Labeled Random Finite Sets and the Bayes Multi-Target Tracking Filter

Ba-Ngu Vo*, Ba-Tuong Vo†, Dinh Phung‡

***Abstract*—An analytic solution to the multi-target Bayes recursion known as the δ-Generalized Labeled Multi-Bernoulli (δ-GLMB) filter has been recently proposed in [1]. As a sequel to [1], this paper details efficient implementations of the δ-GLMB multi-target tracking filter. Each iteration of this filter involves an update operation and a prediction operation, both of which result in weighted sums of multi-target exponentials with intractably large number of terms. To truncate these sums, the ranked assignment and K-th shortest path algorithms are used in the update and prediction, respectively, to determine the most significant terms without exhaustively computing all of the terms. In addition, using tools derived from the same framework, such as probability hypothesis density filtering, we present inexpensive (relative to the δ-GLMB filter) look-ahead strategies to reduce the number of computations. Characterization of the L_1-error in the multi-target density arising from the truncation is presented.**

***Index Terms*—Random finite set, marked point process, conjugate prior, Bayesian estimation, target tracking.**

I. Introduction

Multi-target filtering involves the on-line estimation of an unknown and time-varying number of targets and their individual states from a sequence of observations [2]–[4]. While the term multi-target filtering is often used interchangeably with multi-target tracking, there is a subtle difference. In multi-target tracking we are also interested in the trajectories of the targets (indeed, real multi-target tracking systems require track labels). This work is concerned with a Bayesian multi-target filtering solution that also provides estimates of target trajectories, hence the name multi-target tracking filter.

The key challenges in multi-target filtering/tracking include *detection uncertainty*, *clutter*, and *data association uncertainty*. To date, three major approaches to multi-target tracking/filtering have emerged as the main solution paradigms. These are, Multiple Hypotheses Tracking (MHT), [2], [5]–[7], Joint Probabilistic Data Association (JPDA) [2], [4], and Random Finite Set (RFS) [3].

The random finite set (RFS) approach provides an elegant Bayesian formulation of the multi-target filtering/tracking problem in which the collection of target states, referred to as the multi-target state, is treated as a finite set [3], [8]. The rationale behind this representation traces back to a fundamental consideration in estimation theory–estimation error [9]. This mathematical framework subsequently became a very popular multi-target estimation method with applications in sonar [10], computer vision [11], [12], [13], [14], field robotics [15], [16], [17], [18], [19] traffic monitoring [20], [21], [22], cell biology [23], [13], [24], sensor network and distributed estimation [25], [26], [27], [28] etc.

The centerpiece of the RFS approach is the *Bayes multi-target filter* [3], which recursively propagates the filtering density of the multi-target state forward in time. This filter is also a (multi-target) tracker when target identities or labels are incorporated into individual target states. Due to the numerical complexity of Bayes multi-target filter, the Probability Hypothesis Density (PHD) [8], Cardinalized PHD (CPHD) [29], and multi-Bernoulli filters [30], [9] have been developed as approximations. These filters, in principle, are not multi-target trackers because they rest on the premise that targets are indistinguishable.

In [1], [31], the notion of *labeled RFSs* is introduced to address target trajectories and their uniqueness. The key results include conjugate priors that are closed under the Chapman-Kolmogorov equation, and an analytic solution to the Bayes multi-target tracking filter known as the δ-generalized labeled multi-Bernoulli (δ-GLMB) filter. Although a simulation result was presented to verify the solution, specific implementation details were not given.

As a sequel to [1], the aim of this paper is to complement its theoretical contributions with practical algorithms that will facilitate the development of applications in signal processing and related fields. In particular, we present an efficient and highly parallelizable implementation of the δ-GLMB filter. Each iteration of the δ-GLMB filter involves multi-target filtering and prediction densities that are weighted sums of multi-target exponentials. While these sums are expressible in closed forms, the number of terms grows super-exponentially in time. Furthermore, it is not tractable to exhaustively compute all the terms of the multi-target densities first and then truncate by discarding those deemed insignificant.

The key innovation is the truncation of the multi-target densities without exhaustively computing all their components. The multi-target filtering and prediction densities are truncated using the ranked assignment and K-shortest paths algorithms, respectively. Techniques such as PHD filtering are used as inexpensive look-ahead strategies to drastically reduce the number of calls to ranked assignment and K-shortest paths algorithms. Moreover, we establish that truncation by discard-

Acknowledgement: This work is supported by the Australian Research Council under the Future Fellowship FT0991854 and Discovery Early Career Researcher Award DE120102388

B.-N. Vo and B.-T. Vo are with the Department of Electrical and Computer Engineering, Curtin University, Bentley, WA 6102, Australia (email: {ba-ngu,ba-tuong}.vo@curtin.edu.au). D. Phung is with the Faculty of Science, Engineering & Built Environment, Deakin University, Geelong, VIC 3220, Australia (email: dinh.phung@deakin.edu.au)

ing δ-GLMB components with small weights minimizes the L_1 error in the multi-target density. To our best knowledge, this is the first result regarding the effect of truncation on the multi-target probability law.

The paper is organized as follows. Background on labeled RFS and the δ-GLMB filter is provided in section II. Section III establishes the L_1-distance between a δ-GLMB density and its truncated version. Sections IV and V present efficient implementations δ-GLMB filter update and prediction respectively. Section VI details the multi-target state estimation, and discusses look-ahead strategies to reduce the computational load. Numerical results are presented in Section VII and concluding remarks are given in Section VIII.

II. Background

This section summarizes the labeled RFS formulation of the multi-target tracking problem and the δ-GLMB filter proposed in [1]. Labeled RFS is summarized in subsection II-A, followed by Bayes multi-target filtering basics in subsections II-B, II-C and II-D. The δ-GLMB multi-target density and recursion are summarized in subsections II-E and II-F.

Throughout the paper, we use the standard inner product notation $\langle f,g\rangle \triangleq \int f(x)g(x)dx$, and the following multi-object exponential notation $h^X \triangleq \prod_{x\in X} h(x)$, where h is a real-valued function, with $h^{\emptyset}=1$ by convention. We denote a generalization of the Kroneker delta that takes arbitrary arguments such as sets, vectors, etc., by

$$\delta_Y(X) \triangleq \begin{cases} 1, & \text{if } X=Y \\ 0, & \text{otherwise} \end{cases},$$

and the inclusion function, a generalization of the indicator function, by

$$1_Y(X) \triangleq \begin{cases} 1, & \text{if } X\subseteq Y \\ 0, & \text{otherwise} \end{cases}.$$

We also write $1_Y(x)$ in place of $1_Y(\{x\})$ when $X=\{x\}$.

A. Labeled RFS

An RFS is simply a finite-set-valued random variable [32], [33]. In this paper we use the Finite Set Statistics (FISST) notion of integration/density to characterize RFSs [3], [8]. Treatments of RFS in the context of multi-target filtering can be found in [3], [34], [35].

To incorporate target identity, each state $x\in\mathbb{X}$ is augmented with a unique label $\ell\in\mathbb{L}=\{\alpha_i : i\in\mathbb{N}\}$, where $\mathbb{N}$ denotes the set of positive integers and the α_i's are distinct.

Let $\mathcal{L}:\mathbb{X}\times\mathbb{L}\to\mathbb{L}$ be the projection $\mathcal{L}((x,\ell))=\ell$, then a finite subset $\mathbf{X}$ of $\mathbb{X}\times\mathbb{L}$ has distinct labels if and only if $\mathbf{X}$ and its labels $\mathcal{L}(\mathbf{X})=\{\mathcal{L}(\mathbf{x}):\mathbf{x}\in\mathbf{X}\}$ have the same cardinality, i.e. $\delta_{|\mathbf{X}|}(|\mathcal{L}(\mathbf{X})|)=1$. The function $\Delta(\mathbf{X})\triangleq\delta_{|\mathbf{X}|}(|\mathcal{L}(\mathbf{X})|)$ is called the *distinct label indicator*.

Definition 1 *A labeled RFS with state space $\mathbb{X}$ and (discrete) label space $\mathbb{L}$ is an RFS on $\mathbb{X}\times\mathbb{L}$ such that each realization has distinct labels.*

The unlabeled version of a labeled RFS is obtained by simply discarding the labels. Consequently, the cardinality distribution (the distribution of the number of objects) of a labeled RFS is the same as its unlabeled version.

For the rest of the paper, single-object states are represented by lowercase letters (e.g. x, $\mathbf{x}$), while multi-object states are represented by uppercase letters (e.g. X, $\mathbf{X}$), symbols for labeled states and their distributions are bolded to distinguish them from unlabeled ones (e.g. $\mathbf{x}$, $\mathbf{X}$, $\boldsymbol{\pi}$, etc.), spaces are represented by blackboard bold (e.g. $\mathbb{X}$, $\mathbb{Z}$, $\mathbb{L}$, $\mathbb{N}$, etc.), and the class of finite subsets of a space $\mathbb{X}$ is denoted by $\mathcal{F}(\mathbb{X})$. The integral of a function $\mathbf{f}:\mathbb{X}\times\mathbb{L}\to\mathbb{R}$ is given by

$$\int \mathbf{f}(\mathbf{x})d\mathbf{x}=\sum_{\ell\in\mathbb{L}}\int_{\mathbb{X}}\mathbf{f}((x,\ell))dx.$$

B. Bayesian Multi-target Filtering

To incorporate target tracks in the Bayes multi-target filtering framework, targets are identified by an ordered pair of integers $\ell=(k,i)$, where k is the time of birth, and $i\in\mathbb{N}$ is a unique index to distinguish objects born at the same time. Figure 1 illustrates the assignment of labels to target trajectories. The label space for objects born at time k, denoted as $\mathbb{L}_k$, is then $\{k\}\times\mathbb{N}$. An object born at time k, has state $\mathbf{x}\in\mathbb{X}\times\mathbb{L}_k$. The label space for targets at time k (including those born prior to k), denoted as $\mathbb{L}_{0:k}$, is constructed recursively by $\mathbb{L}_{0:k}=\mathbb{L}_{0:k-1}\cup\mathbb{L}_k$. A multi-object state $\mathbf{X}$ at time k, is a finite subset of $\mathbb{X}\times\mathbb{L}_{0:k}$. Note that $\mathbb{L}_{0:k-1}$ and $\mathbb{L}_k$ are disjoint.

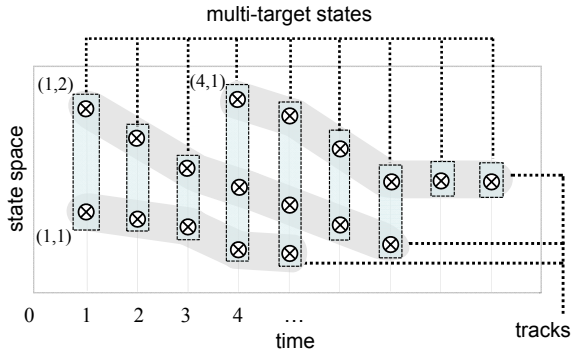

Fig. 1. An example of label assignments. The two tracks born at time 1 are given labels (1,1) and (1,2), while the only track born at time 4 is given label (4,1). Notice also the difference between multi-target states and target tracks.

Suppose that at time k, there are $N(k)$ target states $\mathbf{x}_{k,1},\ldots,\mathbf{x}_{k,N(k)}$, each taking values in the (labeled) state space $\mathbb{X}\times\mathbb{L}$, and $M(k)$ measurements $z_{k,1},\ldots,z_{k,M(k)}$ each taking values in an observation space $\mathbb{Z}$. In the random finite set formulation, the set of targets and observations, at time k, [3], [8] are treated as the *multi-target state* and *multi-target observation*, respectively

$$\mathbf{X}_k=\{\mathbf{x}_{k,1},\ldots,\mathbf{x}_{k,N(k)}\},$$
$$Z_k=\{z_{k,1},\ldots,z_{k,M(k)}\}.$$

The *multi-target posterior density* captures all information on the set of target trajectories conditioned on the measurement history $Z_{0:k}=(Z_0,...,Z_k)$, and is given recursively for $k\geq 1$ by

$$\boldsymbol{\pi}_{0:k}(\mathbf{X}_{0:k}|Z_{0:k}) \propto g_k(Z_k|\mathbf{X}_k)\mathbf{f}_{k|k-1}(\mathbf{X}_k|\mathbf{X}_{k-1})\boldsymbol{\pi}_{0:k-1}(\mathbf{X}_{0:k-1}|Z_{0:k-1}),$$

where $\mathbf{X}_{0:k} = (\mathbf{X}_0, ..., \mathbf{X}_k)$, $g_k(\cdot|\cdot)$ is the *multi-target likelihood function* at time k, $\mathbf{f}_{k|k-1}(\cdot|\cdot)$ is the *multi-target transition density* to time k. The multi-target likelihood function encapsulates the underlying models for detections and false alarms while the multi-target transition density encapsulates the underlying models of target motions, births and deaths.

