{k, \{z_{1:m}\}}^{(v, i)}(\mathbf{x}) = \begin{cases} \frac{P_{D,k}^{(v)}(\mathbf{x})g_k^{(v)}(z_i|\mathbf{x})}{\kappa_k^{(v)}(z_i)} & i > 0 \\ Q_{D,k}^{(v)}(\mathbf{x}) & i = 0 \end{cases}. \quad (5)$$

The *multi-sensor association map* is defined from the single-sensor association maps  $\gamma_k^{(v)}$ ,  $v \in \{1 : V\}$  by

$$\gamma_k = (\gamma_k^{(1)}, \dots, \gamma_k^{(V)}) : \mathbb{L}_k \rightarrow \{-1^V\} \cup \Lambda_k^{(1:V)},$$

where  $1^V$  is the  $V$ -tuple of ones,  $\Lambda_k^{(1:V)} \triangleq \Lambda_k^{(1)} \times \dots \times \Lambda_k^{(V)}$ , and  $\Lambda_k^{(v)} \triangleq \{0, \dots, |Z_k^{(v)}|\}$ . Note that each  $\gamma_k^{(v)}$  is required to be positive 1-1, in which case  $\gamma_k$  is said to be positive 1-1. Here,  $\gamma_k(\ell) = -1^V$  means label  $\ell$  does not exist, whereas  $\gamma_k(\ell) \in \Lambda_k^{(1:V)}$  means label  $\ell$  exists, in which case it is misdetected by sensor  $v$  if  $\gamma_k^{(v)}(\ell) = 0$ , or generates the  $\gamma_k^{(v)}(\ell)$ -th measurement at sensor  $v$  if  $\gamma_k^{(v)}(\ell) > 0$ . Since the live label set  $\mathcal{L}(\gamma_k^{(v)})$  can be written as  $\{\ell \in \mathbb{L}_k : \gamma_k(\ell) \in \Lambda_k^{(1:V)}\}$ , which is independent of  $v$ , we write it as  $\mathcal{L}(\gamma_k)$ .

Assuming that the sensors are independent conditional on  $\mathbf{X}_k$ , the multi-sensor multi-object likelihood function is simply the product of the likelihoods of individual sensors. Let  $Z_k \triangleq (Z_k^{(1)}, \dots, Z_k^{(V)})$  denote the multi-sensor observation, then the multi-sensor multi-object likelihood function is

$$g_k(Z_k|\mathbf{X}_k) \propto \sum_{\gamma_k \in \Gamma_k} \delta_{\mathcal{L}(\gamma_k)}[\mathcal{L}(\mathbf{X}_k)] \left[ \psi_{k, Z_k}^{(\gamma_k \circ \mathcal{L}(\cdot))}(\cdot) \right]^{\mathbf{X}_k}, \quad (6)$$

<sup>1</sup>This notation is similar to  $\mathcal{L}(\mathbf{X})$ , the labels of a labeled set, nonetheless the context is clear from the arguments.

where  $\Gamma_k$  is the space of multi-sensor association maps, and

$$\psi_{k,Z_k}^{(\alpha^{(1:V)})}(\mathbf{x}) \triangleq \prod_{v=1}^V \psi_{k,Z_k^{(v)}}^{(v,\alpha^{(v)})}(\mathbf{x}).$$

#### D. Multi-Object Bayes Recursion

All information about the set of objects in the surveillance region for the interval  $\{0 : k\}$  is captured by the multi-object posterior, which can be recursively propagated in time by,

$$\pi_{0:k}(\mathbf{X}_{0:k}) \propto g_k(Z_k|\mathbf{X}_k) \mathbf{f}_{k|k-1}(\mathbf{X}_k|\mathbf{X}_{k-1}) \pi_{0:k-1}(\mathbf{X}_{0:k-1}).$$

Note that the posterior is conditioned on the measurement history  $Z_{1:k}$ , but we have omitted it for brevity. The dimensionality of the posterior (hence complexity per timestep) grows with time, and computing it at each timestep is impractical. Nevertheless, the complexity per timestep can be capped by smoothing over short windows and linking trajectory estimates between windows via their labels. Multi-object filtering can be regarded as a special case with a window length of one.

For the single-sensor special case, Particle Markov Chain Monte Carlo [38] has been applied to approximate the multi-object posterior in [30]. Further, an analytic solution called the (multi-scan) GLMB filter/smoothen was developed in [11], which conceptually extends to the multi-sensor case.

#### E. Multi-Sensor GLMB Posterior

Before delving into the multi-sensor GLMB posterior, it is informative to consider the association weight and trajectory posterior of a label  $\ell$ . Recall that  $s(\ell)$  and  $t(\ell)$  are, respectively, the earliest and latest times on  $\{0 : k\}$  such that  $\ell$  exists. There are four possible cases for its trajectory: (i) new born,  $s(\ell) = k$ ; (ii) surviving,  $t(\ell) = k > s(\ell)$ ; (iii) die at time  $k$ ,  $t(\ell) = k - 1$ ; (iv) died before time  $k$ ,  $t(\ell) < k - 1$ . Suppose that  $\ell$  generated the sequence  $\alpha_{s(\ell):k}$  of multi-sensor measurement indices, then its trajectory posterior at time  $k$  is

$$\tau_{0:k}^{(\alpha_{s(\ell):k})}(x_{s(\ell):t(\ell)}, \ell) = \begin{cases} \frac{\Lambda_{B,k}^{(\alpha_k)}(x_k, \ell)}{\bar{\Lambda}_{B,k}^{(\alpha_k)}(\ell)}, & s(\ell) = k \\ \frac{\Lambda_{S,k|k-1}^{(\alpha_k)}(x_k|x_{k-1}, \ell) \tau_{0:k-1}^{(\alpha_{s(\ell):k-1})}(x_{s(\ell):k-1}, \ell)}{\bar{\Lambda}_{S,k|k-1}^{(\alpha_{s(\ell):k})}(\ell)}, & t(\ell) = k > s(\ell) \\ \frac{Q_{S,k-1}(x_{k-1}, \ell) \tau_{0:k-1}^{(\alpha_{s(\ell):k-1})}(x_{s(\ell):k-1}, \ell)}{\bar{Q}_{S,k-1}^{(\alpha_{s(\ell):k-1})}(\ell)}, & t(\ell) = k - 1 \\ \tau_{0:t(\ell)}^{(\alpha_{s(\ell):t(\ell)})}(x_{s(\ell):t(\ell)}, \ell), & t(\ell) < k - 1 \end{cases}, \quad (7)$$

where  $\tau_{0:k}^{(\alpha_{s(\ell):k-1})}(x_{s(\ell):k-1}, \ell)$  is the trajectory posterior at time  $k - 1$ ,

$$\Lambda_{B,k}^{(\alpha_k)}(x, \ell) = \psi_{k,Z_k}^{(\alpha_k)}(x, \ell) P_{B,k}(\ell) f_{B,k}(x, \ell), \quad (8)$$

$$\bar{\Lambda}_{B,k}^{(\alpha_k)}(\ell) = \int \Lambda_{B,k}^{(\alpha_k)}(x, \ell) dx, \quad (9)$$

$$\Lambda_{S,k|k-1}^{(\alpha_k)}(x_k|x_{k-1}, \ell) = \psi_{k,Z_k}^{(\alpha_k)}(x_k, \ell) P_{S,k-1}(x_{k-1}, \ell) \times f_{S,k|k-1}(x_k|x_{k-1}, \ell), \quad (10)$$

$$\bar{\Lambda}_{S,k|k-1,k}^{(\alpha_{s(\ell):k})}(\ell) = \int \tau_{0:k-1}^{(\alpha_{s(\ell):k-1})}(x_{s(\ell):k-1}, \ell) \times \Lambda_{S,k|k-1}^{(\alpha_k)}(x_k|x_{k-1}, \ell) dx_{s(\ell):k}, \quad (11)$$

$$\bar{Q}_{S,k-1}^{(\alpha_{s(\ell):k-1})}(\ell) = \int \tau_{0:k-1}^{(\alpha_{s(\ell):k-1})}(x_{s(\ell):k-1}, \ell) \times Q_{S,k-1}(x_{k-1}, \ell) dx_{s(\ell):k-1}. \quad (12)$$

In addition, the association weight for  $\ell \in \mathbb{L}_k$  is defined as

$$\eta_{k|k-1}^{(\alpha_{s(\ell):k})}(\ell) = \begin{cases} \bar{\Lambda}_{B,k}^{(\alpha_k)}(\ell), & s(\ell) = k \\ \bar{\Lambda}_{S,k|k-1}^{(\alpha_{s(\ell):k})}(\ell), & t(\ell) = k > s(\ell) \\ \bar{Q}_{S,k-1}^{(\alpha_{s(\ell):k-1})}(\ell), & t(\ell) = k - 1 \\ Q_{B,k}(\ell), & \ell \in \mathbb{B}_k, \alpha_k = -1^V \end{cases}. \quad (13)$$

We now present the multi-object posterior as follows. Starting with the initial prior  $\pi_0(\mathbf{X}_0) = \delta_0[\mathcal{L}(\mathbf{X}_0)]$  (i.e., there are no live objects at the beginning) with weight  $w_0^{(\gamma_0)} = 1$ , the multi-sensor GLMB posterior at time  $k$  can be written as [11]

$$\pi_{0:k}(\mathbf{X}_{0:k}) \propto \Delta(\mathbf{X}_{0:k}) \sum_{\gamma_{0:k}} w_{0:k}^{(\gamma_{0:k})} \delta_{\mathcal{L}(\gamma_{0:k})}[\mathcal{L}(\mathbf{X}_{0:k})] [\tau_{0:k}^{(\gamma_{0:k} \circ \mathcal{L}(\cdot))}(\cdot)]^{\mathbf{X}_{0:k}}, \quad (14)$$

where  $\Delta(\mathbf{X}_{0:k}) \triangleq \prod_{i=0}^k \Delta(\mathbf{X}_i)$ , and

$$w_{0:k}^{(\gamma_{0:k})} = \prod_{j=1}^k w_j^{(\gamma_{0:j})}, \quad (15)$$

$$w_j^{(\gamma_{0:j})} = 1_{\mathcal{F}(\mathbb{B}_j \cup \mathcal{L}(\gamma_{j-1}))}(\mathcal{L}(\gamma_j)) [\eta_{j|j-1}^{(\gamma_{0:j}(\cdot))}(\cdot)]^{\mathbb{B}_j \cup \mathcal{L}(\gamma_{j-1})}. \quad (16)$$

Some of the relevant posterior statistics from the multi-scan GLMB are [11]:

- The cardinality distribution, i.e., the distribution of the number of trajectories is given by,

$$\Pr(|\mathcal{L}(\mathbf{X}_{0:k})| = n) = \sum_{\gamma_{0:k}} w_{0:k}^{(\gamma_{0:k})} \delta_n[|\mathcal{L}(\gamma_{0:k})|]; \quad (17)$$

- The cardinality distribution of births and deaths at time  $u \in \{0 : k\}$  are given by,

$$\begin{aligned} & \Pr(n \text{ births at time } u) \\ &= \sum_{\gamma_{0:k}} w_{0:k}^{(\gamma_{0:k})} \delta_n \left[ \sum_{\ell \in \mathcal{L}(\gamma_{0:k})} \delta_u[s(\ell)] \right]; \end{aligned} \quad (18)$$

$$\begin{aligned} & \Pr(n \text{ deaths at time } u) \\ &= \sum_{\gamma_{0:k}} w_{0:k}^{(\gamma_{0:k})} \delta_n \left[ \sum_{\ell \in \mathcal{L}(\gamma_{0:k})} \delta_u[t(\ell)] \right]; \end{aligned} \quad (19)$$

- The distribution of trajectory lengths is given by,

$$\Pr(\text{a trajectory has length } m) = \sum_{\gamma_{0:k}} \frac{w_{0:k}^{(\gamma_{0:k})}}{|\mathcal{L}(\gamma_{0:k})|} \sum_{\ell \in \mathcal{L}(\gamma_{0:k})} \delta_m[t(\ell) - s(\ell) + 1]. \quad (20)$$

Several multi-object trajectory estimators [11] are applicable to the GLMB posterior (14). This work uses the GLMB estimator, which first determines the most probable cardinality  $n^*$  by maximizing (17), and then the highest weighted component (indexed by)  $\gamma_{0:k}^*$  with cardinality  $n^*$ . The  $n^*$  estimated trajectories can be taken as the mean or mode of the trajectory posterior  $\tau_{0:k}^{(\gamma_{0:k}^*(\ell))}(\cdot, \ell)$  for each  $\ell \in \mathcal{L}(\gamma_{0:k}^*)$ .

