SAND2018-12052C

Current Challenges in Labeled Random Finite Set Based Distributed Multi-sensor Multi-object Tracking

Augustus Buonviri, Matthew York, Keith LeGrand, and James Meub
Sandia National Laboratories
1515 Eubank Blvd. SE
Albuquerque, New Mexico 87123
apbuonv@sandia.gov, myork@sandia.gov, kalegra@sandia.gov, and jhmeub@sandia.gov

Abstract— **In recent years, increasing interest in distributed sensing networks has led to a demand for robust multi-sensor multi-object tracking (MOT) methods that can take advantage of large quantities of gathered data. However, distributed sensing has unique challenges stemming from limited computational resources, limited bandwidth, and complex network topology that must be considered within a given tracking method. Several recently developed methods that are based upon the random finite set (RFS) have shown promise as statistically rigorous approaches to the distributed MOT problem. Among the most desirable qualities of RFS-based approaches is that they are derived from a common mathematical framework, finite set statistics, which provides a basis for principled fusion of full multi-object probability distributions. Yet, distributed labeled RFS tracking is a still-maturing field of research, and many practical considerations must be addressed before large-scale, real-time systems can be implemented. For example, methods that use label-based fusion require perfect label consistency of objects across sensors, which is impossible to guarantee in scalable distributed systems. This paper discusses the significant challenges that distributed tracking using labeled RFS methods brings. An overview of labeled RFS filtering is presented, the distributed MOT problem is characterized, and recent approaches to distributed labeled RFS filtering are examined. The problems that currently prevent implementation of distributed labeled RFS trackers in scalable real-time systems are identified and demonstrated within the scope of several exemplar scenarios.**

Table of Contents

978-1-5386-6854-2/19/$31.00 ©2019 IEEE

1. Introduction

Multi-sensor multi-object tracking (MOT) is the problem of tracking an unknown, time-varying number of partially observable objects using the noise corrupted measurements produced by multiple sensors. Much of the recent research on this topic has been dedicated to the application of Mahler's finite set statistics (FISST) and random finite set (RFS) theory to the multi-sensor MOT problem [1], [2], [3], [4], [5]. The RFS, which is both random in value and in cardinality, can naturally model multi-object states and multi-object observations. Leveraging the RFS and the associated FISST multi-object calculus, the complexities of MOT are reduced to mathematically-principled operations on multi-object density functions, which share many of the properties of single-object density functions, such as non-negativity and integration to unity.

In general, approaches to multi-sensor MOT fall into two categories: centralized and distributed. The advantages and disadvantages of centralized and distributed architectures are well-documented [6], [7], [8], [9]. The key advantage of centralized architectures is that they can achieve optimal performance in estimation accuracy by using observations from all sensors. One such example is the multi-sensor δ-generalized labeled multi-Bernoulli (δ-GLMB) filter, which processes all sensor data collectively in a multi-sensor multi-object Bayes update [4], [10]. On the other hand, a key disadvantage of centralized architectures is their poor computational scalability, specifically in regard to the multi-sensor measurement-to-track data association problem, which is NP-hard [10].

By contrast, distributed architectures are inherently sub-optimal in estimation accuracy but offer better system scalability by distributing tracking tasks to local sensors, the outputs of which are later combined through multi-sensor fusion techniques. The application of labeled RFS theory in this domain is extremely promising, in part because principled fusion techniques such as covariance intersection can be generalized to multi-object distributions [11]. However, a number of key challenges still need to be addressed before large-scale distributed labeled RFS systems can be implemented. For example, sensing networks in which sensors observe different regions introduce diversity in multi-object density supports, which has yet to be effectively dealt with by distributed labeled RFS fusion strategies.

This paper provides an investigation of these issues with the intent of summarizing the requirements that any distributed labeled RFS tracking algorithm will need to satisfy to enable practical implementation in modern sensing networks. This paper is organized as follows: Section 2 gives a brief background of labeled RFS filtering, Section 3 provides an overview of the distributed multi-sensor MOT problem, Section 4 outlines current distributed labeled RFS fusion strategies, Section 5 explores the challenges of distributed labeled RFS fusion, and Section 6 summarizes key takeaways.

2. Overview of Labeled RFS Filtering

The standard notation conventions of labeled RFS filtering literature are adopted for this work. Single-object and multi-object states are represented by lowercase $(x, \mathbf{x})$ and uppercase letters $(X, \mathbf{X})$, respectively. Labeled states and functions are distinguished from their unlabeled counterparts by bold $(\mathbf{x}, \mathbf{X}, \boldsymbol{\pi})$ symbols. Spaces are represented in blackboard bold $(\mathbb{X}, \mathbb{Z})$, and the class of all finite subsets of a space $\mathbb{X}$ is denoted $\mathcal{F}(\mathbb{X})$.