Multi-target filtering is concerned with the marginal of the multi-target posterior density, at the current time. Let $\boldsymbol{\pi}_k(\cdot|Z_k)$ denote the *multi-target filtering density* at time k, and $\boldsymbol{\pi}_{k+1|k}$ denote the *multi-target prediction density* to time $k+1$ (formally $\boldsymbol{\pi}_k$ and $\boldsymbol{\pi}_{k+1|k}$ should be written respectively as $\boldsymbol{\pi}_k(\cdot|Z_{0:k})$, and $\boldsymbol{\pi}_{k+1|k}(\cdot|Z_{0:k})$, but for simplicity we omit the dependence on past measurements). Then, the *multi-target Bayes filter* propagates $\boldsymbol{\pi}_k$ in time [3], [8] according to the following update and prediction

$$\boldsymbol{\pi}_k(\mathbf{X}_k|Z_k) = \frac{g_k(Z_k|\mathbf{X}_k)\boldsymbol{\pi}_{k|k-1}(\mathbf{X}_k)}{\int g_k(Z_k|\mathbf{X})\boldsymbol{\pi}_{k|k-1}(\mathbf{X})\delta\mathbf{X}}, \quad (1)$$

$$\boldsymbol{\pi}_{k+1|k}(\mathbf{X}_{k+1}) = \int \mathbf{f}_{k+1|k}(\mathbf{X}_{k+1}|\mathbf{X}_k)\boldsymbol{\pi}_k(\mathbf{X}_k|Z_k)\delta\mathbf{X}_k, \quad (2)$$

where the integral is a *set integral* defined for any function $\mathbf{f} : \mathcal{F}(\mathbb{X}\times\mathbb{L}) \to \mathbb{R}$ by

$$\int \mathbf{f}(\mathbf{X})\delta\mathbf{X} = \sum_{i=0}^{\infty} \frac{1}{i!} \int \mathbf{f}(\{\mathbf{x}_1, ..., \mathbf{x}_i\})d(\mathbf{x}_1, ..., \mathbf{x}_i).$$

The multi-target filtering density captures all information on the multi-target state, such as the number of targets and their states, at the current time.

For convenience, in what follows we omit explicit references to the time index k, and denote $\mathbb{L} \triangleq \mathbb{L}_{0:k}$, $\mathbb{B} \triangleq \mathbb{L}_{k+1}$, $\mathbb{L}_+ \triangleq \mathbb{L} \cup \mathbb{B}$, $\boldsymbol{\pi} \triangleq \boldsymbol{\pi}_k$, $\boldsymbol{\pi}_+ \triangleq \boldsymbol{\pi}_{k+1|k}$, $g \triangleq g_k$, $\mathbf{f} \triangleq \mathbf{f}_{k+1|k}$.

C. Measurement likelihood function

For a given multi-target state $\mathbf{X}$, at time k, each state $(x, \ell) \in \mathbf{X}$ is either detected with probability $p_D(x, \ell)$ and generates a point z with likelihood $g(z|x, \ell)$, or missed with probability $1 - p_D(x, \ell)$. The multi-object observation $Z = \{z_1, ..., z_{|Z|}\}$ is the superposition of the detected points and Poisson clutter with intensity function κ.

Definition 2 *An association map (for the current time) is a function $\theta : \mathbb{L} \to \{0, 1, ..., |Z|\}$ such that $\theta(i) = \theta(i') > 0$ implies $i = i'$. The set Θ of all such association maps is called the association map space. The subset of association maps with domain I is denoted by $\Theta(I)$.*

An association map describes which tracks generated which measurements, i.e. track ℓ generates measurement $z_{\theta(\ell)} \in Z$, with undetected tracks assigned to 0. The condition $\theta(i) = \theta(i') > 0$ implies $i = i'$, means that a track can generate at most one measurement at any point in time.

Assuming that, conditional on $\mathbf{X}$, detections are independent, and that clutter is independent of the detections, the multi-object likelihood is given by

$$g(Z|\mathbf{X}) = e^{-\langle\kappa,1\rangle}\kappa^Z \sum_{\theta\in\Theta(\mathcal{L}(\mathbf{X}))} [\psi_Z(\cdot;\theta)]^{\mathbf{X}} \quad (3)$$

where

$$\psi_Z(x, \ell; \theta) = \begin{cases} \frac{p_D(x,\ell)g(z_{\theta(\ell)}|x,\ell)}{\kappa(z_{\theta(\ell)})}, & \text{if } \theta(\ell) > 0 \\ 1 - p_D(x, \ell), & \text{if } \theta(\ell) = 0 \end{cases} \quad (4)$$

Equation (3) is equivalent to the likelihood function given by (54) in [1], and is more convenient for implementation.

D. Multi-target transition kernel

Given the current multi-object state $\mathbf{X}$, each state $(x, \ell) \in \mathbf{X}$ either continues to exist at the next time step with probability $p_S(x, \ell)$ and evolves to a new state (x_+, ℓ_+) with probability density $f(x_+|x, \ell)\delta_\ell(\ell_+)$, or dies with probability $1-p_S(x, \ell)$. The set of new targets born at the next time step is distributed according to

$$\mathbf{f}_B(\mathbf{Y}) = \Delta(\mathbf{Y})w_B(\mathcal{L}(\mathbf{Y}))\,[p_B]^{\mathbf{Y}} \quad (5)$$

where w_B and p_B are given parameters of the multi-target birth density $\mathbf{f}_B$, defined on $\mathbb{X}\times\mathbb{B}$. Note that $\mathbf{f}_B(\mathbf{Y}) = 0$ if $\mathbf{Y}$ contains any element $\mathbf{y}$ with $\mathcal{L}(\mathbf{y}) \notin \mathbb{B}$. The birth model (5) covers both labeled Poisson and labeled multi-Bernoulli [1].

The multi-target state at the next time $\mathbf{X}_+$ is the superposition of surviving targets and new born targets. Assuming that targets evolve independently of each other and that births are independent of surviving targets, it was shown in [1] that the multi-target transition kernel is given by

$$\mathbf{f}(\mathbf{X}_+|\mathbf{X}) = \mathbf{f}_S(\mathbf{X}_+ \cap (\mathbb{X}\times\mathbb{L})|\mathbf{X})\mathbf{f}_B(\mathbf{X}_+ - (\mathbb{X}\times\mathbb{L})) \quad (6)$$

where

$$\mathbf{f}_S(\mathbf{W}|\mathbf{X}) = \Delta(\mathbf{W})\Delta(\mathbf{X})1_{\mathcal{L}(\mathbf{X})}(\mathcal{L}(\mathbf{W}))\,[\Phi(\mathbf{W};\cdot)]^{\mathbf{X}} \quad (7)$$

$$\Phi(\mathbf{W}; x, \ell) = \begin{cases} p_S(x, \ell)f(x_+|x, \ell), & \text{if } (x_+, \ell) \in \mathbf{W} \\ 1 - p_S(x, \ell), & \text{if } \ell \notin \mathcal{L}(\mathbf{W}) \end{cases}. \quad (8)$$

E. Delta-Generalized Labeled Multi-Bernoulli

The δ-generalized labeled multi-Bernoulli filter is a solution to the Bayes multi-target filter based on the family of generalized labeled multi-Bernoulli (GLMB) distributions

$$\boldsymbol{\pi}(\mathbf{X}) = \Delta(\mathbf{X}) \sum_{\xi\in\Xi} w^{(\xi)}(\mathcal{L}(\mathbf{X})) \left[p^{(\xi)}\right]^{\mathbf{X}},$$

where Ξ is a discrete space, each $p^{(\xi)}(\cdot, \ell)$ is a probability density, and each $w^{(\xi)}(I)$ is non-negative with $\sum_{(I,\xi)\in\mathcal{F}(\mathbb{L})\times\Xi} w^{(\xi)}(I) = 1$. A GLMB can be interpreted as a mixture of multi-target exponentials [1]. While this family is closed under the Bayes recursion [1], it is not clear how numerical implementation can be accomplished. Fortunately, an alternative form of the GLMB, known as δ-GLMB

$$\boldsymbol{\pi}(\mathbf{X}) = \Delta(\mathbf{X}) \sum_{(I,\xi)\in\mathcal{F}(\mathbb{L})\times\Xi} \omega^{(I,\xi)}\delta_I(\mathcal{L}(\mathbf{X})) \left[p^{(\xi)}\right]^{\mathbf{X}}, \quad (9)$$

where $\omega^{(I,\xi)} = w^{(\xi)}(I)$, provides a representation that facilitates numerical implementation. Note that the δ-GLMB can be obtained from the GLMB by using the identity $w^{(\xi)}(J) = \sum_{I\in\mathcal{F}(\mathbb{L})} w^{(\xi)}(I)\delta_I(J)$, since the summand is non-zero if and only if $I = J$.

In the δ-GLMB initial multi-target prior

$$\boldsymbol{\pi}_0(\mathbf{X}) = \Delta(\mathbf{X}) \sum_{I\in\mathcal{F}(\mathbb{L}_0)} \omega_0^{(I)}\delta_I(\mathcal{L}(\mathbf{X}))p_0^{\mathbf{X}}, \quad (10)$$

each $I \in \mathcal{F}(\mathbb{L}_0)$ represents a set of tracks labels born at time 0, $\omega_0^{(I)}$ represents the weight of the hypothesis that I is the set of track labels at time 0, and $p_0(\cdot,\ell)$ is the probability density of the kinematic state of track $\ell \in I$. For example, suppose that there are 2 possibilities:

1) 0.3 chance of 1 target with label $(0,2)$, and density $p_0(\cdot,(0,2)) = \mathcal{N}(\cdot;m,P_2)$,
2) 0.7 chance of 2 targets with labels $(0,1)$, $(0,2)$ and respective densities $p_0(\cdot,(0,1)) = \mathcal{N}(\cdot;0,P_1)$, $p_0(\cdot,(0,2)) = \mathcal{N}(\cdot;m,P_2)$.

Then the δ-GLMB representation is

$$\boldsymbol{\pi}_0(\mathbf{X}) = 0.3\delta_{\{(0,2)\}}(\mathcal{L}(\mathbf{X}))p_0^{\mathbf{X}} + 0.7\delta_{\{(0,1),(0,2)\}}(\mathcal{L}(\mathbf{X}))p_0^{\mathbf{X}}.$$

Note that the initial prior (10) is a δ-GLMB with $\Xi = \emptyset$. For δ-GLMB filtering and prediction densities that are conditioned on measurements up to time k, the discrete space Ξ is the space of association map histories $\Theta_{0:k} \triangleq \Theta_0 \times ... \times \Theta_k$, where Θ_t denotes the association map space at time t. In particular, as shown in [1], for each $k \geq 0$ the filtering and prediction densities are δ-GLMB densities:

$$\boldsymbol{\pi}_k(\mathbf{X}|Z_k) = \Delta(\mathbf{X}) \sum_{(I,\xi)\in\mathcal{F}(\mathbb{L}_{0:k})\times\Theta_{0:k}} \omega_k^{(I,\xi)}\delta_I(\mathcal{L}(\mathbf{X}))\left[p_k^{(\xi)}(\cdot|Z_k)\right]^{\mathbf{X}} \quad (11)$$

$$\boldsymbol{\pi}_{k+1|k}(\mathbf{X}) = \Delta(\mathbf{X}) \sum_{(I,\xi)\in\mathcal{F}(\mathbb{L}_{0:k+1})\times\Theta_{0:k}} \omega_{k+1|k}^{(I,\xi)}\delta_I(\mathcal{L}(\mathbf{X}))\left[p_{k+1|k}^{(\xi)}\right]^{\mathbf{X}} \quad (12)$$

Each $I \in \mathcal{F}(\mathbb{L}_{0:k})$ represents a set of track labels at time k, and each $\xi = (\theta_0,...,\theta_k) \in \Theta_{0:k}$ represents a history of association maps up to time k, which also contains the history of target labels encapsulating both births and deaths. The pair $(I,\xi) \in \mathcal{F}(\mathbb{L}_{0:k})\times\Theta_{0:k}$ is called a *hypothesis*, and its associated weight $\omega_k^{(I,\xi)}$ can be interpreted as the probability of the hypothesis. Similarly the pair $(I,\xi) \in \mathcal{F}(\mathbb{L}_{0:k+1})\times\Theta_{0:k}$ is called a *prediction hypothesis*, with probability $\omega_{k+1|k}^{(I,\xi)}$. The densities $p_k^{(\xi)}(\cdot,\ell)$ and $p_{k+1|k}^{(\xi)}(\cdot,\ell)$ are the filtering and prediction densities of the kinematic state of track ℓ for association map history ξ.

F. Delta Generalized Labeled Multi-Bernoulli Recursion

The δ-GLMB filter recursively propagates a δ-GLMB filtering density forward in time via the Bayes update and prediction equations (2) and (1). Closed form solutions to the update and prediction of the δ-GLMB filter are given by the following results [1].