The GLMB posterior (14) is completely parameterized by the set of components  $\{(w_{0:k}^{(\gamma_{0:k})}, \tau_{0:k}^{(\gamma_{0:k})})\}$ , indexed by  $\gamma_{0:k}$ , which grow super-exponentially in number, and hence a tractable approximation is necessary. Truncation by retaining a prescribed number of components with highest weights minimizes the  $L_1$ -norm approximation error [11].

### III. MULTI-SENSOR GLMB POSTERIOR PROPAGATION

This section presents techniques to truncate the multi-sensor GLMB posterior by extending the Gibbs samplers proposed in [11]. Subsection III-A introduces the distributions from which significant components of the GLMB posterior are drawn. Subsection III-B presents an efficient algorithm to generate valid components that can be used to initialize the full Gibbs sampler presented in Subsection III-C, which generates components according to a desired distribution.

#### A. Sampling Distributions

We truncate the posterior GLMB by sampling its components from some discrete distribution  $\pi$  such that those with higher weights are more likely to be chosen. Without loss of generality, we start with  $\mathcal{L}(\gamma_0) = \emptyset$ , and consider

$$\pi(\gamma_{0:k}) = \prod_{j=1}^k \pi^{(j)}(\gamma_j | \gamma_{0:j-1}), \quad (21)$$

where

$$\begin{aligned} \pi^{(j)}(\gamma_j | \gamma_{0:j-1}) &\propto 1_{\Gamma_j}(\gamma_j) 1_{\mathcal{F}(\mathbb{B}_j \uplus \mathcal{L}(\gamma_{j-1}))}(\mathcal{L}(\gamma_j)) \\ &\times [\vartheta_j^{(\gamma_{0:j}(\cdot))}(\cdot)]^{\mathbb{B}_j \uplus \mathcal{L}(\gamma_{j-1})}, \end{aligned} \quad (22)$$

and  $\vartheta_j^{(\gamma_{0:j}(\cdot))}(\cdot)$  is chosen to be approximately proportional to  $\eta_{j|j-1}^{(\gamma_{0:j}(\cdot))}(\cdot)$  so that (21) is approximately proportional to  $w_{0:k}^{(\gamma_{0:k})}$  in (15). A function  $f$  is said to be approximately proportional to  $g$ , i.e.,  $f \propto g$ , if  $f/\langle f, 1 \rangle \simeq g/\langle g, 1 \rangle$ . Note that, if  $\vartheta_j^{(\gamma_{0:j}(\cdot))}(\cdot)$  is equal to  $\eta_{j|j-1}^{(\gamma_{0:j}(\cdot))}(\cdot)$ , (22) is proportional to (16), and (21) is proportional to (15).

The term  $1_{\mathcal{F}(\mathbb{B}_j \uplus \mathcal{L}(\gamma_{j-1}))}(\mathcal{L}(\gamma_j))$  in (22) means  $\mathcal{L}(\gamma_j) \subseteq \mathbb{B}_j \uplus \mathcal{L}(\gamma_{j-1})$ , i.e.,  $\gamma_j(\ell) = -1^V$  for all  $\ell \notin \mathbb{B}_j \uplus \mathcal{L}(\gamma_{j-1})$ . Hence, only the values of  $\gamma_j$  on  $\mathbb{B}_j \uplus \mathcal{L}(\gamma_{j-1})$  need to be considered. In addition, the following decomposition of  $1_{\Gamma_i}(\gamma_i)$  is instrumental for the proposed Gibbs samplers (the proof is given in Supplementary Material, Subsection A).

*Lemma 1:* Let  $\bar{n} \triangleq \{1 : |\mathbb{L}_j|\} - \{n\}$ , and  $\Gamma_j(\bar{n})$  be the set of all  $\gamma_j(\ell_{\bar{n}}) \triangleq (\gamma_j(\ell_{1:n-1}), \gamma_j(\ell_{n+1:|\mathbb{L}_j|})) \in (\{-1\}^V \uplus \Lambda_j^{(1:V)})^{|\mathbb{L}_j|-1}$  that are positive 1-1. Then,

$$1_{\Gamma_j}(\gamma_j) = 1_{\Gamma_j(\bar{n})}(\gamma_j(\ell_{\bar{n}})) \prod_{v=1}^V \beta_n^{(j,v)}(\gamma_j^{(v)}(\ell_n) | \gamma_j^{(v)}(\ell_{\bar{n}})), \quad (23)$$

where

$$\begin{aligned} \beta_n^{(j,v)}(\gamma_j^{(v)}(\ell_n) | \gamma_j^{(v)}(\ell_{\bar{n}})) \\ = [1 - 1_{\{1:|\mathbb{Z}_j^{(v)}|\} \cap \{\gamma_j^{(v)}(\ell_{1:n-1}), \gamma_j^{(v)}(\ell_{n+1:|\mathbb{L}_j|})\}}(\gamma_j^{(v)}(\ell_n))]. \end{aligned} \quad (24)$$

#### B. Sampling From the Factors

Sampling from (21) can be performed by iteratively sampling from the factors (22), i.e.,  $\gamma_j \sim \pi(\cdot | \gamma_{0:j-1})$ , for time  $j = 1 : k$  (note that  $\gamma_j$  consists of  $\gamma_j(\ell_n)$ ,  $\ell_n \in \{\ell_{1:|\mathbb{L}_j|}\} \triangleq \mathbb{L}_j$ ). We sample from  $\pi(\cdot | \gamma_{0:j-1})$  using a Gibbs sampler [40], which constructs a sequence of iterates by generating the next iterate  $\gamma'_j$  from  $\gamma_j$  and the conditionals of  $\pi(\cdot | \gamma_{0:j-1})$  as follows

$$\begin{aligned} \gamma'_j(\ell_1) &\sim \pi_1^{(j)}(\cdot | \gamma_{0:j-1}, \gamma_j(\ell_{2:|\mathbb{L}_j|})) \\ &\vdots \\ \gamma'_j(\ell_n) &\sim \pi_n^{(j)}(\cdot | \gamma_{0:j-1}, \gamma'_j(\ell_{1:n-1}), \gamma_j(\ell_{n+1:|\mathbb{L}_j|})) \\ &\vdots \\ \gamma'_j(\ell_{|\mathbb{L}_j|}) &\sim \pi_{|\mathbb{L}_j|}^{(j)}(\cdot | \gamma_{0:j-1}, \gamma'_j(\ell_{1:|\mathbb{L}_j|-1})), \end{aligned}$$

where for any  $\alpha = \alpha^{(1:V)} \in \{-1\}^V \uplus \Lambda_j^{(1:V)}$

$$\begin{aligned} \pi_n^{(j)}(\alpha | \gamma'_j(\ell_{1:n-1}), \gamma_j(\ell_{n+1:|\mathbb{L}_j|}), \gamma_{0:j-1}) \\ \propto \pi^{(j)}(\gamma'_j(\ell_{1:n-1}), \alpha, \gamma_j(\ell_{n+1:|\mathbb{L}_j|}) | \gamma_{0:j-1}), \end{aligned} \quad (25)$$

The efficiency of Gibbs sampling hinges on the availability of conditionals that are easy to sample from.

We adopt the multi-sensor Gibbs sampling technique developed in [21] to sample  $\gamma'_j(\ell_n)$  from the  $n$ -th conditional (25). The basic steps in this technique are illustrated in Fig. 2, and summarized as pseudocode in Algorithm 1. The pseudocode for generating  $R$  iterates of the factor Gibbs sampler is given in Algorithm 2. Theoretical justifications and analyses of Algorithms 1 and 2 are given in the remainder of this subsection.

Consider first the  $n$ -th conditional distribution by substituting (22) into the right hand side of (25), and noting that we are only interested in the functional dependency on  $\alpha$ ,

$$\begin{aligned} \pi_n^{(j)}(\alpha | \gamma_j(\ell_{\bar{n}}), \gamma_{0:j-1}) \\ \propto 1_{\Gamma_j}((\gamma_j(\ell_{1:n-1}), \alpha, \gamma_j(\ell_{n+1:|\mathbb{L}_j|}))) \\ \times 1_{\mathcal{F}(\mathbb{B}_j \uplus \mathcal{L}(\gamma_{j-1}))}(\mathcal{L}((\gamma_j(\ell_{1:n-1}), \alpha, \gamma_j(\ell_{n+1:|\mathbb{L}_j|})))) \\ \times \vartheta_j^{((\gamma_{0:j-1}(\ell_n), \alpha))}(\ell_n), \end{aligned} \quad (26)$$

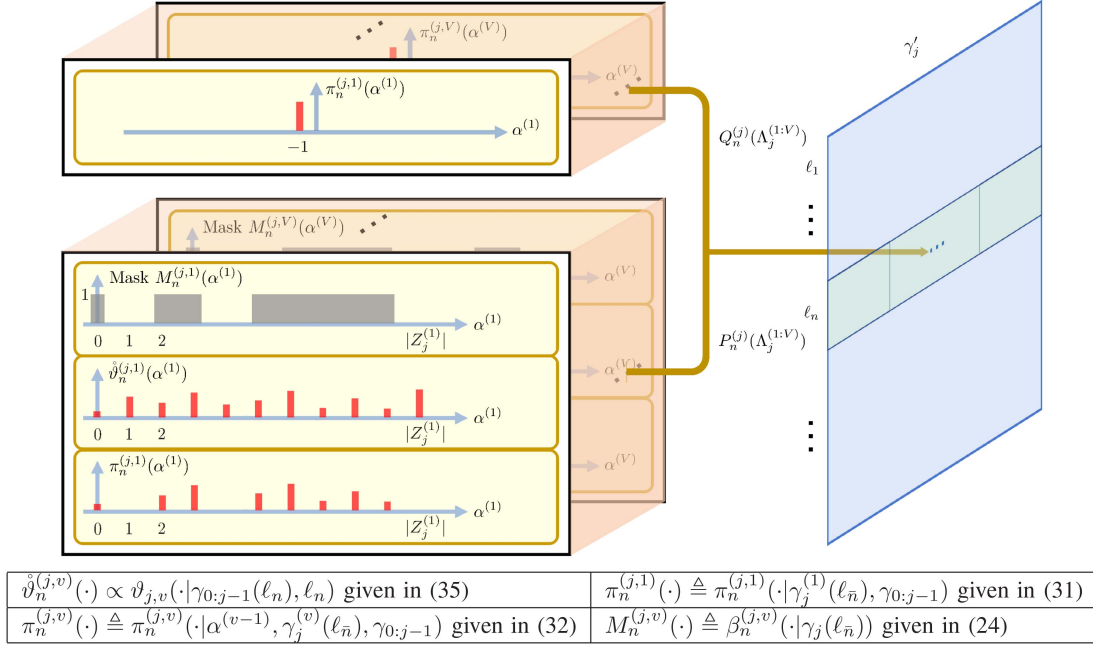


Fig. 2. Generating  $\gamma'_j(\ell_n) = (\alpha^{(1)}, \dots, \alpha^{(V)})$  from  $\gamma_j, \ell_n \in \mathbb{B}_j \uplus \mathcal{L}(\gamma_{j-1})$ . Two cases: (i) bottom branch, with probability  $P_n^{(j)}(\Lambda_j^{(1:V)})$ , given in (33),  $\ell_n$  is either born (if  $\ell_n \in \mathbb{B}_j$ ) or surviving (if  $\ell_n \in \mathcal{L}(\gamma_{j-1})$ ). For each  $v \in \{1 : V\}$ ,  $\alpha^{(v)}$  is sampled from the categorical distribution  $\pi_n^{(j,v)}(\cdot)$  that depends on the pre-computed  $\vartheta_n^{(j,v)}(\cdot)$ . To ensure that  $\gamma'_j$  is positive 1-1, we use  $M_n^{(j,v)}(\cdot)$  to mask out the positive measurement indices taken up by other labels (by multiplying  $\vartheta_n^{(j,v)}(\cdot)$  with  $M_n^{(j,v)}(\cdot)$ ), which results in  $\pi_n^{(j,v)}(\cdot)$ ; (ii) top branch, with probability  $Q_n^{(j)}(\Lambda_j^{(1:V)}) = 1 - P_n^{(j)}(\Lambda_j^{(1:V)})$ ,  $\ell_n$  is non-existent and the only allowable value for  $\alpha^{(v)}$  is  $-1$  for all  $v \in \{1 : V\}$ .