The *multi-object exponential* is defined as $f^X \triangleq \prod_{x \in X} f(x)$, where $f^{\emptyset} \triangleq 0$. The set-generalized Kronecker delta is defined as

$$\delta_A(B) = \begin{cases} 1, & \text{if } A = B \\ 0, & \text{otherwise} \end{cases} .$$

A labeled state $\mathbf{x} = (x, \ell)$ on $\mathbb{X} \times \mathbb{L}$ consists of a kinematic state $x \in \mathbb{X}$ and label $\ell \in \mathbb{L}$. The label portion of a labeled state can be recovered by the projection $\mathcal{L}(\mathbf{x}) : \mathbb{X} \times \mathbb{L} \to \mathbb{L}$, where $\mathcal{L}(\mathbf{x}) = \mathcal{L}(x, \ell) \triangleq \ell$.

RFSs and Labeled RFSs

RFSs are sets whose cardinalities and values are random. Formally, a RFS X defined on a space $\mathbb{X}$ is a set-valued random variable with realizations in $\mathcal{F}(\mathbb{X})$. A labeled RFS $\mathbf{X}$ defined on $\mathbb{X} \times \mathbb{L}$, where $\mathbb{X}$ is the single-object state space and $\mathbb{L}$ is the discrete label space, is a RFS whose states are appended with labels, such that it has realizations in $\mathcal{F}(\mathbb{X} \times \mathbb{L})$. The projection $\mathcal{L}(\cdot)$ is defined for labeled RFSs as $\mathcal{L}(\mathbf{X}) \triangleq \{\mathcal{L}(\mathbf{x}) : x \in \mathbf{X}\}$, and the cardinality of any RFS X is denoted as $|X|$.

The Bayes Multi-object Filter

The Bayes multi-object filter involves recursively refining information about an evolving set of partially observable hidden states, which are collectively referred to as a *multi-object state*. This process may be logically separated into two parts: prediction and update.

In prediction, a posterior π on the multi-object state X is propagated forward to produce a prior $\bar{\pi}$ on the state X_+, where the subscript "+" denotes the next time of the filtering recursion. This is accomplished through use of the multi-object Chapman-Kolmogorov equation

$$\bar{\pi}_+(X_+) = \int f(X_+|X)\pi(X)\delta X \,, \tag{1}$$

where $f(X_+|X)$ is the standard multi-object Markov transition density [8], which accounts for thinning, Markov shift, and superposition. Note that Equation (1) involves the use of the set integral [8], which is defined as

$$\int f(X)\delta X = \sum_{i=0}^{\infty} \frac{1}{i!} \int_{\mathbb{X}^i} f(x_1, \ldots, x_i) d(x_1, \ldots, x_i) \,.$$

In the measurement update, the prior density is refined using the superposition of object measurements and clutter Z_+ to produce the posterior π_+. This is accomplished through application of Bayes' rule,

$$\pi_+(X_+) = \frac{g(Z_+|X_+)\bar{\pi}_+(X_+)}{\int g(Z_+|X_+)\bar{\pi}_+(X_+)\delta X}$$

where $g(Z_+|X_+)$ is the standard multi-object likelihood function [12].

Labeled RFS Densities

Three common families of labeled RFS distributions are considered in this work, namely, the δ-GLMB, marginalized δ-generalized labeled multi-Bernoulli (Mδ-GLMB), and labeled multi-Bernoulli (LMB). The most generalized of these distributions, the δ-GLMB, is a special case of the generalized labeled multi-Bernoulli (GLMB) labeled RFS family and is characterized by its density function

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

where Ξ is the space of all track-to-measurement association histories ξ, $I \in \mathcal{F}(\mathbb{L})$ is a set of object labels, $\omega^{(I,\xi)}$ is the normalized non-negative weight of a pair (I, ξ), $p^{(\xi)}(x, \ell)$ is the single-object density of the object with label ℓ, and $\Delta(\mathbf{X})$ is the distinct label indicator defined as $\Delta(\mathbf{X}) = \delta_{|\mathbf{X}|}(|\mathcal{L}(\mathbf{X})|)$. The label set / association history pair (I, ξ) is called a *hypothesis*, and the associated weight $\omega^{(I,\xi)}$ can be interpreted as the probability of that hypothesis. The δ-GLMB has special significance among the RFSs considered in this work, as it enables tractable, Bayes-optimal multi-object filtering by virtue of being conjugate prior with respect to the standard multi-object likelihood function and closed under the multi-object Chapman-Kolmogorov equation [12].