Proposition 3 *If the current multi-target prediction density is a δ-GLMB of the form (9), then the multi-target filtering density is a δ-GLMB given by*

$$\boldsymbol{\pi}(\mathbf{X}|Z) = \Delta(\mathbf{X}) \sum_{(I,\xi)\in\mathcal{F}(\mathbb{L})\times\Xi} \sum_{\theta\in\Theta(I)} \omega^{(I,\xi,\theta)}(Z)\delta_I(\mathcal{L}(\mathbf{X}))\left[p^{(\xi,\theta)}(\cdot|Z)\right]^{\mathbf{X}} \quad (13)$$

where $\Theta(I)$ denotes the subset of current association maps with domain I,

$$\omega^{(I,\xi,\theta)}(Z) \propto \omega^{(I,\xi)}[\eta_Z^{(\xi,\theta)}]^I, \quad (14)$$

$$\eta_Z^{(\xi,\theta)}(\ell) = \left\langle p^{(\xi)}(\cdot,\ell), \psi_Z(\cdot,\ell;\theta)\right\rangle, \quad (15)$$

$$p^{(\xi,\theta)}(x,\ell|Z) = \frac{p^{(\xi)}(x,\ell)\psi_Z(x,\ell;\theta)}{\eta_Z^{(\xi,\theta)}(\ell)}, \quad (16)$$

Proposition 4 *If the current multi-target filtering density is a δ-GLMB of the form (9), then the multi-target prediction to the next time is a δ-GLMB given by*

$$\boldsymbol{\pi}_+(\mathbf{X}_+) = \Delta(\mathbf{X}_+) \sum_{(I_+,\xi)\in\mathcal{F}(\mathbb{L}_+)\times\Xi} \omega_+^{(I_+,\xi)}\delta_{I_+}(\mathcal{L}(\mathbf{X}_+))\left[p_+^{(\xi)}\right]^{\mathbf{X}_+} \quad (17)$$

where

$$\omega_+^{(I_+,\xi)} = \omega_S^{(\xi)}(I_+\cap\mathbb{L})w_B(I_+\cap\mathbb{B}) \quad (18)$$

$$\omega_S^{(\xi)}(L) = [\eta_S^{(\xi)}]^L \sum_{I\supseteq L}[1-\eta_S^{(\xi)}]^{I-L}\omega^{(I,\xi)} \quad (19)$$

$$\eta_S^{(\xi)}(\ell) = \left\langle p_S(\cdot,\ell), p^{(\xi)}(\cdot,\ell)\right\rangle \quad (20)$$

$$p_+^{(\xi)}(x,\ell) = 1_{\mathbb{L}}(\ell)p_S^{(\xi)}(x,\ell) + 1_{\mathbb{B}}(\ell)p_B(x,\ell) \quad (21)$$

$$p_S^{(\xi)}(x,\ell) = \frac{\left\langle p_S(\cdot,\ell)f(x|\cdot,\ell), p^{(\xi)}(\cdot,\ell)\right\rangle}{\eta_S^{(\xi)}(\ell)} \quad (22)$$

Note from Propositions 3 and 4 that the actual value of the association history ξ is not used in the calculations, it is used merely as an indexing variable. On the other hand, the value of the label set I is used in the calculations.

III. Characterizing Truncation Error

A δ-GLMB is completely characterized by the set of parameters $\{(\omega^{(I,\xi)},p^{(\xi)}) : (I,\xi) \in \mathcal{F}(\mathbb{L})\times\Xi\}$. For implementation it is convenient to consider the set of δ-GLMB parameters as an enumeration of all hypotheses (with positive weight) together with their associated weights and track densities $\{(I^{(h)},\xi^{(h)},\omega^{(h)},p^{(h)})\}_{h=1}^H$, as shown in Figure 2, where $\omega^{(h)} \triangleq \omega^{(I^{(h)},\xi^{(h)})}$ and $p^{(h)} \triangleq p^{(\xi^{(h)})}$. Implementing the δ-GLMB filter then amounts to recursively propagating the set of δ-GLMB parameters forward in time.

$I^{(1)} = \{\ell_1^{(1)},\ldots,\ell_{\lvert I^{(1)}\rvert}^{(1)}\}$	$I^{(2)} = \{\ell_1^{(2)},\ldots,\ell_{\lvert I^{(2)}\rvert}^{(2)}\}$	…	$I^{(h)} = \{\ell_1^{(h)},\ldots,\ell_{\lvert I^{(h)}\rvert}^{(h)}\}$	…
$\xi^{(1)}$	$\xi^{(2)}$	…	$\xi^{(h)}$	…
$\omega^{(1)}$	$\omega^{(2)}$	…	$\omega^{(h)}$	…
$p^{(1)}(\cdot,\ell_1^{(1)})$ ⋮ $p^{(1)}(\cdot,\ell_{\lvert I^{(1)}\rvert}^{(1)})$	$p^{(2)}(\cdot,\ell_1^{(2)})$ ⋮ $p^{(2)}(\cdot,\ell_{\lvert I^{(2)}\rvert}^{(2)})$	…	$p^{(h)}(\cdot,\ell_1^{(h)})$ ⋮ $p^{(h)}(\cdot,\ell_{\lvert I^{(h)}\rvert}^{(h)})$	…

Fig. 2. An enumeration of a δ-GLMB parameter set with each component indexed by an integer h. The hypothesis for component h is $(I^{(h)},\xi^{(h)})$ while its weight and associated track densities are $\omega^{(h)}$ and $p^{(h)}(\cdot,\ell)$, $\ell \in I^{(h)}$.

Since the number of hypotheses grows super-exponentially with time, it is necessary to reduce the number of components in the δ-GLMB parameter set, at each time step. A simple solution is to truncate the δ-GLMB density by discarding "insignificant" hypotheses.

The effect of truncation on the probability law of the multi-target state has not been characterized even though this truncation is widely used in MHT and JPDA. In the RFS approach the probability law of the multi-target state is

completely captured in the multi-target density. Consequently, the effect of discarding hypotheses in a δ-GLMB density can be characterized by the difference/dissimilarity between the untruncated and truncated δ-GLMB densities. The following result establishes the L_1-error between a δ-GLMB density and its truncated version.

Proposition 5 *Let* $\|\mathbf{f}\|_1 \triangleq \int |\mathbf{f}(\mathbf{X})|\,\delta\mathbf{X}$ *denote the* L_1*-norm of* $\mathbf{f}:\mathcal{F}(\mathbb{X}\times\mathbb{L})\to\mathbb{R}$*, and for a given* $\mathbb{H}\subseteq\mathcal{F}(\mathbb{L})\times\Xi$ *let*

$$\mathbf{f}_{\mathbb{H}}(\mathbf{X}) = \Delta(\mathbf{X})\sum_{(I,\xi)\in\mathbb{H}}\omega^{(I,\xi)}\delta_I(\mathcal{L}(\mathbf{X}))\left[p^{(\xi)}\right]^{\mathbf{X}}$$

be an unnormalized δ*-GLMB density (i.e. does not necessarily integrate to 1). If* $\mathbb{T}\subseteq\mathbb{H}$ *then*

$$\begin{aligned}||\mathbf{f}_{\mathbb{H}}-\mathbf{f}_{\mathbb{T}}||_1 &= \sum_{(I,\xi)\in\mathbb{H}-\mathbb{T}}\omega^{(I,\xi)},\\ \left\|\frac{\mathbf{f}_{\mathbb{H}}}{||\mathbf{f}_{\mathbb{H}}||_1}-\frac{\mathbf{f}_{\mathbb{T}}}{||\mathbf{f}_{\mathbb{T}}||_1}\right\|_1 &\leq 2\frac{||\mathbf{f}_{\mathbb{H}}||_1-||\mathbf{f}_{\mathbb{T}}||_1}{||\mathbf{f}_{\mathbb{H}}||_1},\end{aligned}$$

Proof:

$$\begin{aligned}&\|\mathbf{f}_{\mathbb{H}}-\mathbf{f}_{\mathbb{T}}\|_1\\ &= \int\left|\Delta(\mathbf{X})\sum_{(I,\xi)\in\mathbb{H}-\mathbb{T}}\omega^{(I,\xi)}\delta_I(\mathcal{L}(\mathbf{X}))\left[p^{(\xi)}\right]^{\mathbf{X}}\right|\delta\mathbf{X}\\ &= \sum_{(I,\xi)\in\mathbb{H}-\mathbb{T}}\omega^{(I,\xi)}\int\Delta(\mathbf{X})\delta_I(\mathcal{L}(\mathbf{X}))\left[p^{(\xi)}\right]^{\mathbf{X}}\delta\mathbf{X}\\ &= \sum_{(I,\xi)\in\mathbb{H}-\mathbb{T}}\omega^{(I,\xi)}\sum_{L\subseteq\mathbb{L}}\delta_I(L), \qquad \text{(by Lemma 3 of [1])}\\ &= \sum_{(I,\xi)\in\mathbb{H}-\mathbb{T}}\omega^{(I,\xi)}.\end{aligned}$$

Now note that $\|\mathbf{f}_{\mathbb{H}}-\mathbf{f}_{\mathbb{T}}\|_1 = ||\mathbf{f}_{\mathbb{H}}||_1-||\mathbf{f}_{\mathbb{T}}||_1$, moreover,

$$\begin{aligned}&\left\|\frac{\mathbf{f}_{\mathbb{H}}}{||\mathbf{f}_{\mathbb{H}}||_1}-\frac{\mathbf{f}_{\mathbb{T}}}{||\mathbf{f}_{\mathbb{T}}||_1}\right\|_1\\ &\leq \int\left|\Delta(\mathbf{X})\sum_{(I,\xi)\in\mathbb{T}}\left(\frac{\omega^{(I,\xi)}}{||\mathbf{f}_{\mathbb{H}}||_1}-\frac{\omega^{(I,\xi)}}{||\mathbf{f}_{\mathbb{T}}||_1}\right)\delta_I(\mathcal{L}(\mathbf{X}))\left[p^{(\xi)}\right]^{\mathbf{X}}\right|\delta\mathbf{X}\\ &\quad+\int\left|\Delta(\mathbf{X})\sum_{(I,\xi)\in\mathbb{H}-\mathbb{T}}\frac{\omega^{(I,\xi)}}{||\mathbf{f}_{\mathbb{H}}||_1}\delta_I(\mathcal{L}(\mathbf{X}))\left[p^{(\xi)}\right]^{\mathbf{X}}\right|\delta\mathbf{X}\\ &= \sum_{(I,\xi)\in\mathbb{T}}\left|\frac{\omega^{(I,\xi)}}{||\mathbf{f}_{\mathbb{H}}||_1}-\frac{\omega^{(I,\xi)}}{||\mathbf{f}_{\mathbb{T}}||_1}\right|+\sum_{(I,\xi)\in\mathbb{H}-\mathbb{T}}\frac{\omega^{(I,\xi)}}{||\mathbf{f}_{\mathbb{H}}||_1}\\ &= 1-\frac{||\mathbf{f}_{\mathbb{T}}||_1}{||\mathbf{f}_{\mathbb{H}}||_1}+\frac{||\mathbf{f}_{\mathbb{H}}||_1-||\mathbf{f}_{\mathbb{T}}||_1}{||\mathbf{f}_{\mathbb{H}}||_1}\\ &= 2\frac{||\mathbf{f}_{\mathbb{H}}||_1-||\mathbf{f}_{\mathbb{T}}||_1}{||\mathbf{f}_{\mathbb{H}}||_1}.\square\end{aligned}$$

It follows from the above result that the intuitive strategy of keeping δ-GLMB components with high weights and discarding those with the smallest weights minimizes the L_1-error in the truncated multi-target density.

In the δ-GLMB recursion, it is not tractable to exhaustively compute all the components first and then discard those with small weights. The trick is to perform the truncation without having to propagate all the components.

IV. Delta-GLMB Update

This section presents a tractable implementation of the δ-GLMB update by truncating the multi-target filtering density without computing all the hypotheses and their weights, via the ranked assignment algorithm. Subsection IV-A summarizes the ranked assignment problem in the context of truncating the δ-GLMB filtering density. Subsection IV-B details the computation of the updated δ-GLMB parameters and subsection IV-C presents the δ-GLMB update algorithm.

A. Ranked Assignment Problem

Note from the δ-GLMB weight update (14) that each hypothesis (I,ξ) with weight $\omega^{(I,\xi)}$ generates a new set of hypotheses $(I,(\xi,\theta))$, $\theta\in\Theta(I)$, with weights $\omega^{(I,\xi,\theta)}(Z)\propto\omega^{(I,\xi)}[\eta_Z^{(\xi,\theta)}]^I$. For a given hypothesis (I,ξ), if we can generate the association maps $\theta\in\Theta(I)$ in decreasing order of $[\eta_Z^{(\xi,\theta)}]^I$, then the highest weighted components can be selected without exhaustively computing all the new hypothesis and their weights. This can be accomplished by solving the following ranked assignment problem.