---

**Algorithm 1:** Factor-SampleCoord.

---

**Input :**  $G_{0:j-1} = (\gamma_{0:j-1}, w_{0:j-1}, \tau_{0:j-1}), \gamma'_j, n$   
**Output :**  $\gamma'_j(\ell_n)$

**for**  $v = 1 : V$  **do**

$[\vartheta_j^{(n,v)}(\alpha^{(v)}) := \vartheta_{j,v}(\alpha^{(v)} | \gamma_{0:j-1}(\ell_n), \ell_n)]_{\alpha^{(v)} = -1}^{|Z_j^{(v)}|}$  via (35);

Compute  $P_n^{(j)}(\Lambda_j^{(1:V)})$  via (33);  
 $Q_n^{(j)}(\Lambda_j^{(1:V)}) := 1 - P_n^{(j)}(\Lambda_j^{(1:V)})$ ;  
 $\epsilon \sim \text{Categorical}(["+", "-"], [P_n^{(j)}(\Lambda_j^{(1:V)}), Q_n^{(j)}(\Lambda_j^{(1:V)})])$ ;  
**if**  $\epsilon = "+"$  **then**

$\gamma'_j(\ell_n) \sim \text{SampleCoord}(P_n^{(j)}(\Lambda_j^{(1:V)}), \vartheta_j^{(n,v)}, \gamma'_j, n)$ ;  
**else**

$\gamma'_j(\ell_n) := -1 * \text{ones}(1, V)$ ;  
**end**

**end**

---

---

**Algorithm 1a:** SampleCoord.

---

**Input :**  $P(\Lambda_j^{(1:V)}), \vartheta_j^{(n,v)}, \gamma'_j, n$   
**Output :**  $[\phi_n^{(v)}]_{v=1}^V$

**for**  $v = 1 : V$  **do**

$c^{(v)} := [0 : |Z_j^{(v)}|]$ ;  
**for**  $\alpha^{(v)} = 0 : |Z_j^{(v)}|$  **do**

$p_j^{(n,v)}(\alpha^{(v)}) := \vartheta_j^{(n,v)}(\alpha^{(v)}) \beta_n^{(j,v)}(\alpha^{(v)} | \gamma_j^{(v)}(\ell_n))$ ;  
**if**  $v = 1$  **then**

$p_j^{(n,v)}(\alpha^{(v)}) := p_j^{(n,v)}(\alpha^{(v)}) P(\Lambda_j^{(1:V)})$ ;  
 $\phi_n^{(v)} \sim \text{Categorical}(c^{(v)}, [p_j^{(n,v)}(\alpha^{(v)})]_{v=1}^V)$ ;  
**end**

**end**

**end**

---

---

**Algorithm 2:** Factor-Gibbs.

---

**Input :**  $G_{0:j-1} = (\gamma_{0:j-1}, w_{0:j-1}, \tau_{0:j-1}), R$  (no. new samples)  
**Output :**  $[G_{0:k}^{(r)}]_{r=1}^R$

$P_j := |\mathbb{B}_j \uplus \mathcal{L}(\gamma_{j-1})|$ ;  
 $\gamma_j := \text{zeros}(P_j, V)$ ; (or any valid state)  
**for**  $r = 1 : R$  **do**

$\gamma'_j := \gamma_j$ ;  
**for**  $n = 1 : P_j$  **do**

$\gamma'_j(\ell_n) := \text{Factor-SampleCoord}(G_{0:j-1}, \gamma'_j, n)$ ;  
 $\gamma_{0:j} := [\gamma_{0:j-1}, \gamma'_j]$ ;  
 Compute  $w_{0:j}, \tau_{0:j}$  from  $\gamma_{0:j}$  via (15), (7);  
 $G_{0:j}^{(r)} := (\gamma_{0:j}, w_{0:j}, \tau_{0:j})$ ;  
**end**

**end**

**end**

---

where  $\gamma_j(\ell_{u:v}) \triangleq [\gamma_j(\ell_u), \dots, \gamma_j(\ell_v)]$ . Note that, given a positive 1-1 multi-sensor association map, any association map sampled from these conditionals is also positive 1-1.

In general, sampling  $\alpha$  directly from (26) is both memory and computationally expensive since  $\vartheta_j^{((\gamma_{0:j-1}(\ell_n), \alpha))}(\ell_n)$  needs to be evaluated for each of the  $1 + \prod_{v=1}^V (1 + |Z_j^{(v)}|)$  possible values of  $\alpha$ . Fortunately, the computational cost can be drastically reduced by using the strategy in [21] via the so-called minimally-Markovian conditional distributions.

*Definition 2:* The conditional (26) is said to be *Markovian* if

$$\vartheta_j^{((\gamma_{0:j-1}(\ell), \alpha))}(\ell) = \prod_{v=2}^V \vartheta_{j,v}(\alpha^{(v)} | \alpha^{(v-1)}, \gamma_{0:j-1}(\ell), \ell) \times \vartheta_{j,1}(\alpha^{(1)} | \gamma_{0:j-1}(\ell), \ell), \quad (27)$$

and *minimally-Markovian* if  $\vartheta_{j,v}$  can be written in the form

$$\vartheta_{j,v}(\alpha^{(v)} | \alpha^{(v-1)}, \gamma_{0:j-1}(\ell), \ell) = \vartheta_{j,v}(\alpha^{(v)} | \gamma_{0:j-1}(\ell), \ell) \mathbf{1}_{\{-1\}^2 \uplus \Lambda_j^{(v-1:v)}}(\alpha^{(v-1)}, \alpha^{(v)}). \quad (28)$$

The Markov property allows us to sample  $\alpha^{(1)}, \dots, \alpha^{(V)}$  individually as shown in Fig. 2, thereby alleviating the evaluations of  $\vartheta_j^{((\gamma_{0:j-1}(\ell_n), \alpha))}(\ell_n)$  over all possible values of  $\alpha$ . Better still, a complexity of  $\mathcal{O}(kVM)$ , where  $M = \max_{j \in 1:k} \max_{v \in 1:V} \{|Z_j^{(v)}|\}$ , can be achieved using minimally-Markovian conditionals whose explicit forms are given in the following Proposition (which follows from Corollary 4 of [21]).

**Proposition 3:** Let  $\gamma_j(\ell_{\bar{n}}) = (\gamma_j(\ell_{1:n-1}), \gamma_j(\ell_{n+1:|\mathbb{L}_j|}))$  be positive 1-1, and suppose that the conditional  $\pi_n^{(j)}(\cdot | \gamma_j(\ell_{\bar{n}}), \gamma_{0:j-1})$ , given by (26) is minimally-Markovian. Then, for  $\ell_n \in \mathbb{L}_j - (\mathbb{B}_j \uplus \mathcal{L}(\gamma_{j-1}))$ ,

$$\pi_n^{(j)}(\gamma_j(\ell_n) | \gamma_j(\ell_{\bar{n}}), \gamma_{0:j-1}) = 1_{\{-1\}^V}(\gamma_j(\ell_n)), \quad (29)$$

and for  $\ell_n \in \mathbb{B}_j \uplus \mathcal{L}(\gamma_{j-1})$ ,

$$\begin{aligned} & \pi_n^{(j)}(\gamma_j(\ell_n) | \gamma_j(\ell_{\bar{n}}), \gamma_{0:j-1}) \\ &= \left( \prod_{v=2}^V \pi_n^{(j,v)}(\gamma_j^{(v)}(\ell_n) | \gamma_j^{(v-1)}(\ell_n), \gamma_j^{(v)}(\ell_{\bar{n}}), \gamma_{0:j-1}) \right) \\ & \quad \times \pi_n^{(j,1)}(\gamma_j^{(1)}(\ell_n) | \gamma_j^{(1)}(\ell_{\bar{n}}), \gamma_{0:j-1}), \end{aligned} \quad (30)$$

where

$$\begin{aligned} & \pi_n^{(j,1)}(\alpha^{(1)} | \gamma_j^{(1)}(\ell_{\bar{n}}), \gamma_{0:j-1}) \\ &= \begin{cases} 1 - P_n^{(j)}(\Lambda_j^{(1:V)}), & \alpha^{(1)} = -1 \\ P_n^{(j)}(\Lambda_j^{(1:V)}) \beta_n^{(j,1)}(\alpha^{(1)} | \gamma^{(1)}(\ell_{\bar{n}})) \\ \quad \times \frac{\vartheta_{j,1}(\alpha^{(1)} | \gamma_{0:j-1}(\ell_n), \ell_n)}{\Upsilon_n^{(j,1)}}, & \alpha^{(1)} > -1 \end{cases} \end{aligned} \quad (31)$$

$$\begin{aligned} & \pi_n^{(j,v)}(\alpha^{(v)} | \alpha^{(v-1)}, \gamma_j^{(v)}(\ell_{\bar{n}}), \gamma_{0:j-1}) \\ &= \begin{cases} 1, & \alpha^{(v-1)}, \alpha^{(v)} = -1 \\ \beta_n^{(j,v)}(\alpha^{(v)} | \gamma_j^{(v)}(\ell_{\bar{n}})) \\ \quad \times \frac{\vartheta_{j,v}(\alpha^{(v)} | \gamma_{0:j-1}(\ell_n), \ell_n)}{\Upsilon_n^{(j,v)}}, & \alpha^{(v-1)}, \alpha^{(v)} > -1 \end{cases} \end{aligned} \quad (32)$$

for  $v \in \{2 : V\}$ , and

$$\begin{aligned} P_n^{(j)}(\Lambda_j^{(1:V)}) &\triangleq \frac{\prod_{v=1}^V \Upsilon_n^{(j,v)}}{\prod_{v=1}^V \vartheta_{j,v}(-1 | \gamma_{0:j-1}(\ell_n), \ell_n) + \prod_{v=1}^V \Upsilon_n^{(j,v)}}, \\ \Upsilon_n^{(j,v)} &\triangleq \sum_{i=0}^{M_j^{(v)}} \beta_n^{(j,v)}(i | \gamma_j^{(v)}(\ell_{\bar{n}})) \vartheta_{j,v}(i | \gamma_{0:j-1}(\ell_n), \ell_n). \end{aligned} \quad (33)$$

In addition to the minimally-Markovian property (for the desired computational complexity), recall from (22) that

$$\vartheta_j^{((\gamma_{0:j-1}(\ell), \alpha))}(\ell) \propto \eta_{j|j-1}^{(\gamma_{0:j-1}(\ell), \alpha)}(\ell), \quad (34)$$

where  $\eta_{j|j-1}^{(\gamma_{0:j-1}(\ell), \alpha)}(\ell)$  is defined in (13), and can be interpreted as the (unnormalized) probability that (conditioned on  $\gamma_{0:j-1}(\ell)$  and  $Z_{1:j}$ ) label  $\ell$  generates measurements  $z_{\alpha^{(1)}}, \dots, z_{\alpha^{(V)}}$  at time

$j$  (with the understanding that  $\alpha^{(v)} = 0$  refers to a miss-detection and  $\alpha^{(v)} = -1$  refers to a non-existence), abbreviated as  $\Pr_j(\ell \sim z_{\alpha^{(1)}}, \dots, z_{\alpha^{(V)}})$ . This can be accomplished by choosing

$$\vartheta_{j,v}(\alpha^{(v)} | \gamma_{0:j-1}(\ell), \ell) \propto \eta_{j|j-1}^{(\gamma_{0:j-1}(\ell), (v; \alpha^{(v)}))}(\ell), \quad (35)$$

where

$$\begin{aligned} & \eta_{j|j-1}^{(\alpha_{s(\ell):j-1}, (v; \alpha^{(v)}))}(\ell) \\ & \triangleq \begin{cases} \bar{\Lambda}_{B,j}^{(v; \alpha^{(v)})}(\ell), & s(\ell) = j \\ \bar{\Lambda}_{S,j|j-1}^{(\alpha_{s(\ell):j-1}, (v; \alpha^{(v)}))}(\ell), & t(\ell) = j > s(\ell) \\ \bar{Q}_{S,j-1}^{(\alpha_{s(\ell):j-1})}(\ell), & t(\ell) = j - 1 \\ Q_{B,j}(\ell), & \ell \in \mathbb{B}_j, \alpha^{(v)} = -1 \end{cases}, \end{aligned} \quad (36)$$

$$\bar{\Lambda}_{B,j}^{(v; \alpha^{(v)})}(\ell) = \int \psi_{j, Z_j^{(v)}}^{(v, \alpha^{(v)})}(x, \ell) P_{B,j}(\ell) f_{B,j}(x, \ell) dx, \quad (37)$$

$$\begin{aligned} & \bar{\Lambda}_{S,j|j-1}^{(\alpha_{s(\ell):j-1}, (v; \alpha^{(v)}))}(\ell) = \int \tau_{0:j-1}^{(\alpha_{s(\ell):j-1})}(x_{s(\ell):j-1}, \ell) \\ & \quad \times P_{S,j-1}(x_{j-1}, \ell) f_{S,j|j-1}(x_j | x_{j-1}, \ell) \psi_{j, Z_j^{(v)}}^{(v, \alpha^{(v)})}(x_j, \ell) dx_{s(\ell):j}, \end{aligned} \quad (38)$$