The Mδ-GLMB density is also a special case of the GLMB. Mδ-GLMBs densities may be formed by marginalizing δ-GLMBs densities along distinct label sets. The Mδ-GLMB density function is given as

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

where $\omega^{(I)}$ is the normalized non-negative weight of a label set I and $p^{(I)}(x, \ell)$ is the single-object density of the target with label ℓ. Here, a hypothesis is equivalent to a unique label set I.

Whereas GLMB densities may be thought of as "hypothesis-oriented," LMB, densities are parameterized by inidivual object identities, and thus can be thought of as being "track-oriented." The LMB density takes the form

$$\boldsymbol{\pi}(\mathbf{X}) = \Delta(\mathbf{X}) \left[1 - r\right]^{\mathbb{L} - \mathcal{L}(\mathbf{X})} \left[r\right]^{\mathcal{L}(\mathbf{X})} \left[p^{(I)}\right]^{\mathbf{X}} , \tag{2}$$

where $r^{(\ell)}$ and $p^{(\ell)}(x)$ are the probability of existence and density function of an object with label ℓ, respectively. Although Mδ-GLMB and LMB filters are not Bayes-optimal, these approximate forms require less computational resources and, in many cases, offer estimation performance comparable to the full GLMB filter. For a more detailed exposition of the δ-GLMB, Mδ-GLMB, and LMB filters, the reader is referred to [12], [13], and [14], respectively.

Birth Models

Of special relevance to distributed labeled RFS filtering is the issue of object birth. In the standard multi-object Bayes filter, object birth is modeled as part of the multi-object transition density function. The density of birth objects can be written in GLMB form as

$$\mathbf{f}_B(\mathbf{X}) = \Delta(\mathbf{X}) w_B(\mathcal{L}(\mathbf{X})) \left[p_B\right]^{\mathbf{X}} ,$$

where $w_B(L)$ is the probability of the birth label set L and $p_B(x, \ell)$ is the single-object density of the birth object with label ℓ. A common assumption made in recent RFS literature is that target birth follows a LMB distribution of the form of Equation (2), which is a special case of the GLMB distribution. The LMB distribution is parameterized by individual object probabilities of existence and single-object density functions. While this target-wise parameterization fits neatly into data-association problems due to its object-wise cost decomposition, it can prove cumbersome when trying to formulate an object birth model with little prior information. Alternatively, a measurement-driven birth model can be used, which initializes new objects from measurement data and requires less prior information [15], [14], [16], [17]. In principle, these methods reduce the computational and memory demands of modeling birth in complex scenarios and thus improve filter scalability.

Label Convention

Multiple valid conventions are available for track labeling. The simplest method is to treat track labels as integers: e.g. $\mathbb{L}_k = \{1, 2, 3\}$ and $\mathbb{B}_k = \{4, 5, 6\}$. Using this convention, special care must be taken to ensure that persistent object labels and birth object labels are distinct at every time step; i.e. $\mathbb{L}_k \cap \mathbb{B}_k = \emptyset$. Satisfying this condition is made easier by instead treating a label as a tuple $\ell = (k, i)$, where the first tuple element k is time of object birth and the second element i is an integer used to distinguish objects born at the same time.

3. The Distributed Multi-sensor MOT Problem

Distributed multi-sensor MOT is a generalization of MOT to cases where two or more entities, known as *fusion centers*, each maintain track data about the current multi-object state. These data are referred to as *local* with respect to the owning fusion center. Fusion centers communicate data to one another and use fusion to combine local data with those received from from other centers. Through the communications links between them, multiple fusion centers form a *sensing network* and each center constitutes a *node* in that network. In principle, by fusing information from multiple sources nodes of a sensing network produce track data that have lower estimation error than the track data that could be produced from the information of any individual source.

Within a network, the organization of communications links between nodes is referred to as a *topology*. Network topologies can vary widely depending on the application, but the topologies considered in this work are *fully distributed* and *dynamic*. A network is fully distributed when its topology lacks organization; i.e. any node may communicate with one or more other nodes and these communications links may be unidirectional or bidirectional, as illustrated in Figure 1. A network is dynamic when its topology may change over time, where such changes may be due to node failures, node dynamics, and commanded changes. Fully distributed and dynamic sensing networks are generally robust against failures of individual nodes and enable each node to potentially have access to all information collected by the network. The former quality implies fault tolerance of the system, while the latter enables each node to perform more informed decision making based on the multi-object state of interest. Fully distributed and dynamic sensing networks, which will herein be referred to as distributed for compactness, are thus well suited for a variety of applications including space situational awareness [18], [19], wide-area surveillance [20], and traffic monitoring [21].