Enumerating $Z=\{z_1,...,z_{|Z|}\}$, $I=\{\ell_1,...,\ell_{|I|}\}$, each association map $\theta\in\Theta(I)$ can be represented by an $|I|\times|Z|$ *assignment matrix* S consisting of 0 or 1 entries with every row and column summing to either 1 or 0. For $i\in\{1,...,|I|\}$, $j\in\{1,...,|Z|\}$, $S_{i,j}=1$ if and only if the jth measurement is assigned to track ℓ_i, i.e. $\theta(\ell_i)=j$. An all-zero row i means that track ℓ_i is misdetected while an all-zero column j means that measurement z_j is a false alarm. Conversion from S to θ is given by $\theta(\ell_i)=\sum_{j=1}^{|Z|}j\delta_1(S_{i,j})$.

The *cost matrix* of an optimal assignment problem is the $|I|\times|Z|$ matrix:

$$C_Z^{(I,\xi)} = \begin{bmatrix} C_{1,1} & \cdots & C_{1,|Z|}\\ \vdots & \ddots & \vdots\\ C_{|I|,1} & \cdots & C_{|I|,|Z|}\end{bmatrix} \tag{23}$$

where for $i\in\{1,...,|I|\}$, $j\in\{1,...,|Z|\}$

$$C_{i,j} = -\ln\left(\frac{\left\langle p^{(\xi)}(\cdot,\ell_i),p_D(\cdot,\ell_i)g(z_j|\cdot,\ell_i)\right\rangle}{\left\langle p^{(\xi)}(\cdot,\ell_i),1-p_D(\cdot,\ell_i)\right\rangle\kappa(z_j)}\right) \tag{24}$$

is the cost of the assigning the jth measurement to track ℓ_i (Subsection IV-B details the numerical computation of $C_{i,j}$).

The cost of an assignment (matrix) S is the combined costs of every measurement to target assignments, which can be succinctly written as the Frobenius inner product

$$\text{tr}(S^TC_Z^{(I,\xi)}) = \sum_{i=1}^{|I|}\sum_{j=1}^{|Z|}C_{i,j}S_{i,j}.$$

Substituting (4) into equation (15), it follows that the cost of S (and the corresponding association map θ) is related to the filtered hypothesis weight $\omega^{(I,\xi,\theta)}(Z)\propto\omega^{(I,\xi)}[\eta_Z^{(\xi,\theta)}]^I$ by

$$\left[\eta_Z^{(\xi,\theta)}\right]^I = \exp\left(-\text{tr}(S^TC_Z^{(I,\xi)})\right)\prod_{\ell\in I}\left\langle p^{(\xi)}(\cdot,\ell),1-p_D(\cdot,\ell)\right\rangle.$$

The optimal assignment problem seeks an assignment matrix S^* (and corresponding association map θ^*) that minimizes the cost $\text{tr}(S^{*T}C_Z^{(I,\xi)})$ [36]. The ranked assignment problem

seeks an enumeration of the least cost assignment matrices in non-decreasing order [37]. Consequently, solving the ranked optimal assignment problem with cost matrix $C_Z^{(I,\xi)}$ generates, starting from θ^*, an enumeration of association maps θ in order of non-increasing $[\eta_Z^{(\xi,\theta)}]^I$ (and weights $\omega^{(I,\xi,\theta)}(Z) \propto \omega^{(I,\xi)}[\eta_Z^{(\xi,\theta)}]^I$).

Remark: The standard ranked assignment formulation involves square cost and assignment matrices with rows and columns of the assignment matrix summing to 1. Ranked assignment problems with non-square matrices can be reformulated with square matrices by introducing dummy variables.

The optimal assignment problem, is a well-known combinatorial problem, introduced by Kuhn [36], who also proposed the Hungarian algorithm to solve it in polynomial time. Munkres further observed that it is strongly polynomial [38]. The ranked assignment problem is a generalization to enumerate the T least cost assignments, which was first solved by Murty [37]. Murty's algorithm needs an effective bipartite assignment algorithm such as Munkres [38] or Jonker-Volgenant [39]. In the context of multi-target tracking, ranked assignment algorithms with $O(T|Z|^4)$ complexity have been proposed for MHT in [40], [41], [42]. More efficient algorithms with $O(T|Z|^3)$ complexity have been proposed in [44], [45], [46], with the latter showing better efficiency for large $|Z|$. For further details on ranked assignment solutions in MHT, we refer the reader to [2], [43].

B. Computing update parameters

We now detail the computation of the cost matrix $C_Z^{(I,\xi)}$ in (23) for the ranked assignment problem and the parameters $\eta_Z^{(\xi,\theta)}(\ell)$, $p^{(\xi,\theta)}(\cdot,\ell|Z)$ of the updated δ-GLMB components.

1) Gaussian mixture: For a linear Gaussian multi-target model, $p_D(x,\ell) = p_D$, $g(z|x,\ell) = \mathcal{N}(z;Hx,R)$, where $\mathcal{N}(\cdot;m,P)$ denotes a Gaussian density with mean m and covariance P, H is the observation matrix, and R is the observation noise covariance. The Gaussian mixture representation provides the most general setting for linear Gaussian models. Suppose that each single target density $p^{(\xi)}(\cdot,\ell)$ is a Gaussian mixture of the form

$$\sum_{i=1}^{J^{(\xi)}(\ell)} w_i^{(\xi)}(\ell)\mathcal{N}(x;m_i^{(\xi)}(\ell),P_i^{(\xi)}(\ell)). \tag{25}$$

Then

$$C_{i,j} = -\ln\left(\frac{p_D\sum_{k=1}^{J^{(\xi)}(\ell_i)} w_k^{(\xi)}(\ell_i)q_k^{(\xi)}(z_j;\ell_i)}{(1-p_D)\kappa(z_j)}\right) \tag{26}$$

Moreover, for the updated association history (ξ,θ),

$$\eta_Z^{(\xi,\theta)}(\ell) = \sum_{i=1}^{J^{(\xi)}(\ell)} w_{Z,i}^{(\xi,\theta)}(\ell) \tag{27}$$

$$p^{(\xi,\theta)}(x,\ell|Z) = \sum_{i=1}^{J^{(\xi)}(\ell)} \frac{w_{Z,i}^{(\xi,\theta)}(\ell)}{\eta_Z^{(\xi,\theta)}(\ell)}\mathcal{N}(x;m_{Z,i}^{(\xi,\theta)}(\ell),P_i^{(\xi,\theta)}(\ell)) \tag{28}$$

where

$$w_{Z,i}^{(\xi,\theta)}(\ell) = w_i^{(\xi)}(\ell)\begin{cases}\frac{p_D q_i^{(\xi)}(z_{\theta(\ell)};\ell)}{\kappa(z_{\theta(\ell)})} & \text{if } \theta(\ell)>0\\ (1-p_D) & \text{if } \theta(\ell)=0\end{cases}$$

$$q_i^{(\xi)}(z;\ell) = \mathcal{N}(z;Hm_i^{(\xi)}(\ell),HP_i^{(\xi)}(\ell)H^T+R)$$

$$m_{Z,i}^{(\xi,\theta)}(\ell) = \begin{cases}m_i^{(\xi)}(\ell)+K_i^{(\xi,\theta)}(\ell)(z_{\theta(\ell)}-Hm_i^{(\xi)}(\ell)) & \text{if } \theta(\ell)>0\\ m_i^{(\xi)}(\ell) & \text{if } \theta(\ell)=0\end{cases}$$

$$P_i^{(\xi,\theta)}(\ell) = [I-K_i^{(\xi,\theta)}(\ell)H]P_i^{(\xi)}(\ell),$$

$$K_i^{(\xi,\theta)}(\ell) = \begin{cases}P_i^{(\xi)}(\ell)H^T\left[HP_i^{(\xi)}(\ell)H^T+R\right]^{-1} & \text{if } \theta(\ell)>0\\ 0 & \text{if } \theta(\ell)=0\end{cases}.$$

When the measurement model parameters depend on the label ℓ, we simply substitute $p_D = p_D(\ell), H = H(\ell), R = R(\ell)$ into the above equations.

2) Sequential Monte Carlo: For a sequential Monte Carlo approximation, suppose that each of the single target density $p^{(\xi)}(\cdot,\ell)$ is represented as a set of weighted samples $\{(w_n^{(\xi)}(\ell),x_n^{(\xi)}(\ell))\}_{n=1}^{J^{(\xi)}(\ell)}$. Then

$$C_{i,j} = -\ln\left(\frac{\sum_{n=1}^{J^{(\xi)}(\ell_i)} w_n^{(\xi)}(\ell_i)p_D(x_n^{(\xi)}(\ell_i),\ell_i)g(z_j|x_n^{(\xi)}(\ell_i),\ell_i)}{\sum_{n=1}^{J^{(\xi)}(\ell_i)} w_n^{(\xi)}(\ell_i)(1-p_D(x_n^{(\xi)}(\ell_i),\ell_i))\kappa(z_j)}\right). \tag{29}$$

Moreover, for a given updated association history (ξ,θ),

$$\eta_Z^{(\xi,\theta)}(\ell) = \sum_{n=1}^{J^{(\xi)}(\ell)} w_n^{(\xi)}(\ell)\psi_Z(x_n^{(\xi)}(\ell),\ell;\theta), \tag{30}$$

and $p^{(\xi,\theta)}(\cdot,\ell|Z)$ is represented by the following set of weighted samples

$$\left\{\left(\frac{\psi_Z(x_n^{(\xi)}(\ell),\ell;\theta)w_n^{(\xi)}(\ell)}{\eta_Z^{(\xi,\theta)}(\ell)},x_n^{(\xi)}(\ell)\right)\right\}_{n=1}^{J^{(\xi)}(\ell)}. \tag{31}$$

C. Filtering Density Truncation

Given the δ-GLMB prediction density $\boldsymbol{\pi}$ with enumerated parameter set $\{(I^{(h)},\xi^{(h)},\omega^{(h)},p^{(h)})\}_{h=1}^H$, the δ-GLMB filtering density (13) can be written as

$$\boldsymbol{\pi}(\mathbf{X}|Z) = \sum_{h=1}^{H}\boldsymbol{\pi}^{(h)}(\mathbf{X}|Z) \tag{32}$$

where

$$\boldsymbol{\pi}^{(h)}(\mathbf{X}|Z) = \Delta(\mathbf{X})\sum_{j=1}^{|\Theta(I^{(h)})|}\omega^{(h,j)}\delta_{I^{(h)}}(\mathcal{L}(\mathbf{X}))\left[p^{(h,j)}\right]^{\mathbf{X}},$$

$$\omega^{(h,j)} \triangleq \omega^{(I^{(h)},\xi^{(h)},\theta^{(h,j)})}(Z),$$

$$p^{(h,j)} \triangleq p^{(\xi^{(h)},\theta^{(h,j)})}(\cdot|Z).$$

Each δ-GLMB prediction component (indexed by) h generates $\left|\Theta(I^{(h)})\right|$ components for the δ-GLMB filtering density.

A simple and highly parallelizable strategy for truncating the filtered δ-GLMB (32) is to truncate $\boldsymbol{\pi}^{(h)}(\cdot|Z)$. For each $h = 1,...,H$, solving the ranked optimal assignment problem with cost matrix $C_Z^{(I^{(h)},\xi^{(h)})}$ discussed in subsection IV-A yields $\theta^{(h,j)}$, $j = 1,...,T^{(h)}$, the $T^{(h)}$ hypotheses with highest

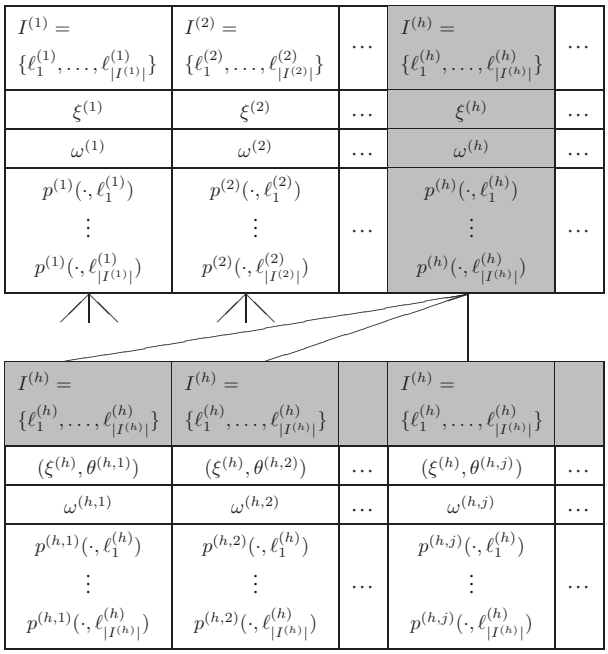

Fig. 3. δ-GLMB update. Component h of the prior generates a (large) set of posterior components. The ranked assignment algorithm determines the $T^{(h)}$ components with highest weights $\omega^{(h,1)} \geq \omega^{(h,2)} \geq ... \geq \omega^{(h,T^{(h)})}$.

weights in non-increasing order, as illustrated in Figure 3. Consequently, the truncated version of $\boldsymbol{\pi}^{(h)}(\cdot|Z)$ is

$$\hat{\boldsymbol{\pi}}^{(h)}(\mathbf{X}|Z) = \Delta(\mathbf{X}) \sum_{j=1}^{T^{(h)}} \omega^{(h,j)} \delta_{I^{(h)}}(\mathcal{L}(\mathbf{X})) \left[p^{(h,j)}\right]^{\mathbf{X}}.$$

It follows from Proposition 5 that the truncated density $\hat{\boldsymbol{\pi}}(\cdot|Z) = \sum_{h=1}^{H} \hat{\boldsymbol{\pi}}^{(h)}(\cdot|Z)$ minimizes the L_1-distance from the filtered δ-GLMB over all truncations with $T^{(h)}$ components for each $h = 1, ..., H$. The truncated density, with a total of $T = \sum_{h=1}^{H} T^{(h)}$ components, is normalized (by the sum of the weights) to give the truncated filtered δ-GLMB. Table 1 summarizes the update operation via pseudo code. Note that both the outer and inner for-loops can be parallelized.