$s(\ell)$  and  $t(\ell)$  in (36) are, respectively, the earliest and latest times on  $\{0 : j\}$  such that  $\ell$  exists, and  $\tau_{0:j-1}^{(\alpha_{s(\ell):j-1})}(x_{s(\ell):j-1}, \ell)$  is the trajectory posterior of  $\ell$  at time  $j-1$ . Eq. (36) can be interpreted as the (unnormalized) probability that (conditioned on  $\gamma_{0:j-1}(\ell)$  and  $Z_{1:j}$ ) label  $\ell$  generates measurement  $z_{\alpha^{(v)}}^{(v)}$  at time  $j$ , abbreviated as  $\Pr_j(\ell \sim z_{\alpha^{(v)}}^{(v)})$ .

The rationale for choosing (35) can be seen by substituting (35) into (28) and (27), which gives

$$\vartheta_j^{((\gamma_{0:j-1}(\ell), \alpha))}(\ell) \propto \prod_{v=1}^V \eta_{j|j-1}^{(\gamma_{0:j-1}(\ell), (v; \alpha^{(v)}))}(\ell), \quad (39)$$

i.e.,  $\vartheta_j^{((\gamma_{0:j-1}(\ell), \alpha))}(\ell)$  is the product of the probabilities  $\Pr_j(\ell \sim z_{\alpha^{(v)}}^{(v)})$ ,  $v = 1 : V$ . Consequently, the approximation in (34) boils down to approximating  $\Pr_j(\ell \sim z_{\alpha^{(1)}}, \dots, z_{\alpha^{(V)}})$  by  $\Pr_j(\ell \sim z_{\alpha^{(1)}}^{(1)}) \times \dots \times \Pr_j(\ell_n \sim z_{\alpha^{(V)}}^{(V)})$ . This is reasonable, because the event space is  $\alpha \in \{-1\}^V \uplus \Lambda_j^{(1:V)}$ , and conditioned on  $Z_{1:j}$  and  $\gamma_{0:j-1}$ , the probability that  $\ell$  generating a measurement from one sensor is almost independent from generating a measurement from another sensor. Furthermore, it follows from [21] that, the support of (22) with the minimally-Markovian property contains the support of (16). Algorithm 2 uses the minimally-Markovian strategy above.

Note that although this approach produces valid multi-sensor association history  $\gamma_{0:k}$ , in order to produce samples distributed according to (21), the Gibbs sampler for each factor  $\pi_n^{(j)}$  needs to be iterated for a sufficiently long time (i.e., executing Algorithm 2 with a large  $R$ ) before proceeding to the next timestep. While this is not efficient, sampling from the factors can provide a cheap way to generate valid  $\gamma_{0:k}$  to initialize the full Gibbs sampler (for faster convergence) presented in the next subsection.

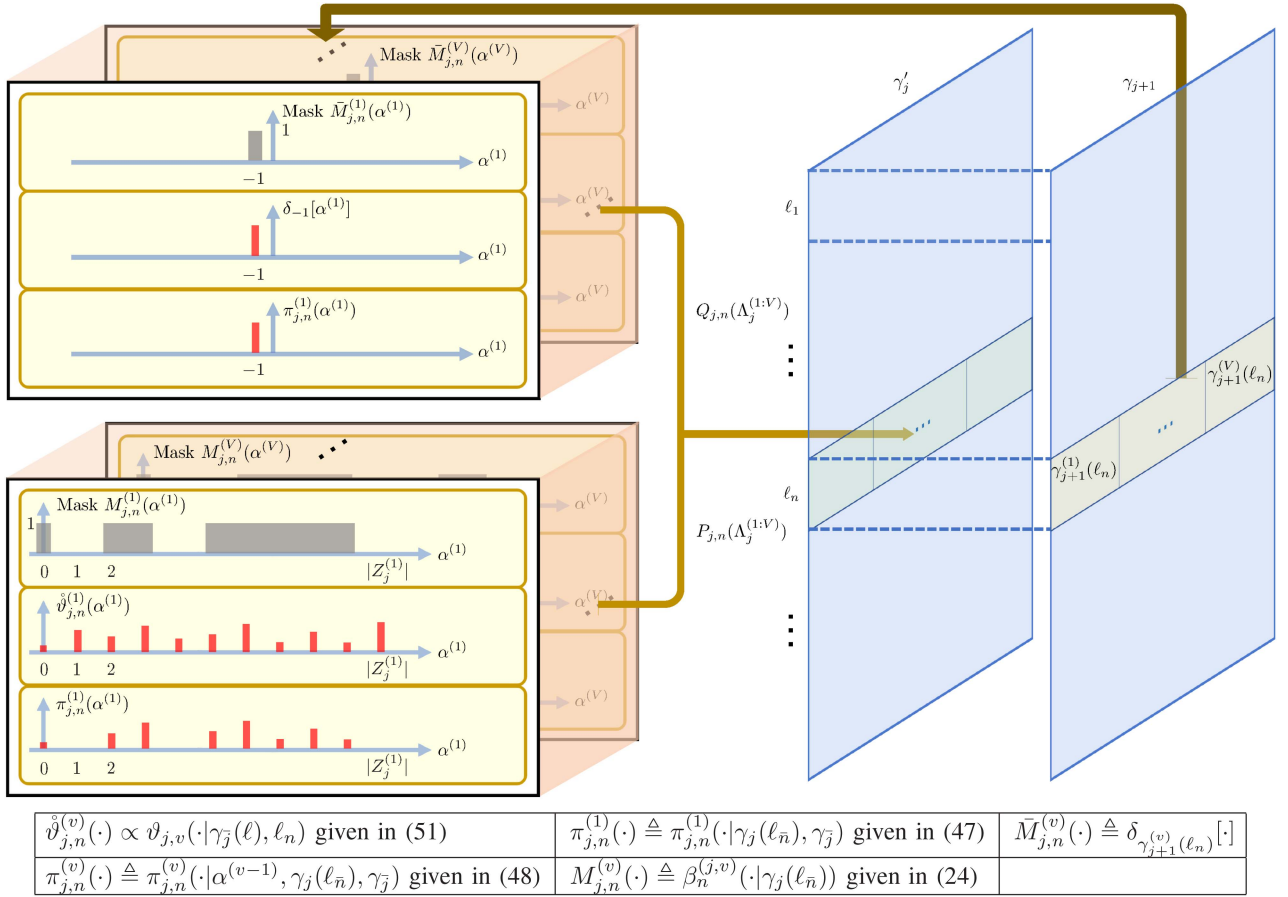


Fig. 3. Generating  $\gamma'_j(\ell_n) = (\alpha^{(1)}, \dots, \alpha^{(V)})$  from  $\gamma_j$ ,  $j \in \{1 : k\}$ ,  $\ell_n \in \mathbb{B}_j \uplus \mathcal{L}(\gamma_{j-1})$ . Two cases: (i) bottom branch, with probability  $P_{j,n}(\Lambda_j^{(1:V)})$ , given in (49),  $\ell_n$  is either born (if  $\ell_n \in \mathbb{B}_j$ ) or surviving (if  $\ell_n \in \mathcal{L}(\gamma_{j-1})$ ). For each  $v \in \{1 : V\}$ ,  $\alpha^{(v)}$  is sampled from the categorical distribution  $\pi_{j,n}^{(v)}(\cdot)$  that depends on the pre-computed  $\vartheta_{j,n}^{(v)}(\cdot)$ . To ensure that  $\gamma'_j$  is positive 1-1 (and  $\gamma'_{0:k} \in \Gamma_{0:k}$ ), we mask out the positive indices of sensor  $v$  that have already been allocated to other labels (by multiplying  $\vartheta_{j,n}^{(v)}(\cdot)$  with the mask  $M_{j,n}^{(v)}(\cdot)$  to produce  $\pi_{j,n}^{(v)}(\cdot)$ ); (ii) top branch, with probability  $Q_{j,n}(\Lambda_j^{(1:V)}) = 1 - P_{j,n}(\Lambda_j^{(1:V)})$ ,  $\ell_n$  is non-existent, and the only allowable value for  $\alpha^{(v)}$  is  $-1$  for all  $v \in \{1 : V\}$ . In addition, to ensure  $\gamma'_{0:k} \in \Gamma_{0:k}$ , a non-existent  $\ell_n$  at time  $j$  must remain non-existent thereafter. Therefore,  $\pi_{j,n}^{(v)}(\cdot)$  must be 0 if  $\gamma_{j+1}^{(v)}(\ell_n)$  is non-negative. This is accomplished by multiplying  $\delta_{-1}[\cdot]$  with the mask  $\bar{M}_{j,n}^{(v)}(\cdot)$ .

For completeness, the pseudocode to generate a set of valid  $\gamma_{0:k}$  is given in Algorithm 3, where the factor sampler (Algorithm 2) is used to sample  $\gamma_j$  for times  $j = 1 : k$ . Let  $M_j \triangleq \max_{v \in \{1:V\}} \{|Z_j^{(v)}|\}$ ,  $P_j \triangleq |\mathbb{B}_j \uplus \mathcal{L}(\gamma_{j-1})|$ , and  $R$  be the number of new samples generated at time  $j$ . Then, the complexity of this algorithm is  $\mathcal{O}(R \sum_{j=1}^k P_j^2 V M_j)$  (although it can be implemented in  $\mathcal{O}(R \sum_{j=1}^k P_j (P_j + V M_j))$  in practice by pre-calculating  $\beta_n^{(k,v)}$  values). Denoting  $P \triangleq \max_{j \in \{1:k\}} \{P_j\}$ , and  $\bar{M} \triangleq \max_{j \in \{1:k\}} \{M_j\}$ , indicatively, the complexity of the algorithm is  $\mathcal{O}(k R P^2 V \bar{M})$ . The Gibbs sampler described in Factor-Gibbs (pseudocode given in Algorithm 2) converges to the stationary distribution (22) at an exponential rate [21, Proposition 2].

### C. The Full Gibbs Sampler

Instead of sampling from each factor, a full Gibbs sampler for (21) constructs a sequence of iterates, where the next iterate  $\gamma'_{0:k}$

#### Algorithm 3: Factor-SampleJoint.

**Input** :  $Q$  (max samples),  $R$

**Output** :  $[G_{0:k}^{(q)}]_{q=1}^Q$

Initialize  $G_0^{(1)} = (\gamma_0, w_0, \tau_0)$ ;

**for**  $j = 1 : k$  **do**

$Q_0 = \min(Q, \text{size}(G_{0,j-1}))$ ;

**for**  $q = 1 : Q_0$  **do**

$[G_{0:j}^{(q,r)}]_{r=1}^R := \text{Factor-Gibbs}(G_{0,j-1}^{(q)}, R)$ ;

$[G_{0:j}^{(q)}]_{q=1}^{Q_0} := Q$  best of  $[G_{0:j}^{(q,r)}]_{(q,r)=(1,1)}$ ;

Normalize weights  $[w_{0:j}^{(q)}]_{q=1}^{Q_0}$ ;

is generated from the current  $\gamma_{0:k}$  by sampling  $\gamma'_j(\ell_n)$  from the conditional

$$\begin{aligned} & \pi_{j,n}(\alpha | \overbrace{\gamma'_{0:j-1}}^{\text{past}}, \overbrace{\gamma'_j(\ell_{1:n-1})}^{\text{current (processed)}}, \overbrace{\gamma_j(\ell_{n+1:|\mathbb{L}_j|})}^{\text{current (unprocessed)}}, \overbrace{\gamma_{j+1:k}}^{\text{future}}) \\ & \propto \pi(\gamma'_{0:j-1}, \gamma'_j(\ell_{1:n-1}), \alpha, \gamma_j(\ell_{n+1:|\mathbb{L}_j|}), \gamma_{j+1:k}). \end{aligned} \quad (40)$$



for each  $j \in \{1 : k\}$ ,  $\ell_n \in \{\ell_{1:\mathbb{L}_j}\}$ .