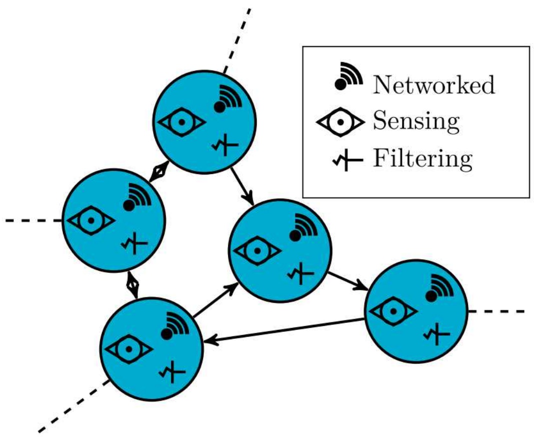


Figure 1. Example of a fully distributed network topology.

It should not be understated the advantage that labeled RFS filtering brings to the distributed MOT problem. The ability to fully represent a multi-object state as a multi-object probability density imparts on labeled RFS filters a quality unique among MOT methods: the filtering product can be fused in a mathematically principled way. Traditional track-to-track fusion techniques can be readily applied to labeled RFS estimates. However, unlike traditional methods, labeled RFS theory enables a more principled fusion strategy through the combination of multi-object densities, which promises greater performance.

4. Current Methods for Distributed Labeled RFS Tracking

Optimal Bayesian fusion of multi-object posterior densities is known to be of the form

$$\boldsymbol{\pi}\left(\mathbf{X}|\bigcup_{s=1}^{\mathcal{S}} Z^{(s)}\right) \propto \frac{\prod_{s=1}^{\mathcal{S}} \boldsymbol{\pi}^{(s)}(\mathbf{X}|Z^{(s)})}{\boldsymbol{\pi}\left(\mathbf{X}|\bigcap_{s=1}^{\mathcal{S}} Z^{(s)}\right)},$$

where $\mathcal{S}$ is the total number of sensors, $Z^{(s)}$ is the measurement RFS generated by a sensor s, and $\boldsymbol{\pi}^{(s)}$ is the posterior density maintained by sensor s. However, the mutual information $\boldsymbol{\pi}(\mathbf{X}|\bigcap_{s=1}^{\mathcal{S}} Z^{(s)})$ of the posteriors grows infinitely over time if successive fusion is performed and thus becomes intractable to track, share, and utilize. When the mutual information is not accounted for, fusion methods can be susceptible to the error known as *double counting* in which fusion will produce estimates that are overly confident due to correlations in the fused posteriors. As such, robustness to double counting is a well-known prerequisite for any functional distributed fusion strategy [7], [22], [23].

One of the key advancements in the development of RFS fusion strategies was made in 2015 with the proof that a specific fusion method for multi-object densities is inherently immune to double counting [24]. Unfortunately, there is wide disagreement in the literature about what to call this form of fusion. In various places, it is referred to as Kullback-Leibler averaging (KLA) [24], [25], generalized covariance intersection (GCI) [11], [26], Chernoff fusion [27], [28], exponential mixture density (EMD) fusion [5], [29], logarithmic opinion pooling [30], and geometric mean density (GMD) fusion [31]. To avoid confusion, this work uses the term GMD fusion, as it is highly general, does not suggest specific underlying principles, emphasizes its relation to the simpler and more familiar concept of geometric averaging, and highlights its parallel to the method of arithmetic mean density (AMD) fusion. Regardless of name, GMD is among the earliest principled distributed fusion strategies applied to labeled RFSs, and it has the weighted form

$$\boldsymbol{\pi}(\mathbf{X}) \propto \prod_{s=1}^{\mathcal{S}} \left(\boldsymbol{\pi}^{(s)}(\mathbf{X})\right)^{w^{(s)}},$$

where the $w^{(s)}$ are normalized, non-negative weights and conditioning of the posterior densities on measurement information is implied. Closed form weighted GMD fusion rules have been developed for both LMB and Mδ-GLMB densities [25]. These rules perform fusion by fusing density components with matching labels in the LMB case and components with matching label sets in the Mδ-GLMB case. As such, the rules assume perfect label consistency (that the same labels correspond to the same objects) across posteriors. It will be explained in Section 5 that it is difficult to satisfy this assumption in practice, and it has been shown in [11] that failure to ensure label consistency results in a breakdown of direct GMD fusion.