Specific values for the number of requested components $T^{(h)}$ are generally user specified and application dependent. A generic strategy is to choose $T^{(h)} = \lceil \omega^{(h)} J_{\max} \rceil$ where $J_{\max}$ is the desired overall number of hypotheses. The alternative strategy of keeping the $T = J_{\max}$ strongest components of $\boldsymbol{\pi}(\cdot|Z)$ would yield a smaller L_1-error. However, in addition to an H-fold increase in the dimension of the resulting ranked assignment problem, parallelizability is lost.

V. Delta-GLMB Prediction

This subsection presents an implementation of the δ-GLMB prediction using the K-shortest path algorithm to truncate the predicted δ-GLMB without computing all the prediction hypotheses and their weights.

The prediction density given in Proposition 4 has a compact form but is difficult to implement due to the sum over all supersets of L in (19). We use an equivalent form for the prediction, eq. (58) in [1]:

$$\boldsymbol{\pi}_+(\mathbf{X}_+) = \Delta(\mathbf{X}_+) \sum_{(I,\xi)\in\mathcal{F}(\mathbb{L})\times\Xi} \omega^{(I,\xi)} \sum_{J\in\mathcal{F}(I)} [\eta_S^{(\xi)}]^J [1-\eta_S^{(\xi)}]^{I-J} \times \sum_{L\in\mathcal{F}(\mathbb{B})} w_B(L)\delta_{J\cup L}(\mathcal{L}(\mathbf{X}_+)) \left[p_+^{(\xi)}\right]^{\mathbf{X}_+}, \quad (33)$$

Table 1. Update

- input: $\{(I^{(h)}, \xi^{(h)}, \omega^{(h)}, p^{(h)}, T^{(h)})\}_{h=1}^{H}$, Z
- output: $\{(I^{(h,j)}, \xi^{(h,j)}, \omega^{(h,j)}, p^{(h,j)})\}_{(h,j)=(1,1)}^{(H,T^{(h)})}$

```
for h = 1 : H
  C_Z^(h) := C_Z^(I^(h),ξ^(h))   according to (23), (26)/(29)
  {θ^(h,j)}_{j=1}^{T^(h)} := ranked_assignment(Z, I^(h), C_Z^(h), T^(h))
  for j = 1 : T^(h)
    η_Z^(h,j) := η_Z^(ξ^(h),θ^(h,j))   according to (27)/(30)
    p^(h,j) := p^(ξ^(h),θ^(h,j))(·|Z)   according to (28)/(31)
    ω^(h,j) := ω^(h) [η_Z^(h,j)]^{I^(h)}
    I^(h,j) := I^(h)
    ξ^(h,j) := (ξ^(h),θ^(h,j))
  end
end
normalize weights {ω^(h,j)}_{(h,j)=(1,1)}^{(H,T^(h))}
```

Note that analogous to the update, each current hypothesis (I, ξ) with weight $\omega^{(I,\xi)}$ generates a set of prediction hypotheses $(J \cup L, \xi)$, $J \subseteq I$, $L \subseteq \mathbb{B}$, with weights $\omega_S^{(I,\xi)}(J) w_B(L)$, where

$$\omega_S^{(I,\xi)}(J) = \omega^{(I,\xi)} [\eta_S^{(\xi)}]^J [1-\eta_S^{(\xi)}]^{I-J} \quad (34)$$

Intuitively, each predicted label set $J \cup L$ consists of a surviving label set J with weight $\omega_S^{(I,\xi)}(J)$ and a birth label set L with weight $w_B(L)$. The weight $\omega_S^{(I,\xi)}(J)$ can be interpreted as the probability that the current label set is I, and the labels in J survives to the next time while the remaining labels $I-J$ die. The birth label set L and the surviving label set J are mutually exclusive since the space of new labels $\mathbb{B}$ cannot contain any existing labels. Since the weight of $J \cup L$ is the product $\omega_S^{(I,\xi)}(J) w_B(L)$, we can truncate the double sum over J and L by separately truncating the sum over J and the sum over L.

Subsection V-A discusses the K-shortest paths problem in the context of truncating the δ-GLMB prediction. Subsection V-B details the computation of the prediction δ-GLMB parameters and subsection V-C presents the δ-GLMB prediction algorithm.

A. K-Shortest Paths Problem

Consider a given hypothesis (I, ξ), and note that the weight of a surviving label set $J \subseteq I$ can be rewritten as

$$\omega_S^{(I,\xi)}(J) = \omega^{(I,\xi)} [1-\eta_S^{(\xi)}]^I \left[\frac{\eta_S^{(\xi)}}{1-\eta_S^{(\xi)}}\right]^J$$

If we can generate the surviving label sets $J \subseteq I$ in non-increasing order of $[\eta_S^{(\xi)}/(1-\eta_S^{(\xi)})]^J$, then the highest weighted survival sets for hypothesis (I, ξ) can be selected without exhaustively computing all the survival hypotheses weights. This can be accomplished by solving the K-shortest path problem in the directed graph of Figure 4.

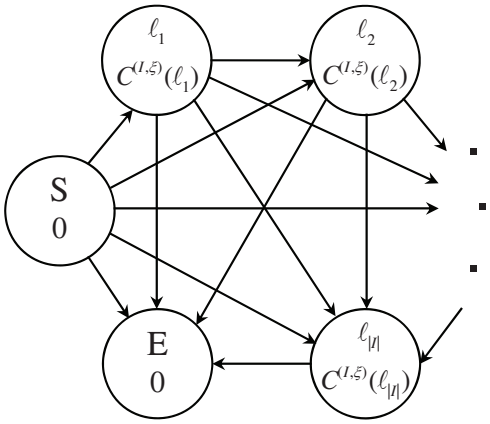

Fig. 4. A directed graph with nodes ℓ_1,...,$\ell_{|I|} \in I$, and corresponding costs $C^{(I,\xi)}(\ell_1)$,...,$C^{(I,\xi)}(\ell_{|I|})$. S and E are the start and end nodes respectively.

Define a cost vector $C^{(I,\xi)} = \left[C^{(I,\xi)}(\ell_1), ..., C^{(I,\xi)}(\ell_{|I|})\right]$, where

$$C^{(I,\xi)}(\ell_j) = -\ln\left[\frac{\eta_S^{(\xi)}(\ell_j)}{1-\eta_S^{(\xi)}(\ell_j)}\right] \tag{35}$$

is the cost of node $\ell_j \in I$ (the numerical computation of $C^{(I,\xi)}(\ell_j)$ is detailed in Subsection V-B). The nodes are ordered in non-decreasing costs and the distance from node ℓ_j to $\ell_{j'}$ is defined as

$$d(\ell_j, \ell_{j'}) = \begin{cases} C^{(I,\xi)}(\ell_{j'}), & \text{if } j' > j \\ \infty, & \text{otherwise} \end{cases}$$

Hence, a path from S to E which traverses the set of nodes $J \subseteq I$ accumulates a total distance of

$$\begin{aligned} \sum_{\ell\in J} C^{(I,\xi)}(\ell) &= -\sum_{\ell\in J} \ln\left(\eta_S^{(\xi)}(\ell)/(1-\eta_S^{(\xi)}(\ell))\right) \\ &= -\ln\left([\eta_S^{(\xi)}(\ell)/(1-\eta_S^{(\xi)}(\ell)]^J\right). \end{aligned}$$

The shortest path from S to E traverses the set of nodes $J^* \subseteq I$ with the shortest distance $\sum_{\ell\in J^*} C^{(I,\xi)}(\ell)$ and hence largest $[\eta_S^{(\xi)}/(1-\eta_S^{(\xi)})]^{J^*}$. The K-shortest paths problem seeks K subsets of I with the shortest distances in non-decreasing order. Consequently, solving the K-shortest path problem generates, starting from J^*, an enumeration of subsets J of I in order of non-increasing $[\eta_S^{(\xi)}/(1-\eta_S^{(\xi)})]^J$.

For the target births we use a labeled multi-Bernoulli birth model where

$$\begin{aligned} w_B(L) &= \prod_{\ell\in\mathbb{B}}\left(1-r_B^{(\ell)}\right)\prod_{\ell\in L}\frac{1_{\mathbb{B}}(\ell)r_B^{(\ell)}}{1-r_B^{(\ell)}}, \\ p_B(x,\ell) &= p_B^{(\ell)}(x). \end{aligned}$$

Thus, solving the K-shortest paths problem with cost vector $C_B = \left[C_B(\ell_1), ..., C_B(\ell_{|\mathbb{B}|})\right]$, where

$$C_B(\ell_j) = -\ln\left[r_B^{(\ell_j)}/(1-r_B^{(\ell_j)})\right] \tag{36}$$

is the cost of node ℓ_j, yields the subsets of $\mathbb{B}$ with the best birth weights.

Remark: It is possible to obtain the overall K best components by extending the directed graphs to include birth nodes with appropriate costs. However, our experience indicated that since the birth weights $w_B(L)$ are quite small compared to surviving weights $\omega_S^{(I,\xi)}(J)$, many components with births will be discarded and new births may not detected by the filter. To avoid dropping new tracks, a very large K is required to retain hypothesis with births. On the other hand the proposed separate truncation strategy ensures that there are hypotheses with births to accommodate new tracks, and is highly parallelizable.

The K-shortest paths algorithm is a well-known solution to the combinatorial problem of finding the K paths with minimum total cost from a given source to a given destination in a weighted network [47]. This problem can be solved with complexity $O\left(|I|\log(|I|)+K\right)$. In our case the nodes have negative values, hence the Bellman-Ford algorithm [48], [49] was employed. This problem can also be solved by ranked assignment algorithms, however K-shortest paths algorithm is much more efficient. Note that the ranking of the association maps cannot be formulated as the K-shortest path problem, due to the constraint that each target can only generate at most one measurement.

B. Computing prediction parameters

This subsection details the computation of the parameters $\eta_S^{(\xi)}(\ell)$ and $p_+^{(\xi)}(\cdot,\ell)$ of the prediction δ-GLMB components.

1) Gaussian mixture: For a linear Gaussian multi-target model, $p_S(x,\ell) = p_S$, $f(x_+|x,\ell) = \mathcal{N}(x_+; Fx, Q)$, where F is the state transition matrix, Q is the process noise covariance and the birth density parameter $p_B^{(\ell)}(x)$ is a Gaussian mixture. If the single target density $p^{(\xi)}(\cdot,\ell)$ is a Gaussian mixture of the form (25). Then,

$$\eta_S^{(\xi)}(\ell) = p_S, \tag{37}$$

$$p_+^{(\xi)}(x,\ell) = 1_{\mathbb{L}}(\ell)\sum_{i=1}^{J^{(\xi)}(\ell)} w_i^{(\xi)}(\ell)\mathcal{N}(x; m_{S,i}^{(\xi)}(\ell), P_{S,i}^{(\xi)}(\ell)) + 1_{\mathbb{B}}(\ell)p_B^{(\ell)}(x) \tag{38}$$

where

$$\begin{aligned} m_{S,i}^{(\xi)}(\ell) &= Fm_i^{(\xi)}(\ell), \\ P_{S,i}^{(\xi)}(\ell) &= Q + FP_i^{(\xi)}(\ell)F^T. \end{aligned}$$

When the motion model parameters depend on the label ℓ, we simply substitute $p_S = p_S(\ell)$, $F = F(\ell)$, $Q = Q(\ell)$ into the above equations. With $\eta_S^{(\xi)}(\ell)$ evaluated, the node cost $C^{(I,\xi)}(\ell)$ for the K-shortest paths problem can then be computed by (35).