The proposed algorithm for sampling  $\gamma'_j(\ell_n)$  from the conditional  $\pi_{j,n}$  is illustrated in Fig. 3, and summarized as pseudocode in Algorithm 4. The pseudocode for generating  $T$  iterates of the full Gibbs sampler is given in Algorithm 5. Theoretical justifications and analyses of Algorithm 4 and 5 are given in the remainder of this subsection. Mathematical proofs are given in the Supplementary Material.

**Lemma 4:** Let  $\gamma_{\bar{j}} \triangleq (\gamma_{0:j-1}, \gamma_{j+1:k})$ , and  $a \vee b$  denotes  $\min\{a, b\}$ . Then, the conditional (40) can be expressed as

$$\begin{aligned} & \pi_{j,n}(\alpha|\gamma_j(\ell_n), \gamma_{\bar{j}}) \\ & \propto 1_{\Gamma_j}((\gamma_j(\ell_{1:n-1}), \alpha, \gamma_j(\ell_{n+1:\mathbb{L}_j}))) \\ & \quad \times 1_{\mathcal{F}(\mathbb{B}_j \uplus \mathcal{L}(\gamma_{j-1}))}(\mathcal{L}((\gamma_j(\ell_{1:n-1}), \alpha, \gamma_j(\ell_{n+1:\mathbb{L}_j})))) \\ & \quad \times 1_{\mathcal{F}(\mathbb{B}_{j+1} \uplus \mathcal{L}((\gamma_j(\ell_{1:n-1}), \alpha, \gamma_j(\ell_{n+1:\mathbb{L}_j}))))}(\mathcal{L}(\gamma_{j+1})) \\ & \quad \times \vartheta_j(\alpha|\gamma_{\bar{j}}(\ell_n), \ell_n). \end{aligned} \quad (41)$$

where

$$\vartheta_j(\alpha|\gamma_{\bar{j}}(\ell_n), \ell_n) \triangleq \prod_{i=j}^{k \vee (t(\ell_n)+1)} \vartheta_i^{(\gamma_{0:j-1}(\ell_n), \alpha, \gamma_{j+1:i}(\ell_n))}(\ell_n). \quad (42)$$

Similar to sampling from the factors, in general, sampling  $\alpha$  from the conditional (41) is both memory and computationally expensive since  $\vartheta_j(\alpha|\gamma_{\bar{j}}(\ell_n), \ell_n)$  needs to be evaluated for each of the  $1 + \prod_{v=1}^V (1 + |Z_j^{(v)}|)$  possible values of  $\alpha$ . Further, since each  $\vartheta_j(\alpha|\gamma_{\bar{j}}(\ell_n), \ell_n)$  is a product of terms that need to be evaluated for  $i = j : k \vee (t(\ell_n) + 1)$ , it is more computationally expensive than the factor sampler. Again, the computational cost can be drastically reduced using minimally-Markovian conditional distributions.

**Definition 5:** The conditional (41) is said to be *Markovian* if

$$\begin{aligned} \vartheta_j(\alpha|\gamma_{\bar{j}}(\ell), \ell) &= \prod_{v=2}^V \vartheta_{j,v}(\alpha^{(v)}|\alpha^{(v-1)}, \gamma_{\bar{j}}(\ell), \ell) \\ & \quad \times \vartheta_{j,1}(\alpha^{(1)}|\gamma_{\bar{j}}(\ell), \ell), \end{aligned} \quad (43)$$

and *minimally-Markovian* if  $\vartheta_{j,v}$  can be written in the form

$$\begin{aligned} \vartheta_{j,v}(\alpha^{(v)}|\alpha^{(v-1)}, \gamma_{\bar{j}}(\ell), \ell) &= \\ \vartheta_{j,v}(\alpha^{(v)}|\gamma_{\bar{j}}(\ell), \ell) 1_{\{-1\}^2 \uplus \Lambda_j^{(v-1:v)}}(\alpha^{(v-1)}, \alpha^{(v)}). \end{aligned} \quad (44)$$

The Markovian conditional allows us to sample individual  $\alpha^{(1)}, \alpha^{(2)}, \dots, \alpha^{(V)}$ , alleviating the evaluation of  $\vartheta_j(\alpha|\gamma_{\bar{j}}(\ell_n), \ell_n)$  over all possible values of  $\alpha$ , but still incurs a total complexity of  $\mathcal{O}(kVM^2|t(\ell_n) - s(\ell_n)|)$  for computing normalization constants. Further, a linear complexity of  $\mathcal{O}(kVM|t(\ell_n) - s(\ell_n)|)$  can be achieved using the minimally-Markovian conditional given in the following Proposition, which extends Corollary 4 of [21] to multi-scan.

---

**Algorithm 4:** Full-SampleCoord.

---

**Input** :  $\gamma_{0:k}, \gamma'_j, j, n$   
**Output** :  $\gamma'_j(\ell_n)$

---

**for**  $v = 1 : V$  **do**  
 $\lfloor \vartheta_j^{(n,v)}(\alpha^{(v)}) := \vartheta_{j,v}(\alpha^{(v)}|\gamma_{\bar{j}}(\ell_n), \ell_n)|_{\alpha^{(v)}=-1}^{|Z_j^{(v)}|}$  via (51);  
 Compute  $P_{j,n}(\Lambda_j^{(1:V)})$  via (49);  
 $Q_{j,n}(\Lambda_j^{(1:V)}) := 1 - P_{j,n}(\Lambda_j^{(1:V)})$ ;  
 $\epsilon \sim \text{Categorical}(["+", "-", [P_{j,n}(\Lambda_j^{(1:V)})], [Q_{j,n}(\Lambda_j^{(1:V)})]])$ ;  
**if**  $\epsilon = "+"$  **then**  
 $\lfloor \gamma'_j(\ell_n) \sim \text{SampleCoord}(P_{j,n}(\Lambda_j^{(1:V)}), \vartheta_j^{(n,v)}, \gamma'_j, n)$ ;  
**else**  
 $\lfloor \gamma'_j(\ell_n) := -1 * \text{ones}(1, V)$ ;

---



---

**Algorithm 5:** Full-Gibbs.

---

**Input** :  $G_{0:k} = (\gamma_{0:k}, w_{0:k}, \tau_{0:k}), T$  (no. samples)  
**Output** :  $[G_{0:k}^{(t)}]_{t=1}^T$

---

**for**  $t = 1 : T$  **do**  
**for**  $j = 1 : k$  **do**  
 $P_j := |\mathbb{B}_j \uplus \mathcal{L}(\gamma_{j-1})|$ ;  $\gamma'_j := \gamma_j$ ;  
**for**  $n = 1 : P_j$  **do**  
 $\lfloor \gamma'_j(\ell_n) := \text{Full-SampleCoord}(\gamma_{0:k}, \gamma'_j, j, n)$ ;  
 $\gamma_{0:j} := [\gamma_{0:j-1}, \gamma'_j]$ ;  
 Compute  $w_{0:j}, \tau_{0:j}$  from  $\gamma_{0:j}$  via (15), (7);  
 $G_{0:k}^{(t)} := (\gamma_{0:k}, w_{0:k}, \tau_{0:k})$ ;

---

**Proposition 6:** Consider  $\gamma_j$  of a valid association history  $\gamma_{0:k}$ ,  $j \in \{1 : k\}$ , and define

$$M_j(\alpha; \beta) \triangleq \begin{cases} \delta_{\beta}[\alpha], & \alpha \in \{-1\}^V \\ 1, & \alpha \in \Lambda_j^{(1:V)}. \end{cases}$$

Suppose that the conditional  $\pi_{j,n}(\alpha|\gamma_j(\ell_n), \gamma_{\bar{j}})$  given by (41) is minimally-Markovian. Then, for  $\ell_n \in \mathbb{L}_j - (\mathbb{B}_j \uplus \mathcal{L}(\gamma_{j-1}))$ ,

$$\begin{aligned} \pi_{j,n}(\gamma_j(\ell_n)|\gamma_j(\ell_n), \gamma_{\bar{j}}) &= 1_{\{-1\}^V}(\gamma_j(\ell_n)) \delta_{\gamma_{\min\{j+1, k\}}(\ell_n)}[\gamma_j(\ell_n)], \end{aligned} \quad (45)$$

and for  $\ell_n \in \mathbb{B}_j \uplus \mathcal{L}(\gamma_{j-1})$

$$\begin{aligned} \pi_{j,n}(\gamma_j(\ell_n)|\gamma_j(\ell_n), \gamma_{\bar{j}}) &= \left( \prod_{v=2}^V \pi_{j,n}^{(v)}(\gamma_j^{(v)}(\ell_n)|\gamma_j^{(v-1)}(\ell_n), \gamma_j(\ell_n), \gamma_{\bar{j}}) \right) \\ & \quad \times \pi_{j,n}^{(1)}(\gamma_j^{(1)}(\ell_n)|\gamma_j(\ell_n), \gamma_{\bar{j}}) \\ & \quad \times M_j(\gamma_j(\ell_n); \gamma_{\min\{k, j+1\}}(\ell_n)), \end{aligned} \quad (46)$$

where

$$\pi_{j,n}^{(1)}(\alpha^{(1)}|\gamma_j(\ell_n), \gamma_{\bar{j}}) \quad (47)$$

$$= \begin{cases} 1 - P_{j,n}(\Lambda_j^{(1:V)}), & \alpha^{(1)} = -1 \\ P_{j,n}(\Lambda_j^{(1:V)}) \beta_n^{(j,1)}(\alpha^{(1)}|\gamma_j^{(1)}(\ell_n)) \\ \times \frac{\vartheta_{j,1}(\alpha^{(1)}|\gamma_{\bar{j}}(\ell_n), \ell_n)}{\Upsilon_{j,n}^{(1)}}, & \alpha^{(1)} > -1 \end{cases}$$

$$\pi_{j,n}^{(v)}(\alpha^{(v)}|\alpha^{(v-1)}, \gamma_j(\ell_n), \gamma_{\bar{j}})$$

$$= \begin{cases} 1, & \alpha^{(v-1)}, \alpha^{(v)} = -1 \\ \beta_n^{(j,v)}(\alpha^{(v)}|\gamma_j^{(v)}(\ell_n)) \\ \times \frac{\vartheta_{j,v}(\alpha^{(v)}|\gamma_j^{(v)}(\ell_n), \ell_n)}{\Upsilon_{j,n}^{(v)}}, & \alpha^{(v-1)}, \alpha^{(v)} > -1 \end{cases} \quad (48)$$

for  $v \in \{2 : V\}$ , and

$$P_{j,n}(\Lambda_j^{(1:V)}) \triangleq \frac{\prod_{v=1}^V \Upsilon_{j,n}^{(v)}}{\prod_{v=1}^V \vartheta_{j,v}(-1|\gamma_j^{(v)}(\ell_n), \ell_n) + \prod_{v=1}^V \Upsilon_{j,n}^{(v)}},$$

$$\Upsilon_{j,n}^{(v)} \triangleq \sum_{\alpha^{(v)}=0}^{|\mathcal{Z}_j^{(v)}|} \beta_n^{(j,v)}(\alpha^{(v)}|\gamma_j^{(v)}(\ell_n)) \vartheta_{j,v}(\alpha^{(v)}|\gamma_j^{(v)}(\ell_n), \ell_n). \quad (49)$$

In addition to the minimally-Markovian property, we require  $\vartheta_j^{(\gamma_{0:j}(\ell))}(\ell)$  to be approximately proportional to  $\eta_{j|j-1}^{(\gamma_{0:j}(\ell))}(\ell)$  (see (22)). Since  $\vartheta_j(\alpha|\gamma_j(\ell_n), \ell)$  in (41) is a product of  $\vartheta_i^{(\gamma_{0:j-1}(\ell), \alpha, \gamma_{j+1:i}(\ell))}(\ell)$ 's,  $i = j : k \vee (t(\ell) + 1)$ , this translates to

$$\vartheta_i^{(\gamma_{0:j-1}(\ell), \alpha, \gamma_{j+1:i}(\ell))}(\ell) \propto \eta_{i|i-1}^{(\gamma_{0:j-1}(\ell), \alpha, \gamma_{j+1:i}(\ell))}(\ell) \quad (50)$$

for each  $i$ , where  $\eta_{i|i-1}^{(\gamma_{0:j-1}(\ell), \alpha, \gamma_{j+1:i}(\ell))}(\ell)$  can be interpreted as the (unnormalized) probability that (conditioned on  $\gamma_j(\ell)$  and  $Z_{1:k}$ ) label  $\ell$  generates measurements  $z_{\alpha^{(1)}}^{(1)}, \dots, z_{\alpha^{(V)}}^{(V)}$  at time  $j$ , abbreviated as  $\Pr_j(\ell_n \sim z_{\alpha^{(1)}}^{(1)}, \dots, z_{\alpha^{(V)}}^{(V)})$ . To fulfill (50) and the minimally-Markovian property, we choose

$$\vartheta_{j,v}(\alpha^{(v)}|\gamma_j(\ell), \ell) \propto \prod_{i=j}^{k \vee (t(\ell)+1)} \eta_{i|i-1}^{(\gamma_{0:j-1}(\ell), (v; \alpha^{(v)}), \gamma_{j+1:i}(\ell))}(\ell), \quad (51)$$

where

$$\eta_{i|i-1}^{(\alpha_{0:j-1}, (v; \alpha^{(v)}), \alpha_{j+1:i})}(\ell) \triangleq \begin{cases} \bar{\Lambda}_{B,i}^{((v; \alpha^{(v)}), \alpha_{j+1:i})}(\ell), & s(\ell) = j \\ \bar{\Lambda}_{S,i|i-1}^{(\alpha_s(\ell); j-1, (v; \alpha^{(v)}), \alpha_{j+1:i})}(\ell), & t(\ell) = j > s(\ell) \\ \bar{Q}_{S,i-1}^{(\alpha_s(\ell); j-1)}(\ell), & t(\ell) = j - 1 \\ Q_{B,i}(\ell), & \ell \in \mathbb{B}_j, \alpha^{(v)} = -1 \end{cases}, \quad (52)$$

$$\Lambda_{S,j+1:i}^{(\alpha_{j+1:i})}(x_{j:i}, \ell) = \prod_{q=j+1}^i \Lambda_{S,q|q-1}^{(\alpha_q)}(x_q|x_{q-1}, \ell),$$

$$\bar{\Lambda}_{B,i}^{((v; \alpha^{(v)}), \alpha_{j+1:i})}(\ell) = \int \tau_{0;j-1}^{(\alpha_s(\ell); j-1)}(x_{s(\ell); j-1}, \ell) P_{B,j}(\ell) f_{B,j}(x_j, \ell) \times \psi_{j, Z_j^{(v)}}^{(v, \alpha^{(v)})}(x_j, \ell) \Lambda_{S,j+1:i}^{(\alpha_{j+1:i})}(x_{j:i}, \ell) dx_{s(\ell); i},$$

$$\bar{\Lambda}_{S,i|i-1}^{(\alpha_s(\ell); j-1, (v; \alpha^{(v)}), \alpha_{j+1:i})}(\ell)$$

$$= \int \tau_{0;j-1}^{(\alpha_s(\ell); j-1)}(x_{s(\ell); j-1}, \ell) P_{S,j-1}(x_{j-1}, \ell) \times f_{S,j|j-1}(x_j|x_{j-1}, \ell) \psi_{j, Z_j^{(v)}}^{(v, \alpha^{(v)})}(x_j, \ell) \Lambda_{S,j+1:i}^{(\alpha_{j+1:i})}(x_{j:i}, \ell) \times dx_{s(\ell); i},$$

$s(\ell)$  and  $t(\ell)$  in (52) are, respectively, the earliest and latest times on  $\{j : i\}$  such that  $\ell$  exists. The right hand side of Eq. (51) can be interpreted as the (unnormalized) probability that (conditioned on  $\gamma_j(\ell)$  and  $Z_{1:k}$ ) label  $\ell$  generates measurement  $z_{\alpha^{(v)}}^{(v)}$  at time  $j$ , abbreviated as  $\Pr_j(\ell \sim z_{\alpha^{(v)}}^{(v)})$ .

The rationale for choosing (51) can be seen by substituting (51) into (43) and (44), which gives

$$\vartheta_j(\alpha|\gamma_j(\ell), \ell) \propto \prod_{v=1}^V \prod_{i=j}^{k \vee (t(\ell)+1)} \eta_{i|i-1}^{(\gamma_{0:j-1}(\ell), (v; \alpha^{(v)}), \gamma_{j+1:i}(\ell))}(\ell),$$

i.e., the product of probabilities  $\Pr_j(\ell_n \sim z_{\alpha^{(v)}}^{(v)})$ ,  $v = 1 : V$ . Consequently, the approximation in (50) boils down to approximating  $\Pr_j(\ell_n \sim z_{\alpha^{(1)}}^{(1)}, \dots, z_{\alpha^{(V)}}^{(V)})$  by  $\Pr_j(\ell_n \sim z_{\alpha^{(1)}}^{(1)}) \times \dots \times \Pr_j(\ell_n \sim z_{\alpha^{(V)}}^{(V)})$ . This is reasonable, because the event space is  $\alpha \in \{-1\}^V \uplus \Lambda_j^{(1:V)}$ , and conditioned on  $Z_{1:k}$  and  $\gamma_j$ , the probability that  $\ell$  generating a measurement from one sensor is almost independent from generating a measurement from another sensor. Furthermore, similar to [21], the support of (21) with the minimally-Markovian property contains the support of (15) as per Proposition 7.

*Proposition 7:* The support of (21) with minimally-Markovian property contains the support of (15).

Our proposed full Gibbs sampler (Algorithm 5), uses the minimally-Markovian strategy above. Note that, once each  $\gamma_j'(\ell_n)$  is sampled, the corresponding trajectory posterior and the association weight are updated using (7) and (13). The complexity of this algorithm is  $\mathcal{O}(TV \sum_{j=1}^k P_j^2 M_j)$  (although it can be implemented in  $\mathcal{O}(T \sum_{j=1}^k (P_j^2 + P_j V M_j))$  by pre-calculating  $\beta_n^{(k,v)}$  values), where  $T$  denotes the number of generated new samples,  $M_j \triangleq \max_{v \in \{1:V\}} \{|\mathcal{Z}_j^{(v)}|\}$  and  $P_j \triangleq |\mathbb{B}_j \uplus \mathcal{L}(\gamma_{j-1})|$ . Indicatively, this complexity is  $\mathcal{O}(kTV P^2 \dot{M})$ , where  $\dot{M} \triangleq \max_{j \in \{1:k\}} \{M_j\}$  and  $P \triangleq \max_{j \in \{1:k\}} \{P_j\}$ .

Similar to the single-sensor case [11], the full Gibbs sampler has an exponential convergence rate.

*Proposition 8:* Starting from any valid initial state  $\gamma_{1:k} \in \Gamma_{1:k}$ , the Gibbs sampler described by Algorithm 5 converges to the stationary distribution (21) at an exponential rate. More specifically, let  $\pi^j$  denote the  $j$ -th power of the transition kernel. Then,

$$\max_{\gamma_{1:k}, \dot{\gamma}_{1:k} \in \Gamma_{1:k}} (|\pi^j(\dot{\gamma}_{1:k}|\gamma_{1:k}) - \pi(\dot{\gamma}_{1:k})|) \leq (1 - 2\beta)^{\lfloor \frac{j}{k} \rfloor}$$

where  $h = k + 1$ ,  $\beta \triangleq \min_{\gamma_{1:k}, \dot{\gamma}_{1:k} \in \Gamma_{1:k}} \pi^h(\dot{\gamma}_{1:k}|\gamma_{1:k}) > 0$  is the least likely  $h$ -step transition probability.

Algorithm 6 provides the pseudocode for batch computation of the multi-sensor GLMB posterior (14) using the full Gibbs sampler (Algorithm 5). Note that the factor sampler is used

**Algorithm 6:** Batch.

---

**Input** :  $Q, R, T, Q_0$  (no. fact. samples),  
 $H$  (no. full samples)  
**Output** :  $[G_{0:k}^{(h)}]_{h=1}^H$

---

$[G_{0:k}^{(q)}]_{q=1}^Q := \text{Factor-SampleJoint}(Q, R);$   
 $[G_{0:k}^{(q)}]_{q=1}^{Q_0} := Q_0$  best of  $[G_{0:k}^{(q)}]_{q=1}^Q;$   
**for**  $q = 1 : Q_0$  **do**  
   $[G_{0:k}^{(q,t)}]_{t=1}^T := \text{Unique}(\text{Full-Gibbs}(G_{0:k}^{(q)}, T));$   
 $[G_{0:k}^{(h)}]_{h=1}^H := H$  best of  $\text{Unique}([G_{0:k}^{(q,t)}]_{(q,t)=(1,1)}^{(Q_0, T)});$   
 Normalize weights  $[w_{0:k}^{(h)}]_{h=1}^H;$

---

**Algorithm 7:** Smoothing-While-Filtering.

---

**Input** :  $[G_{0:k-1}^{(h)}]_{h=1}^{H_{k-1}}, [T^{(h)}]_{h=1}^{H_{k-1}}, T$   
**Output** :  $[G_{0:k}^{(h)}]_{h=1}^{H_k}$

---

**for**  $h = 1 : H_{k-1}$  **do**  
   $[G_{0:k}^{(h,t)}]_{t=1}^T := \text{Unique}(\text{Factor-Gibbs}(G_{0:k-1}^{(h)}, T^{(h)}));$   
 $[G_{0:k}^{(h)}]_{h=1}^{H_k} := \bar{H}_k$  best of  $[G_{0:k}^{(h,t)}]_{(h,t)=(1,1)}^{(H_{k-1}, T^{(h)})};$   
**for**  $h = 1 : H_k$  **do**  
   $[G_{0:k}^{(h,t)}]_{t=1}^T := \text{Full-Gibbs}(G_{0:k}^{(h)}, T);$   
 $[G_{0:k}^{(h)}]_{h=1}^{H_k} := H_k$  best of  $\text{Unique}([G_{0:k}^{(h,t)}]_{h,t=(1,1)}^{(H_k, T)});$   
 Normalize weights  $[w_{0:k}^{(h)}]_{h=1}^{H_k};$

---

to initialize the full Gibbs sampler with a set of  $Q_0$  valid association histories, each of which is used to generate a set of  $T$  samples distributed according to (21). Since Algorithm 6 can be executed in parallel for each  $q$ , indicatively its complexity is also  $\mathcal{O}(kTV P^2 M)$ . Alternatively, the posterior (14) can be recursively propagated, i.e., smoothing-while-filtering, as shown in Algorithm 7. Note that Algorithms 6 and 7 both produce the same posterior with the same complexity. However, the smoothing-while-filtering algorithm allows us to obtain an estimate of the multi-object trajectory at each point in time, while the batch algorithm is an offline method, where the multi-object trajectory can only be estimated after all the observations have been collected. For a fixed complexity per time step, a moving window-based implementation can be adopted, which incurs an  $\mathcal{O}(LTV P^2 M)$  complexity per time step, where  $L$  is the length of the window.

## IV. NUMERICAL EXPERIMENTS

This section evaluates the performance of the proposed (multi-scan) multi-sensor GLMB smoother and benchmarks it against the multi-sensor GLMB filter via simulations on single-sensor, two-sensors and four-sensor scenarios.

Births, deaths and movements of an unknown and time-varying number of objects are simulated in a 3D surveillance region over  $k = 100$  s. The objects' 3D positions are measured using multiple sensors with severely limited observability (detection probability) and high measurement noise. There are 12 births at  $j = 1$  s, 20 s, 40 s, 60 s and 80 s (respectively 3, 3, 2, 2, and 2 objects), and the probability of survival is set at  $P_S(x^{(\ell_i)}, \ell_i) = 0.95$ . Two of the objects born at time  $j = 1$  s die at 70 s, and the peak number of 10 live objects occurs from 80 s onwards. The three objects born at  $j = 1$  s cross paths at

$[0, -400, 0]^T$  at time 40 s, and two of the objects born at time 20 s cross paths at  $[300, -200, 200]^T$  at time 59 s, constituting a more challenging multi-object tracking scenario.