2) Sequential Monte Carlo: For a sequential Monte Carlo approximation, suppose that each single target density $p^{(\xi)}(\cdot,\ell)$ is represented as a set of weighted sample $\{(w_i^{(\xi)}(\ell), x_i^{(\xi)}(\ell))\}_{i=1}^{J^{(\xi)}(\ell)}$ and that the birth density $p_B^{(\ell)}(\cdot)$ is represented by $\{(w_{B,i}^{(\xi)}(\ell), x_{B,i}^{(\xi)}(\ell))\}_{i=1}^{B^{(\xi)}(\ell)}$. Then,

$$\eta_S^{(\xi)}(\ell) = \sum_{i=1}^{J^{(\xi)}(\ell)} w_i^{(\xi)}(\ell)p_S(x_i^{(\xi)}(\ell),\ell) \tag{39}$$

and $p_+^{(\xi)}(x,\ell)$ is represented by

$$\left\{(1_{\mathbb{L}}(\ell)\tilde{w}_{S,i}^{(\xi)}(\ell), x_{S,i}^{(\xi)}(\ell))\right\}_{i=1}^{J^{(\xi)}(\ell)} \cup \left\{(1_{\mathbb{B}}(\ell)w_{B,i}^{(\xi)}(\ell), x_{B,i}^{(\xi)}(\ell))\right\}_{i=1}^{B^{(\xi)}(\ell)}, \tag{40}$$

where

$$\begin{aligned} x_{S,i}^{(\xi)}(\ell) &\sim q^{(\xi)}(\cdot|x_i^{(\xi)}(\ell),\ell,Z),\ i=1,...,J^{(\xi)}(\ell), \\ w_{S,i}^{(\xi)}(\ell) &= \frac{w_i^{(\xi)}(\ell) f(x_{S,i}^{(\xi)}(\ell)|x_i^{(\xi)}(\ell)) p_S(x_i^{(\xi)}(\ell),\ell)}{q^{(\xi)}(x_{S,i}^{(\xi)}(\ell)|x_i^{(\xi)}(\ell),\ell,Z)}, \\ \tilde{w}_{S,i}^{(\xi)}(\ell) &= \frac{w_{S,i}^{(\xi)}(\ell)}{\sum_{i=1}^{J^{(\xi)}(\ell)} w_{S,i}^{(\xi)}(\ell)}, \end{aligned}$$

and $q^{(\xi)}(\cdot|x_i^{(\xi)}(\ell),\ell,Z)$ is a proposal density.

C. Prediction Density Truncation

Given the current δ-GLMB filtering density $\boldsymbol{\pi}(\cdot|Z)$ with enumerated parameter set $\{(I^{(h)},\xi^{(h)},\omega^{(h)},p^{(h)})\}_{h=1}^{H}$, then the δ-GLMB prediction (33) becomes

$$\boldsymbol{\pi}_+(\mathbf{X}_+) = \sum_{h=1}^{H} \boldsymbol{\pi}_+^{(h)}(\mathbf{X}_+),$$

where

$$\boldsymbol{\pi}_+^{(h)}(\mathbf{X}_+) = \Delta(\mathbf{X}_+) \sum_{J\subseteq I^{(h)}} \sum_{L\subseteq\mathbb{B}} \omega_S^{(I^{(h)},\xi^{(h)})}(J) w_B(L) \delta_{J\cup L}(\mathcal{L}(\mathbf{X}_+)) \left[p_+^{(\xi^{(h)})}\right]^{\mathbf{X}_+}.$$

The δ-GLMB filtered component (indexed by) h generates $2^{|I^{(h)}|+|\mathbb{B}|}$components for the δ-GLMB prediction.

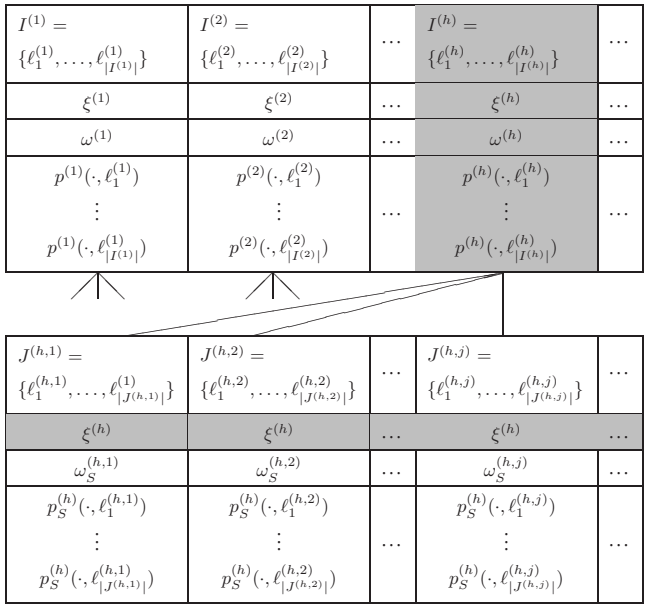

Fig. 5. Prediction of survival components. Component h of the prior, generates all subsets of $I^{(h)}$, i.e. $J^{(h,j)}, j = 1,..., 2^{|I^{(h)}|}$ with weights $\omega_S^{(h,j)} \triangleq \omega_S^{(I^{(h)},\xi^{(h)})}(J^{(h,j)})$. The K-shortest paths algorithm determines the $K^{(h)}$ subsets with largest weights $\omega_S^{(h,1)} \geq \omega_S^{(h,2)} \geq ... \geq \omega_S^{(h,K^{(h)})}$.

A simple and highly parallelizable strategy for truncating the prediction δ-GLMB $\boldsymbol{\pi}_+$ is to truncate each $\boldsymbol{\pi}_+^{(h)}$ as follows. For each $h = 1,...,H$, we solve the K-shortest paths problem with cost vector $C^{(I^{(h)},\xi^{(h)})}$ to obtain $J^{(h,j)}, j = 1,...,K^{(h)}$, the $K^{(h)}$ subsets of $I^{(h)}$ with highest survival weights as depicted in Figure 5. We also solve the K-shortest paths problem with cost vector C_B to obtain $L^{(b)}, b = 1,...,K_B$, the K_B birth subsets with highest birth weights. Consequently for each h, the truncated version of $\boldsymbol{\pi}_+^{(h)}$ is

$$\hat{\boldsymbol{\pi}}_+^{(h)}(\mathbf{X}_+) = \Delta(\mathbf{X}_+) \sum_{j=1}^{K^{(h)}} \sum_{b=1}^{K_B} \omega_+^{(h,j,b)} \delta_{J^{(h,j)}\cup L^{(b)}}(\mathcal{L}(\mathbf{X}_+)) \left[p_+^{(h)}\right]^{\mathbf{X}_+},$$

where

$$\begin{aligned} \omega_+^{(h,j,b)} &\triangleq \omega_S^{(I^{(h)},\xi^{(h)})}(J^{(h,j)}) w_B(L^{(b)}) \\ p_+^{(h)} &\triangleq p_+^{(\xi^{(h)})}. \end{aligned}$$

Since the weights of the (untruncated) prediction density sum to 1, it follows from Proposition 5 that the resulting truncated density $\hat{\boldsymbol{\pi}}_+ = \sum_{h=1}^{H} \hat{\boldsymbol{\pi}}_+^{(h)}$, which has a total of $T = K_B \sum_{h=1}^{H} K^{(h)}$ components, incurs an L_1 truncation error of

$$1 - \sum_{h=1}^{H} \sum_{j=1}^{K^{(h)}} \sum_{b=1}^{K_B} \omega_+^{(h,j,b)}.$$

Moreover, the truncated density minimizes the L_1-distance from the predicted δ-GLMB over all truncations with $K^{(h)}$ components for each $h = 1,...,H$, and K_B components for the births. The final expression for the approximation is obtained by normalizing the truncated density. Table 2 shows the pseudo code for the prediction operation. Note that all three for-loops can be implemented in parallel.

Table 2. Prediction

- input: $\{(I^{(h)},\xi^{(h)},\omega^{(h)},p^{(h)},K^{(h)})\}_{h=1}^{H}$
- input: K_B, $\{(r_B^{(\ell)},p_B^{(\ell)})\}_{\ell\in\mathbb{B}}$,
- output $\{(I_+^{(h,j,b)},\omega_+^{(h,j,b)},p_+^{(h)})\}_{(h,j,b)=(1,1,1)}^{(H,K^{(h)},K_B)}$

compute C_B according to (36)
$\{L^{(b)}\}_{b=1}^{K_B} := \mathsf{k\_shortest\_path}(\mathbb{B}, C_B, K_B)$
for $b = 1 : K_B$
  $\omega_B^{(b)} := \prod_{\ell\in L^{(b)}} r_B^{(\ell)} \prod_{\ell\in\mathbb{B}-L^{(b)}} \left[1-r_B^{(\ell)}\right]$
end
for $h = 1 : H$
  $\eta_S^{(h)} := \eta_S^{(\xi^{(h)})}$ according to (37)/(39)
  $C^{(h)} := C^{(I^{(h)},\xi^{(h)})}$ according to (35)
  $\{J^{(h,j)}\}_{j=1}^{K^{(h)}} := \mathsf{k\_shortest\_path}(I^{(h)}, C^{(h)}, K^{(h)})$
  for $(j,b) = (1,1) : (K^{(h)}, K_B)$
    $I_+^{(h,j,b)} := J^{(h,j)} \cup L^{(b)}$
    $\omega_+^{(h,j,b)} := \omega^{(h)} \left[\eta_S^{(h)}\right]^{J^{(h,j)}} \left[1-\eta_S^{(h)}\right]^{I^{(h)}-J^{(h,j)}} \omega_B^{(b)}$
  end
  $p_+^{(h)} := p_+^{(\xi^{(h)})}$ according to (38)/(40)
end
normalize weights $\{\omega_+^{(h,j,b)}\}_{(h,j,b)=(1,1,1)}^{(H,K^{(h)},K_B)}$

Specific values for the number of requested components $K^{(h)}$ and K_B are generally user specified and application dependent. A generic strategy is to choose $K^{(h)} = \lceil \omega^{(h)} J_{\max} \rceil$

where $J_{\max}$ is the desired overall number of hypotheses, and to choose K_B such that the resulting truncation captures a desired proportion (say 99%) of the probability mass of the birth density. The alternative strategy of keeping the $T = J_{\max}$ strongest components of $\boldsymbol{\pi}_+$ would yield a smaller L_1-error than the proposed strategy. However, in addition to an $(H + K_B)$-fold increase in the dimension of the resulting problem, parallelizability is lost.

Remark: As noted previously, the actual value of the association history $\xi^{(h)}$ is not needed in the update and prediction calculations, it is used merely as an index for the track density $p^{(\xi^{(h)})}$. Since the track density is now equivalently indexed by h, i.e. $p^{(h)} \triangleq p^{(\xi^{(h)})}$, in practice it is not necessary to propagate $\xi^{(h)}$. Nonetheless, for clarity of exposition we have retained $\xi^{(h)}$ in the update and prediction pseudo codes.

VI. Delta GLMB filter

The main steps of the δ-GLMB filter algorithm is summarized in the following pseudo code:

Main Loop (Filter)

for $k = 1 : K$
  Prediction
  Update
  Compute State Estimates
end

In Subsection VI-A we describe the multi-target state estimation process in the "Compute State Estimate" module. Subsection VI-B presents look-ahead strategies to reduce number of calls to the ranked optimal assignment and K-shortest paths algorithms.

A. Multi-target state estimation

Given a multi-target filtering density, several multi-target state estimators are available. The Joint Multi-object Estimator and Marginal Multi-object Estimator are Bayes optimal, but difficult to compute [3]. A simple and intuitive multi-target estimator for a δ-GLMB density is the multi-Bernoulli estimator, which selects the set of tracks or labels $L \subseteq \mathbb{L}$ with existence probabilities above a certain threshold, and for the states of the tracks, the maximum *a posteriori* (MAP) or the mean estimates from the densities $p^{(\xi)}(\cdot, \ell), \ell \in L$. From [1], the existence probability of track ℓ is given by the sum of the weights of all hypotheses containing track ℓ :

$$\sum_{(I,\xi)\in\mathcal{F}(\mathbb{L})\times\Xi} \omega^{(I,\xi)} 1_I(\ell).$$

An alternative multi-target estimate is the MAP or the mean estimate of the states of the hypothesis with the highest weight.

In this work we use a suboptimal but tractable version of the Marginal Multi-object Estimator by finding the MAP cardinality estimate from the cardinality distribution [1]:

$$\boldsymbol{\rho}(n) = \sum_{(I,\xi)\in\mathcal{F}_n(\mathbb{L})\times\Xi} \omega^{(I,\xi)},$$

Table 3. Compute State Estimate

- input: $N_{\max}$, $\{(I^{(h,j)}, \xi^{(h,j)}, \omega^{(h,j)}, p^{(h,j)})\}_{(h,j)=(1,1)}^{(H,T^{(h)})}$
- output: $\hat{\mathbf{X}}$

$\boldsymbol{\rho}(n) := \sum_{h=1}^{H} \sum_{j=1}^{T^{(h)}} \omega^{(h,j)} \delta_n(|I^{(h,j)}|)$; $n = 0, ..., N_{\max}$
$\hat{N} := \arg\max \boldsymbol{\rho}$
$(\hat{h}, \hat{\jmath}) := \arg\max_{(h,j)} \omega^{(h,j)} \delta_{\hat{N}}(|I^{(h,j)}|)$
$\hat{\mathbf{X}} := \{(x, \ell) : \ell \in I^{(\hat{h},\hat{\jmath})}, x = \int y p^{(\hat{h},\hat{\jmath})}(y, \ell) dy\}$

where $\mathcal{F}_n(\mathbb{L})$ denotes the class of finite subsets of $\mathbb{L}$ with exactly n elements. We then find the labels and the mean estimates of the states from the highest weighted component that has the same cardinality as the MAP cardinality estimate. Table 3 shows the pseudo code for the multi-target state estimation

B. PHD look-ahead

In this subsection we use the PHD/CPHD update and prediction [8], [29], [50]–[52] to look-ahead and identify the prediction/update components that generate significant updated/predicted components. This is analogous to the idea of using a measurement driven proposal in the particle filter, to guide particles towards important areas of the state space [53], [54].