The kinematic state of an object is represented by a 6D state vector, i.e.,  $x_k = [p_{x,k}, \dot{p}_{x,k}, p_{y,k}, \dot{p}_{y,k}, p_{z,k}, \dot{p}_{z,k}]^T$  consisting of 3D position and velocity, and follows a constant velocity motion model. The single object transition density is linear Gaussian and given by  $f_{S,k|k-1}(x_{k+1}^{(\ell)} | x_k^{(\ell)}) = \mathcal{N}(x_{k+1}^{(\ell)}; F_k x_k^{(\ell)}, Q_k)$  where

$$F_k = I_3 \otimes \begin{bmatrix} 1 & \Delta \\ 0 & 1 \end{bmatrix}, \quad Q_k = \sigma_a^2 I_3 \otimes \begin{bmatrix} \frac{\Delta^4}{4} & \frac{\Delta^3}{2} \\ \frac{\Delta^3}{2} & \Delta^2 \end{bmatrix},$$

$I_3$  is the  $3 \times 3$  identity matrix,  $\Delta = 1$  s is the sampling time,  $\sigma_a = 5$  m/s<sup>2</sup>, and  $\otimes$  denotes the matrix outer product. Births are modeled by a Labeled Multi-Bernoulli (LMB) Process with (birth and spatial distribution) parameters  $\{r_{B,k}(\ell_i), p_{B,k}^{(i)}(\ell_i)\}_{i=1}^4$ , where  $\ell_i = (k, i) \in \mathbb{B}_k$ ,  $r_{B,k}(\ell_i) = 0.03$ ,  $p_{B,k}^{(i)}(x^{(\ell_i)}, \ell_i) = \mathcal{N}(x^{(\ell_i)}; m_{B,k}^{(i)}, Q_{B,k})$ ,

$$m_{B,k}^{(1)} = (0.1, 0, 0.1, 0, 0.1, 0)^T,$$

$$m_{B,k}^{(2)} = (400, 0, -600, 0, 200, 0)^T,$$

$$m_{B,k}^{(3)} = (-800, 0, -200, 0, -400, 0)^T,$$

$$m_{B,k}^{(4)} = (-200, 0, 800, 0, 600, 0)^T,$$

and  $Q_{B,k} = \text{diag}([10, 10, 10, 10, 10, 10]^2)$ .

The noisy 3D position of an object captured by sensor  $v$  in the surveillance region  $[-1000, 1000] \times [-1000, 1000] \times [-1000, 1000] m^3$ , takes the form  $z_k^{(v)} = [z_{x,k}^{(v)}, z_{y,k}^{(v)}, z_{z,k}^{(v)}]^T$ , and modeled by the linear gaussian likelihood function  $g_k^{(v)}(z_k^{(v)} | x_k^{(\ell)}) = \mathcal{N}(z_k^{(v)}; H_k^{(v)} x_k^{(\ell)}, R_k^{(v)})$ , where  $H_k^{(v)} = I_3 \otimes [1 \ 0]$ , and  $R_k^{(v)} = \text{diag}([20, 20, 20]^2)$ . The probability of detection of each sensor is  $P_D^{(v)}(x^{(\ell_i)}, \ell_i) = 0.3$ , and clutter is modeled as a Poisson RFS with intensity  $\kappa_k^{(v)}(z) = 3.75 \times 10^{-10} m^{-3}$  over the surveillance region, i.e., a clutter rate of 3 per scan. This scenario is more challenging than those in [11], [21] due to far lower detection probability and higher measurement noise in a cluttered 3D environment.

We plot single runs of the multi-sensor GLMB smoother and filter for the same sets of measurements under the single-sensor, two-sensor and four-sensor cases, and provide an analysis of the results in the remainder of this section.

## A. Filtering Vs. Smoothing

The multi-sensor GLMB smoother (Algorithm 6) is run with 100 scans, and 1000 valid initial samples generated via factor sampling. Its performance is compared with the single-sensor GLMB filter (Algorithm 2 of [41]), single-sensor multi-scan GLMB smoother (Algorithm 3 of [11] with the same parameters as the proposed multi-sensor smoother), and multi-sensor GLMB filter (Algorithm 3 of [21] using the minimally-Markovian strategy with the same multi-sensor parameters). The smoothing problem involves 400 dimensional rank assignment problems with approximately 10 variables in each dimension.

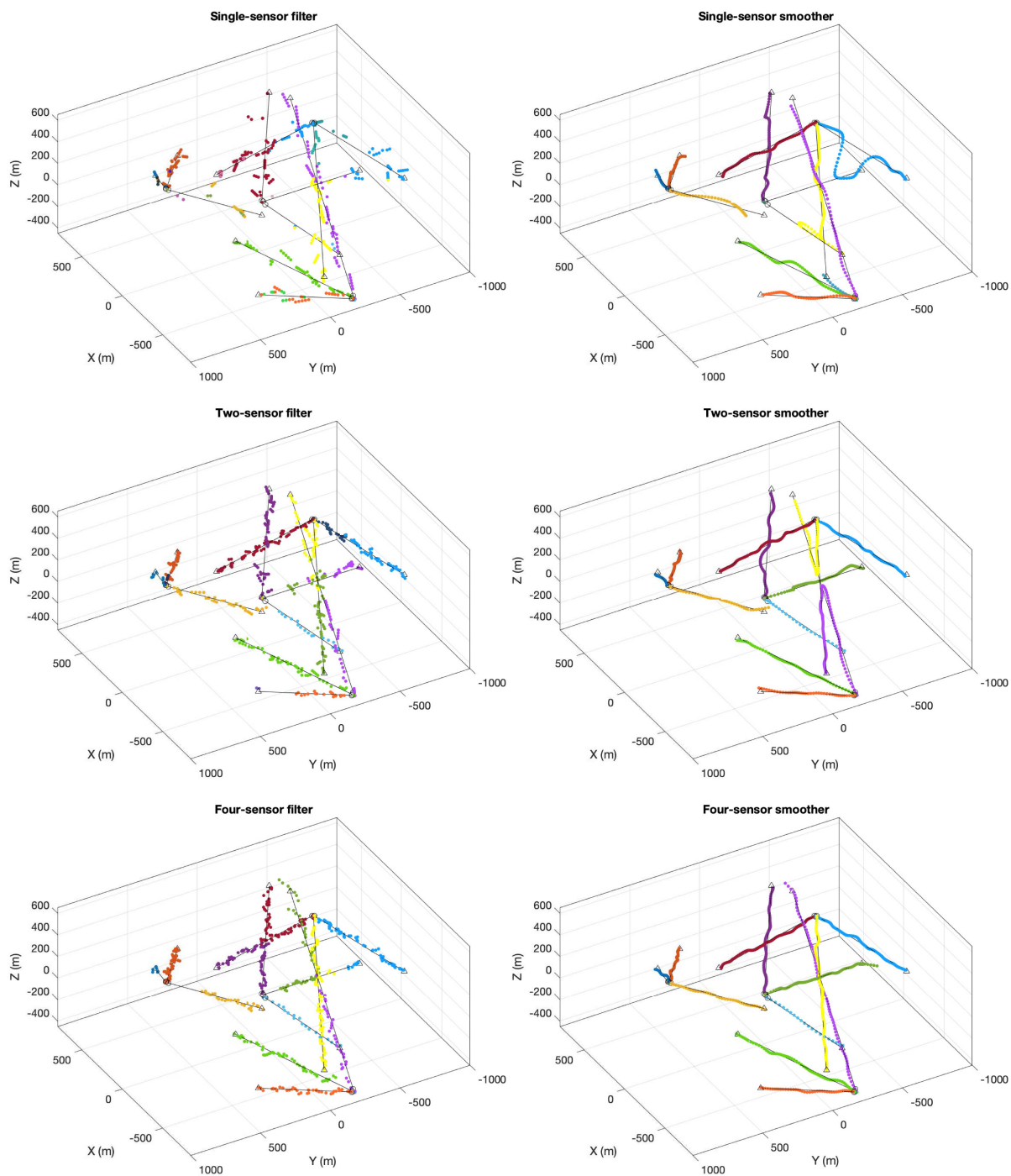


Fig. 4. Estimated trajectories (in colored dots) from GLMB filtering and GLMB smoothing superimposed on true trajectories (in black). Starting and stopping positions of objects are shown with  $\circ$  and  $\triangle$  respectively.

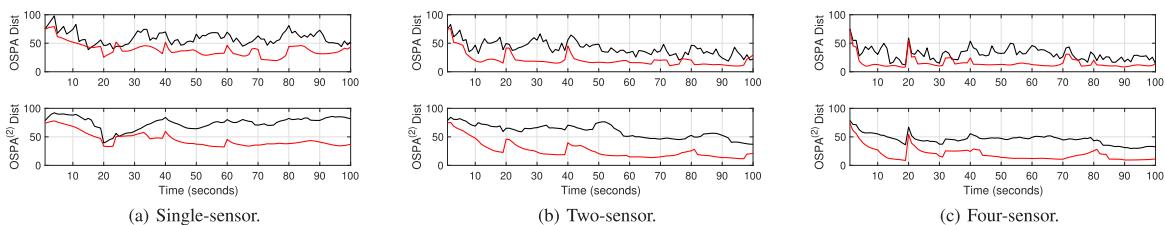


Fig. 5. OSPA and OSPA<sup>(2)</sup> performance of GLMB filtering (in black) and GLMB smoothing (in red).

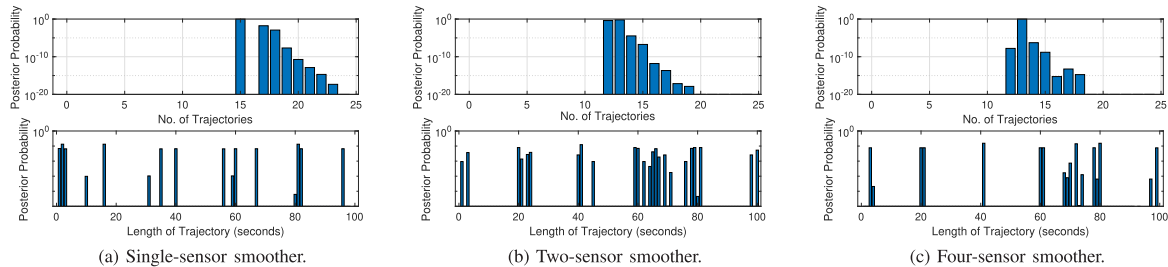


Fig. 6. Posterior distributions of the number of trajectories, and posterior distributions of trajectory lengths.

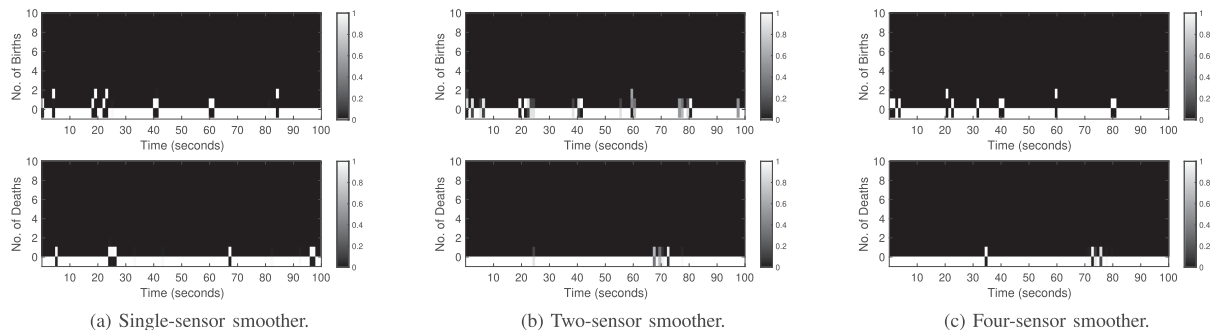


Fig. 7. Posterior distributions of the births and deaths at each timestep.

State-of-the-art solvers with dedicated hardware cannot handle problems with such large dimensions [17].