Recall that each prediction component generates a set of update components. Typically, more than 90% of the best predicted components generate update components with negligible weights. Since updating is expensive, it is important to minimize the number of prediction components to be updated (and hence the number of calls to the ranked assignment algorithm). Knowing in advance which prediction components would generate significant update components will save substantial computations. Similarly, further saving in computations can be achieved by knowing in advance which update components would generate significant prediction components.

The PHD/CPHD filter is a good approximation to the multi-target Bayes filter and is inexpensive compared to the δ-GLMB update. Moreover, integration of the PHD filter within the δ-GLMB filter is seamless as both filters are developed from the same RFS framework. Indeed, the PHD of (the unlabeled version of) a δ-GLMB of the form (9) is given by [1]:

$$v(x) = \sum_{(I,\xi)\in\mathcal{F}(\mathbb{L})\times\Xi} v^{(I,\xi)}(x)$$

where

$$v^{(I,\xi)}(x) = \sum_{\ell\in I} \omega^{(I,\xi)} p^{(\xi)}(x, \ell).$$

is the PHD of hypothesis (I, ξ). Assuming that the detection probability and single measurement likelihood do not depend on the labels, the updated PHD is given by (see [8], [50], [51])

$$v(x|Z) = \sum_{(I,\xi)\in\mathcal{F}(\mathbb{L})\times\Xi} \hat{v}^{(I,\xi)}(x|Z),$$

where the constituent updated PHD $\hat{v}^{(I,\xi)}(\cdot|Z)$ due to hypothesis (I,ξ) is given by

$$\hat{v}^{(I,\xi)}(x|Z) = (1-p_D(x))v^{(I,\xi)}(x) + \sum_{z\in Z}\frac{p_D(x)g(z|x)v^{(I,\xi)}(x)}{\kappa(z)+\sum_{(I,\xi)\in\mathcal{F}(\mathbb{L})\times\Xi}\left\langle p_D g(z|\cdot), v^{(I,\xi)}\right\rangle},$$

Note that $\hat{v}^{(I,\xi)}(\cdot|Z)$ is not the same as the updated PHD of $v^{(I,\xi)}$ given by

$$v^{(I,\xi)}(x|Z) = (1-p_D(x))v^{(I,\xi)}(x) + \sum_{z\in Z}\frac{p_D(x)g(z|x)v^{(I,\xi)}(x)}{\kappa(z)+\left\langle p_D(x)g(z|x), v^{(I,\xi)}(x)\right\rangle}.$$

The ratio of the constituent updated PHD mass $\int \hat{v}^{(I,\xi)}(x|Z)dx$ to the total updated PHD mass $\int v(x|Z)dx$ can be thought of as a selection criteria and is a good indicator of the significance of hypothesis (I,ξ) after the update. A higher score indicates a greater significance. The proposed look-ahead strategy selects prediction hypotheses with highest constituent updated PHD masses that together makes up most (say 95%) of the total updated PHD mass. These PHD masses can be readily computed with $O(|Z|)$ complexity, using SMC [50] or Gaussian mixtures [51]. Further improvement can be achieved by replacing the PHD update with the CPHD update with $O(|Z|^3)$ complexity [29], [52]. Note that the δ-GLMB update can reuse the computations in the PHD/CPHD update such as the Kalman gain and other variables.

A similar strategy can be employed to identify updated components that generates weak prediction hypotheses. Using the constituent predicted PHD masses as the selection criterion, we select updated hypotheses whose combined predicted PHD mass makes up most of the total predicted PHD mass.

A parallelizable look-ahead strategy can be formulated using the updated PHD $v^{(I,\xi)}(\cdot|Z)$. The relative cardinality error $\left|\int v^{(I,\xi)}(x|Z)dx - |I|\right|/|I|$ of hypothesis (I,ξ) is a measure of how well it explains the observed data Z, and is a possible selection criterion. Additional selection criteria are possible with the CPHD update since the cardinality distribution is available. Consider the PHD $v^{(I,\xi)}$ of hypothesis (I,ξ) and a Poisson cardinality distribution $\rho^{(I,\xi)}$ with mean $\int v^{(I,\xi)}(x)dx$. If hypothesis (I,ξ) explains the observed data well, then the CPHD updated cardinality distribution $\rho^{(I,\xi)}(\cdot|Z)$ should be close to $\delta_{|I|}$. Hence, a possible selection criterion is the Kullback-Leibler divergence between $\rho^{(I,\xi)}(\cdot|Z)$ and $\delta_{|I|}$. In both cases a lower score indicates a greater significance for hypothesis (I,ξ) after the update.

Regardless of the particular look ahead strategy used, the selection criterion yields an unnormalized score for each hypothesis or component, which generally indicates how well the component explains the observed data. Thus for implementation a normalized score for each component can be derived such that a higher score indicates a better fit. A generic method for setting $T^{(h)}$ is then to choose its value to be proportional to the normalized score for component $(I^{(h)},\xi^{(h)})$ and the desired overall number of components $J_{\max}$.

VII. Numerical Example

A non-linear example has been presented in [1]. In this section, we compare the performance of the δ-GLMB and CPHD filters with a linear Gaussian example and hence Gaussian mixture implementation. A typical scenario is employed to highlight the susceptibility of the CPHD filter to the so called "spooky effect" [55]. The term spooky effect refers to phenomenon where the CPHD filter encounters a missed detection for a particular track and significantly reduces the weight of the undetected track by shifting part of its mass to other targets. The δ-GLMB filter is generally immune to the spooky effect since it does not use an i.i.d. approximation of the filtering density and hence cannot shift the probability mass of individual tracks between each other. Furthermore the δ-GLMB filter is generally able to correct for CPHD look ahead errors, since the weights of the CPHD-generated hypotheses are computed exactly from the δ-GLMB update.

Consider a set of multi-target trajectories on the two dimensional region $[-1000,1000]m \times [-1000,1000]m$, as shown in Figure 6. The duration of the scenario is $K = 100s$. All targets travel in straight paths and with different but constant velocities. The number of targets is time varying due to births and deaths. There is a crossing of 3 targets at the origin at time $k = 20$, and a crossing of two pairs of targets at position $(\pm 300, 0)$ at time $k = 40$. The targets also become more dispersed as the time increases in order to elucidate the estimation errors caused by the spooky effect.

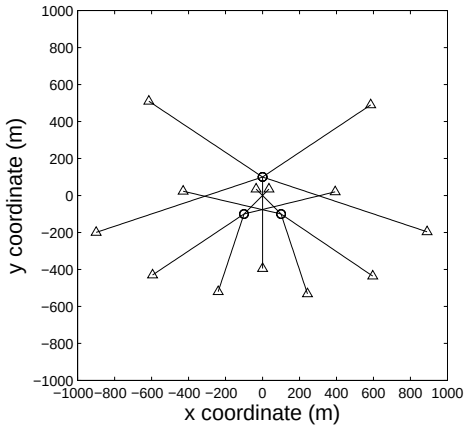

Fig. 6. Multiple trajectories in the xy plane. Start/Stop positions for each track are shown with ○/△. Targets become more dispersed with increasing time.

The kinematic target state is a vector of planar position and velocity $x_k = [\ p_{x,k}, p_{y,k}, \dot{p}_{x,k}, \dot{p}_{y,k}\ ]^T$. Measurements are noisy vectors of planar position only $z_k = [\ z_{x,k}, z_{y,k}\ ]^T$. The single-target state space model is linear Gaussian according to transition density $f_{k|k-1}(x_k|x_{k-1}) = \mathcal{N}(x_k; F_k x_{k-1}, Q_k)$ and likelihood $g_k(z_k|x_k) = \mathcal{N}(z_k; H_k x_k, R_k)$ with parameters

$$F_k = \begin{bmatrix} I_2 & \Delta I_2 \\ 0_2 & I_2 \end{bmatrix} \quad Q_k = \sigma_\nu^2 \begin{bmatrix} \frac{\Delta^4}{4} I_2 & \frac{\Delta^3}{2} I_2 \\ \frac{\Delta^3}{2} I_2 & \Delta^2 I_2 \end{bmatrix}$$
$$H_k = \begin{bmatrix} I_2 & 0_2 \end{bmatrix} \quad R_k = \sigma_\varepsilon^2 I_2$$

where I_n and 0_n denote the $n \times n$ identity and zero matrices respectively, $\Delta = 1s$ is the sampling period, $\sigma_\nu = 5m/s^2$

and $\sigma_\varepsilon = 10m$ are the standard deviations of the process noise and measurement noise. The survival probability is $p_{S,k} = 0.99$ and the birth model is a Labeled Multi-Bernoulli RFS with parameters $\pi_B = \{r_B^{(i)}, p_B^{(i)}\}_{i=1}^{3}$ where $r_B^{(i)} = 0.04$ and $p_B^{(i)}(x) = \mathcal{N}(x; m_B^{(i)}, P_B)$ with $m_B^{(1)} = [\ 0, 0, 100, 0\ ]^T$, $m_B^{(2)} = [\ -100, 0, -100, 0\ ]^T$, $m_B^{(3)} = [\ 100, 0, -100, 0\ ]^T$, $P_B = \mathrm{diag}([\ 10,\ 10,\ 10,\ 10\ ]^T)^2$. The detection probability is $p_{D,k} = 0.88$ and clutter follows a Poisson RFS with an average intensity of $\lambda_c = 1.65 \times 10^{-5}\ m^{-2}$ giving an average of 66 false alarms per scan.

The δ-GLMB filter is capped to 10000 components and is coupled with the parallel CPHD look ahead strategy described in the previous section. The CPHD filter is similarly capped to 10000 components through pruning and merging of mixture components. Results are shown over 100 Monte Carlo trials. Figures 7 and 8 show the mean and standard deviation of the estimated cardinality versus time. Figures 9 and 10 show the Optimal Sub-Pattern Assignment (OSPA) distance [56] and its localization and cardinality components for $c = 100m$ and $p = 1$.

It can be seen that both filters estimate the target cardinality accurately, with the δ-GLMB exhibiting better estimated cardinality variance. However the δ-GLMB filter significantly outperforms the CPHD filter on the overall miss distance. Examination of the localization and cardinality components reveals that the δ-GLMB filter outperforms the CPHD filter on both components. The improved cardinality performance is attributed mainly due to a lower estimated cardinality variance. The improved localization performance is attributed to two factors: (a) the spooky effect causes CPHD filter to temporarily drop tracks which are subjected to missed detections and to declare multiple estimates for existing tracks in place of the dropped tracks, and (b) the δ-GLMB filter is generally able to better localize targets due to a more accurate propagation of the filtering density.

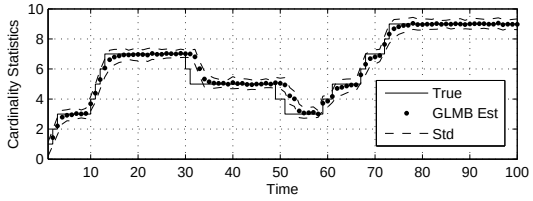

Fig. 7. Cardinality statistics for δ-GLMB filter (100 Monte Carlo trials)

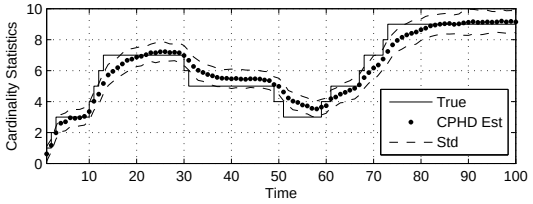

Fig. 8. Cardinality statistics for CPHD filter (100 Monte Carlo trials)

VIII. Concluding Remarks

This paper detailed the first implementation of the δ-GLMB multi-target tracking filter that is general enough to accommodate unknown and time-varying number of targets, nonlinear target dynamics, non-uniform probability of detection and clutter intensity. A salient feature of this implementation is the high parallelizability. The key innovation lies in the truncation of δ-GLMB densities without exhaustive computation of all the components and the integration of PHD lookahead to reduce the number of computations. Furthermore, it is established that truncation of a δ-GLMB density by keeping the highest weighted components minimizes the L_1-error in the multi-target densities.

It is also of interest to examine information theoretic criteria, such as the Kullback-Leibler divergence, for δ-GLMB truncation/approximation. Implementations using other multi-target filters (in place of the ranked assignment algorithm) to generate the significant δ-GLMB components can be explored to reduce numerical complexity. Approximation of δ-GLMB by other families, for example the labeled multi-Bernoulli–a GLMB with only one term [57]–is another venue for further work.

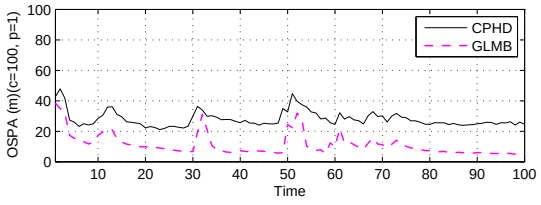

Fig. 9. OSPA distance for δ-GLMB and CPHD filters (100 Monte Carlo trials)

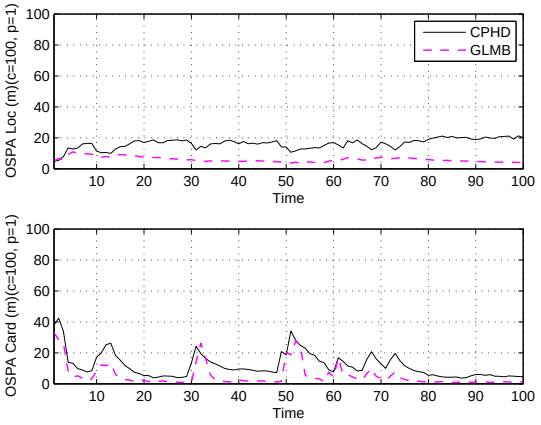

Fig. 10. OSPA components for δ-GLMB and CPHD filters (100 Monte Carlo trials)

References

[1] B.-T. Vo, and B.-N. Vo, "Labeled Random Finite Sets and multi-object conjugate priors," *IEEE Trans. Signal Processing*, Vol. 61, No. 13, pp. 3460–3475, 2013.

[2] S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*, Artech House, 1999.

[3] R. Mahler, *Statistical Multisource-Multitarget Information Fusion*, Artech House, 2007.

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

[5] D. Reid, "An algorithm for tracking multiple targets," *IEEE Trans. Automatic Control*, Vol. 24, No. 6, pp. 843–854, 1979.

[6] T. Kurien, "Issues in the design of Practical Multitarget Tracking algorithms," in *Multitarget-Multisensor Tracking: Advanced Applications*, Y. Bar-Shalom (Ed), Artech House, pp. 43–83, 1990.

[7] M. Mallick, S. Coraluppi, and C. Carthel, "Multi-target tracking using Multiple Hypothesis Tracking," in *Integrated Tracking, Classification, and Sensor Management: Theory and Applications*, M. Mallick, V. Krishnamurthy, B.-N. Vo (Eds.), Wiley/IEEE, pp. 165–201, 2012.

[8] R. Mahler, "Multi-target Bayes filtering via first-order multi-target moments," *IEEE Transactions of Aerospace and Electronic Systems*, Vol. 39, No. 4, pp. 1152–1178, 2003.

[9] 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 Procesing*, Vol. 58, No. 10, pp. 5129–5241, 2010.

[10] D. E. Clark, I. T. Ruiz, Y. Petillot, and J. Bell, "Particle PHD filter multiple target tracking in sonar image," *IEEE Trans. Aerospace & Electronic Systems*, Vol. 43, No. 1, pp. 409–416, 2007.

[11] N. T. Pham, W. Huang, and S. H. Ong, "Tracking multiple objects using Probability Hypothesis Density filter and color measurements," *IEEE Int. Conf. Multimedia and Expo*, pp. 1511–1514, July 2007,.

[12] E. Maggio, M. Taj, and A. Cavallaro,"Efficient multitarget visual tracking using random finite sets," *IEEE Trans. Circuits & Systems for Video Technology*, Vol. 18, No. 8, pp. 1016–1027, 2008.

[13] R. Hoseinnezhad, B.-N.Vo, D. Suter, and B.-T. Vo, "Visual tracking of numerous targets via multi-Bernoulli filtering of image data," *Pattern Recognition*, Vol. 45, No. 10, pp. 3625–3635, 2012.

[14] R. Hoseinnezhad, B.-N. Vo and B.-T. Vo, "Visual tracking in background subtracted image sequences via multi-Bernoulli filtering", *IEEE Trans. Signal Processing*, Vol. 61, No. 2, pp. 392–397, 2013.

[15] J. Mullane, B.-N. Vo, M. Adams, and B.-T. Vo, "A Random Finite Set approach to Bayesian SLAM," *IEEE Trans. Robotics*, Vol. 27, No. 2, pp. 268–282, 2011.

[16] F. Zhang, H. Stahle, A. Gaschler, C. Buckl, and A. Knoll, "Single camera visual odometry based on Random Finite Set Statistics," *IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS)*, pp. 559–566, Oct. 2012.

[17] D. Moratuwage, B.-N. Vo, D. Wang, and H. Wang, "Extending Bayesian RFS SLAM to multi-vehicle SLAM", *12th Int. Conf. Control, Automation, Robotics & Vision*, (ICARCV'12), Guangzhou, China, Dec. 2012.

[18] D. Moratuwage, D. Wang, and B.-N. Vo, "Collaborative multi-vehicle SLAM with moving object tracking", *IEEE Int. Conf. Robotics & Automation*, (ICRA'13), Karlsruhe, Germany, May 2013.

[19] C.-S. Lee, D.E. Clark, and J. Salvi, "SLAM with dynamic targets via single-cluster PHD filtering," *IEEE Journal of Selected Topics in Signal Processing*, Vol. 7, No. 3, pp. 543–552, 2013.

[20] G. Battistelli, L. Chisci, S. Morrocchi, F. Papi, A. Benavoli, A. D. Lallo, A. Farina, and A. Graziano, "Traffic intensity estimation via PHD filtering," *5th European Radar Conf.*, Amsterdam, The Netherlands, pp. 340–343, Oct. 2008.

[21] L. Mihaylova, "Probability Hypothesis Density filtering for real-time traffic state estimation and prediction," *Network and Heterogeneous Media. An Applied Mathematics Journal*, 2013.

[22] D. Meissner, S. Reuter, and K. Dietmayer, "Road user tracking at intersections using a multiple-model PHD filter," *IEEE Intelligent Vehicles Symposium* (IV), pp. 377–382, June 2013.

[23] R. Juang, A. Levchenko, and P. Burlina, "Tracking cell motion using GM-PHD," *Int. Symp. Biomedical Imaging*, pp. 1154–1157, June 2009.

[24] S. H. Rezatofighi, S. Gould, B.-N. Vo, K. Mele, W. Hughes, and R. Hartley, "A multiple model Probability Hypothesis Density tracker for time-lapse cell microscopy sequences," *Int. Conf. Inf. Proc. Medical Imaging* (IPMI), May 2013.

[25] X. Zhang, "Adaptive control and reconfiguration of mobile wireless sensor networks for dynamic multi-target tracking," *IEEE Trans. Automatic Control,* Vol. 56, No. 10, pp. 2429–2444, 2011.

[26] J. Lee and K. Yao, "Initialization of multi-Bernoulli Random Finite Sets over a sensor tree," *Int. Conf. Acoustics, Speech & Signal Processing* (ICASSP), pp. 25–30, Mar. 2012.

[27] G. Battistelli, L. Chisci, C. Fantacci, A. Farina, and A. Graziano, "Consensus CPHD filter for distributed multitarget tracking," *IEEE Journal on Selected Topics in Signal Processing*, Vol. 7, No. 3, pp. 508–520, 2013.

[28] M. Uney, D.E Clark, S.J. Julier, "Distributed fusion of PHD filters via exponential mixture densities," *IEEE Journal on Selected Topics in Signal Processing*, Vol. 7, No. 3, pp. 521–531, 2013.

[29] R. Mahler, "PHD filters of higher order in target number," *IEEE Trans. Aerospace & Electronic Systems*, Vol. 43, No. 3, pp. 1523–1543 2007.

[30] B.-T. Vo, B.-N. Vo, and A. Cantoni, "The cardinality balanced multi-target multi-Bernoulli filter and its implementations," *IEEE Trans. Signal Processing*, Vol. 57, No. 2, pp. 409–423, 2009.

[31] B.-T. Vo, and B.-N. Vo, "A Random Finite Set conjugate prior and application to multi-target tracking," Proc. 7th Int. *Conf. Intelligent Sensors, Sensor Networks & Information Processing* (ISSNIP'2011), Adelaide, Australia, Dec. 2011.

[32] D.J. Daley and D. Vere-Jones, *An Introduction to the Theory of Point Processes*, Springer, New York, 1988.

[33] D. Stoyan, W., Kendall, and J. Mecke *Stochastic geometry and its applications* (2nd Ed). Chichester: Wiley, 1995.

[34] B.-T. Vo, *Random Finite Sets in Multi-Object Filtering*, PhD Thesis, University of Western Australia, 2008.

[35] B. Ristic, *Particle Filters for Random Set Models*, Springer, 2013.

[36] H. Kuhn, "The Hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, Vol. 2, pp. 83–97, 1955

[37] K. G. Murty, "An algorithm for ranking all the assignments in order of increasing cost," *Operations Research*, Vol. 16, No. 3, pp. 682–687, 1968.

[38] J. Munkres, "Algorithms for assignment and transportation problems," *Journal of the Society for Industrial and Applied Mathematics*, Vol. 5, No. 1, Mar. 1957.

[39] R. Jonker, and T. Volgenant, "A shortest augmenting path algorithm for dense and sparse linear assignment problems," *Computing* Vol. 38, No. 11, pp. 325–340, Nov. 1987.

[40] R. Danchick and G. E. Newman, "A fast method for finding the exact N-best hypotheses for multitarget tracking", *IEEE Trans. Aerospace & Electronic Systems*, Vol. 29, No. 2, pp. 555–560, 1993.

[41] I. Cox, and M. Miller, "On finding ranked assignments with application to multitarget tracking and motion correspondence," *IEEE Trans. Aerospace & Electronic Systems*, Vol. 32, No. 1, pp. 486–489, 1995.

[42] I. Cox, and S. Hingorani "An efficient implementation of Reid's multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking," *IEEE Trans. Pattern Analysis & Machine Intelligence*, Vol. 18, No. 2, pp. 138–150, 1996.

[43] K. R. Pattipati, R. L. Popp, and T. T. Kirubarajan, "Survey of Assignment Techniques for Multitarget Tracking," in *Multitarget-Multisensor Tracking: Applications and Advances*, Vol. 3, Y. Bar-Shalom and D. Blair (Eds.), Artech House, pp. 77–159, 2000.

[44] M. Miller, H. Stone, and I. Cox, "Optimizing Murty's ranked assignment method," *IEEE Trans. Aerospace & Electronic Systems*, Vol. 33, No. 3, pp. 851–862, 1997.

[45] M. Pascoal, M. Captivo, and J. Clımaco, "A note on a new variant of Murty's ranking assignments algorithm," *4OR: Quarterly Journal of the Belgian, French and Italian Operations Research Societies*, Vol. 1, No. 3, pp. 243–255, 2003.

[46] C. Pedersen, L. Nielsen, and K. Andersen, "An algorithm for ranking assignments using reoptimization, " *Computers & Operations Research*, Vol. 35, No. 11, pp. 3714–3726, 2008.

[47] D. Eppstein, "Finding the k shortest paths," *SIAM Journal on computing*, Vol. 28, No. 2, pp. 652–673, 1998.

[48] R. Bellman, "On a routing problem," *Quarterly of Applied Mathematics*, Vol. 16, pp. 87–90, 1958

[49] L. R. Ford, and D. R. Fulkerson, *Flow in Networks*, Princeton University Press, 1962.

[50] B.-N. Vo, S. Singh and A. Doucet, "Sequential Monte Carlo methods for multi-target filtering with Random Finite Sets," *IEEE Trans. Aerospace & Electronic Systems*, Vol. 41, No. 4, pp. 1224–1245, 2005.

[51] B.-N. Vo and W.-K. Ma, "The Gaussian mixture Probability Hypothesis Density filter", *IEEE Trans. Signal Processing*, Vol. 54, No. 11, pp. 4091–4104, 2006.

[52] B.-T. Vo, B.-N. Vo, and A. Cantoni, "Analytic implementations of the Cardinalized Probability Hypothesis Density filter," *IEEE Trans. Signal Processing*, Vol. 55, No. 7, pp. 3553–3567, 2007.

[53] A. Doucet, S. J. Godsill, and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," *Stat. Comp.*, Vol. 10, pp. 197–208, 2000.

[54] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman filter: Particle Filters for Tracking Applications*, Artech House, 2004.

[55] D. Franken, M. Schmidt, and M. Ulmke, "Spooky action at a distance in the Cardinalized Probability Hypothesis Density filter," *IEEE Trans. Aerospace and Electronic Systems*, Vol. 45, No. 4, pp. 1657–1664, 2009.

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

[57] S. Reuter, B.-T. Vo, B.-N. Vo, and K. Dietmayer, "The labelled multi-Bernoulli filter," *IEEE Trans. Signal Processing*, Vol. 62, No. 12, pp. 3246–3260, 2014.