Fig. 4 shows the estimated trajectories from multi-sensor GLMB filter and multi-sensor GLMB smoother (for single-sensor, two-sensor and four-sensor scenarios) superimposed on the true trajectories. Due to the challenging signal settings, the single-sensor GLMB filter produces several fragmented tracks, fails to estimate large portions of some tracks, and yields significant errors. Increasing the number of sensors to two and four results in fewer fragmented tracks, produce estimates for almost all portions of truth tracks, and improves estimates. However, it does not prevent track switchings (at  $[0, -400, 0]^T$  and  $[300, -200, 200]^T$ , respectively, for the two-sensor and four-sensor scenarios) from occurring at track crossings.

The multi-sensor GLMB smoother outperforms the filter on each scenario. The single-sensor scenario shows worse performance than the multiple sensor scenarios due to lower observability, resulting in fragmented tracks, undetected tracks and significant errors in two tracks. In the two-sensor scenario, except for the two track switchings at  $[0, -400, 0]^T$ , all other tracks promptly start and terminate, resulting in smooth trajectory estimates with smaller errors. The four-sensor scenario produces superior results due to increased observability. It handles the track crossings well, resulting in no track switchings, nor fragmented tracks, and produces even smaller errors with each pertinent track starting and terminating promptly.

Fig. 5 compares OSPA and OSPA<sup>(2)</sup> errors over time between the multi-sensor GLMB smoother (final estimate) and multi-sensor GLMB filter. The OSPA and OSPA<sup>(2)</sup> parameters are  $c = 100m$ ,  $p = 1$ , with a scan window size of 10 for OSPA<sup>(2)</sup>. It is clear that increasing the number of sensors improves the performance of the multi-sensor GLMB filter. The smoother

produces significantly lower errors due to its ability correct earlier errors in the multi-sensor assignments.

### B. Posterior Statistics Computation

This section illustrates the proposed multi-object posterior capability to provide useful statistical information [11] about the ensemble of multi-object trajectories (mathematical expressions are given in eqs. (17)–(20)). Fig. 6 shows the probability distribution of the number of trajectories. As the number of sensors increases, the mode of this distribution settles at 13, although the actual number of trajectories is 12. This mismatch arises from a short (3s) false track appearing near the birth location  $(-800, -200, -400)^T$  due to a false measurement falling close to it (recall that the detection probability of each sensor is 0.3). From the distribution of the length of trajectories shown in Fig. 6, it can be seen that except for the mode at time  $j = 3$  s (corresponding to the short false track), the posterior correctly captures the modes at times 20s, 40s, 60s, 80s and 100s in the four-sensor scenario. Also, note that the uncertainty at time 80s is higher in the two-sensor scenario than the four-sensor scenario. Fig. 7 shows the birth and death cardinality distributions against time. Except for the birth and death around time 30s (due to the short false track), the smoother correctly identifies all instances of birth and deaths in the four-sensor scenario.

## V. CONCLUSION

In this paper, we developed an efficient numerical solution to the multi-sensor multi-object smoothing problem via Gibbs sampling, which unifies the implementation techniques for multi-sensor GLMB filtering and multi-scan GLMB smoothing

in [11], [21]. The proposed solution, which involves solving large-scale NP-hard multi-dimensional assignment problems, reduces to that of [11] when the number of sensors is one and to that of [21] when the number of scans is one. Theoretical justification and analysis of the proposed algorithms are also presented. Numerical studies, conducted for the first time with 100 scans and up to four sensors, show that excellent tracking performance can be achieved despite having sensors with poor detection probabilities (as low as 0.3) by combining them over a longer integration time. This also demonstrates the benefits of multi-sensor fusion for low observability sensors, which is vital for applications in harsh signal environments such as underwater.

## REFERENCES

- [1] R. Mahler, *Statistical Multisource-Multitarget Information Fusion*. Norwood, MA, USA: Artech House, 2007.
- [2] S. S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*. Norwood, MA, USA: Artech House, 1999.
- [3] R. Mahler, *Advances in Statistical Multisource-Multitarget Information Fusion*. Norwood, MA, USA: Artech House, 2014.
- [4] Y. Bar-Shalom, P. K. Willett, and X. Tian, *Tracking and Data Fusion: A Handbook of Algorithms*. Bradford, U.K.: YBS Publishing, 2011.
- [5] R. Szeliski, *Computer Vision: Algorithms and Applications*. Berlin, Germany: Springer, 2010.
- [6] J. Ong, B.-T. Vo, B.-N. Vo, D. Y. Kim, and S. Nordholm, "A bayesian filter for multi-view 3D multi-object tracking with occlusion handling," *IEEE Trans. Pattern Anal. Mach. Intel.*, vol. 44, no. 5, pp. 2246–2263, May 2022.
- [7] N. Ishtiaq, A. K. Gostar, A. Bab-Hadiashar, and R. Hosein-zhad, "Interaction-aware labeled multi-bernoulli filter," 2022, *arXiv:2204.08655*.
- [8] E. Meijering, O. Dzyubachyk, and I. Smal, "Methods for cell and particle tracking," in *Imaging and Spectroscopic Analysis of Living Cells: Optical and Spectroscopic Techniques*. vol. 504, Cambridge, MA, USA: Academic Press, 2012, pp. 183–200.
- [9] T. T. D. Nguyen, B.-N. Vo, B.-T. Vo, D. Y. Kim, and Y. S. Choi, "Tracking cells and their lineages via labeled random finite sets," *IEEE Trans. Signal Process.*, vol. 69, pp. 5611–5626, 2021.
- [10] Y. Bar-Shalom and T. E. Fortmann, *Tracking and Data Association*. Cambridge, MA, USA: Academic Press, 1988.
- [11] B.-N. Vo and B.-T. Vo, "A multi-scan labeled random finite set model for multi-object state estimation," *IEEE Trans. Signal Process.*, vol. 67, no. 19, Oct. 2019, Art. no. 4948.
- [12] M. E. Liggins, D. L. Hall, and J. Llinas, *Handbook of Multisensor Data Fusion: Theory and Practice*, 2nd ed., Boca Raton, FL, USA: CRC Press, 2008.
- [13] X. Chen, R. Tharmarasa, and T. Kirubarajan, "Multitarget Multisensor Tracking," in *Academic Press Library in Signal Processing*. vol. 2, Amsterdam, Netherlands: Elsevier, 2014, pp. 759–812.
- [14] S. Thrun et al., "Stanley: The robot that won the DARPA grand challenge," *J. Field Robot.*, vol. 23, no. 9, 2006, Art. no. 661.
- [15] P. Groves, *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems*, 2nd ed., Morristown, NJ, USA: Artech, 2013.
- [16] Y. Bar-Shalom and X.-R. Li, *Multitarget-Multisensor Tracking: Principles and Techniques*. Bradford, U.K.: YBS publishing, 1995.
- [17] O. Reynen, S. Vadrevu, R. Nagi, and K. LeGrand, "Large-scale multi-dimensional assignment: Problem formulations and GPU accelerated solutions," in *Proc. 22th Int. Conf. Inf. Fusion*, 2019, pp. 1–8.
- [18] N. T. Pham, W. Huang, and S. H. Ong, "Multiple sensor multiple object tracking with GMPHD filter," in *Proc. IEEE 10th Int. Conf. Inf. Fusion*, 2007, pp. 1–7.
- [19] 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.
- [20] A.-A. Saucan, M. J. Coates, and M. Rabbat, "A multisensor multi-bernoulli filter," *IEEE Trans. Signal Process.*, vol. 65, no. 20, pp. 5495–5509, Oct. 2017.
- [21] B. N. Vo, B. T. Vo, and M. Beard, "Multi-sensor multi-object tracking with the generalized labeled multi-bernoulli filter," *IEEE Trans. Signal Process.*, vol. 67, no. 23, pp. 5952–5967, Dec. 2019.
- [22] M. Uney, D. E. Clark, and S. J. Julier, "Distributed fusion of PHD filters via exponential mixture densities," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 3, pp. 521–531, Jun. 2013.
- [23] G. Battistelli, L. Chisci, C. Fantacci, A. Farina, and R. Mahler, "Distributed fusion of multitarget densities and consensus PHD/CPHD filters," *Proc. SPIE*, vol. 9474, 2015, pp. 122–136.
- [24] G. Battistelli, L. Chisci, C. Fantacci, A. Farina, and A. Graziano, "Consensus CPHD filter for distributed multitarget tracking," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 3, pp. 508–520, Jun. 2013.
- [25] M. B. Guldogan, "Consensus bernoulli filter for distributed detection and tracking using multi-static doppler shifts," *IEEE Signal Process. Lett.*, vol. 24, no. 6, pp. 672–676, Jun. 2014.
- [26] W. Yi, M. Jiang, R. Hosein-zhad, and B. Wang, "Distributed multi-sensor fusion using generalised multi-bernoulli densities," *IET Radar, Sonar Navig.*, vol. 11, no. 3, pp. 434–443, Mar. 2016.
- [27] B. Wang, W. Yi, R. Hosein-zhad, S. Li, L. Kong, and X. Yang, "Distributed fusion with multi-bernoulli filter based on generalized covariance intersection," *IEEE Trans. Signal Process.*, vol. 65, no. 1, pp. 242–255, Jan. 2017.
- [28] C. Fantacci, B.-N. Vo, B.-T. Vo, G. Battistelli, and L. Chisci, "Robust fusion for multisensor multiobject tracking," *IEEE Signal Process. Lett.*, vol. 25, no. 5, pp. 640–644, May 2018.
- [29] O. E. Drummond, "Multiple target tracking with multiple frame, probabilistic data association," in *Proc. SPIE Conf. Signal Data Proc. Small Targets*, 1993, pp. 394–408.
- [30] T. Vu, B.-N. Vo, and R. J. Evans, "A particle marginal metropolis-hastings multi-target tracker," *IEEE Trans. Signal Process.*, vol. 62, No 15, pp. 3953–3964, Aug. 2014.
- [31] D. Reid, "An algorithm for tracking multiple targets," *IEEE Trans. Autom. Control*, vol. 24, no. 6, pp. 843–854, Dec. 1979.
- [32] A. K. Mahalanabis, B. Zhou, and N. K. Bose, "Improved multi-target tracking in clutter by PDA smoothing," *IEEE Trans. Aerosp. Electron. Sys.*, vol. 26, no. 1, pp. 113–121, Jan. 1990.
- [33] S. P. Puranik and J. K. Tugnait, "Multisensor tracking of multiple maneuvering targets using multiscan JPDA and IMM filtering," in *Proc. Amer. Control Conf.*, 2005, pp. 2058–2063.
- [34] R. Mahler, "Exact closed-form multitarget bayes filters," *Sensors*, vol. 19, no. 12, 2019, Art. no. 2818.
- [35] R. Mahler, "Mathematical representation of multitarget systems," 2022, *arXiv:2203.10972*.
- [36] S. Sarkka, *Bayesian Filtering and Smoothing*. Cambridge, U.K.: Cambridge Univ. Press, 2013.
- [37] B.-N. Vo and B.-T. Vo, "Labeled random finite sets and multi-object conjugate priors," *IEEE Trans. Signal Process.*, vol. 61, no 13, pp. 3460–3475, Jul. 2013.
- [38] C. Andrieu, A. Doucet, and R. Holenstein, "Particle Markov chain monte carlo methods," *J. Roy. Statist. Soc. B-Stat. Method.*, vol. 72, no 3, pp. 269–352, 2010.
- [39] R. Mahler, "Multitarget bayes filtering via first-order multitarget moments," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 39, no. 4, pp. 1152–1178, Oct. 2003.
- [40] C. P. Robert and G. Casella, *Monte Carlo Statistical Methods*, 2nd ed., Berlin, Germany: Springer, 2004.
- [41] B.-N. Vo, B.-T. Vo, and H. G. Hoang, "An efficient implementation of the generalized labeled multi-bernoulli filter," *IEEE Trans. Signal Process.*, vol. 65, no. 8, pp. 1975–1987, Apr. 2017.
- [42] H. Taylor and S. Karlin, *An Introduction to Stochastic Modeling*, 3rd ed. New York, NY, USA: Academic Press, 1998.
- [43] G. Roberts and A. Smith, "Simple conditions for the convergence of the gibbs sampler and metropolis-hastings algorithms," *Stochastic Process. Appl.*, vol. 49, no. 2, pp. 207–216, 1994.
- [44] R. Gallager, *Stochastic Processes: Theory for Applications*. Cambridge, U.K.: Cambridge Univ. Press, 2014.