As stated, the counterpart of GMD fusion is AMD fusion, which has the weighted form

$$\boldsymbol{\pi}(\mathbf{X}) = \sum_{s=1}^{\mathcal{S}} w^{(s)} \boldsymbol{\pi}^{(s)}(\mathbf{X}). \tag{3}$$

Just as GMD fusion is an extension of the concept of geometric averaging to probability densities, AMD fusion is an extension of the concept of arithmetic averaging. As may be inferred from Equation 3, direct AMD fusion tends to be computationally cheaper and facilitates easier implementation when compared to GMD fusion. AMD fusion has been shown to be more robust to missed detections and low signal-to-noise ratios in unlabeled RFS distributed filtering [32].

In order to relax the label consistency requirement of GMD fusion of labeled RFS densities, it is possible to perform fusion on the corresponding unlabeled densities. The method of [11] makes use of multi-Bernoulli (MB) RFSs, which are the unlabeled equivalent of LMBs [14], for this purpose. In short, this method involves approximating δ-GLMB distributions as MB distributions, fusing the MB distributions, and then reconstructing a δ-GLMB distribution, where the approximation and reconstruction preserve first-order moments. During the approximation, the relation of labels to their states is computed and represented in a probabilistic manner. It is then used to re-associate labels with the unlabeled density's components after fusion. As a result, this method requires that there be no ambiguity about which Bernoulli component of the unlabeled MB distribution a label is associated with, i.e. tracks must be well separated.

As demonstrated in [33], it is possible to fuse track-to-measurement data association information, as opposed to posterior densities. This method relies on the sharing of cross-entropy information associated with measurements at different nodes. It is the only existing fusion method known to the authors to enable distributed δ-GLMB filtering where the only approximation made is the truncation of the filtering density. This method makes a similar assumption to the discussed GMD fusion rules [25] in that it assumes the δ-GLMB priors at each node share identical hypotheses and labels.

The labeled RFS fusion methods discussed within this section each operate within a consensus framework [11], [25], [33]. Fundamentally, networks operating under the principle of consensus reach a global agreement on the output of some process by having each node iteratively share and fuse its local output in such a way that the fusion product is increasingly similar across all properly functioning nodes. This is illustrated in Figure 2, where four networked nodes iteratively share and average a set of local values. In this example, the obvious solution to the averaging problem is five, and it is clear that all local values in the network converge to this value as the number of iterations grows. In consensus networks, new and updated information is propagated through the network by the iterative sharing and fusion of information between network neighbors. Because nodes process data locally, the computational workload of the global problem is distributed across the network.

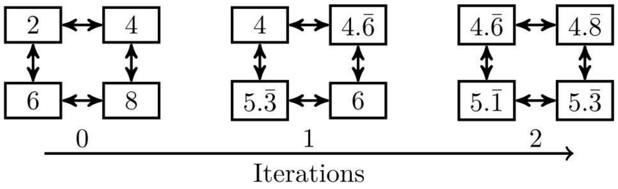


Figure 2. Example consensus network consisting of four nodes iteratively sharing and averaging local values. As the number of iterations approaches infinity, all local values converge to the true average of the original local values.

5. Current Challenges in Distributed Labeled RFS Tracking

Major strides have been made in the last several years in the development of distributed labeled RFS filtering techniques. However, several challenges must still be resolved before distributed labeled RFS tracking can be implemented in large, practical sensing networks tracking large numbers of objects. This section characterizes and explores these challenges.

Double Counting

It is possible and likely that communications loops may form in distributed networks, due to their unstructured and dynamic nature. Such a loop exists when data can be communicated from a source node to a recipient node along multiple paths. This gives rise to the well documented problem of double counting [7], [22], [23], which is illustrated in Figure 3. Double counting occurs when data arriving at a recipient from multiple paths or multiple times along the same path are fused multiple times as if they were new data. The result is that the fused data become biased and corrupted. Robustness against double counting is thus necessary in distributed networks, and modern fusion strategies are immune to double counting [11], [23], [24], [25].

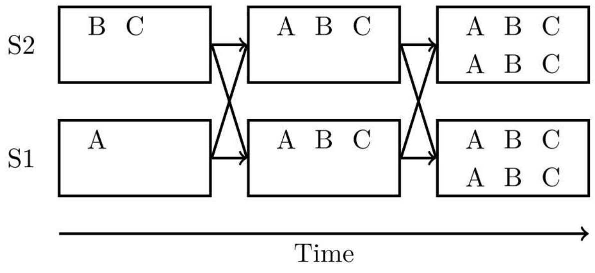
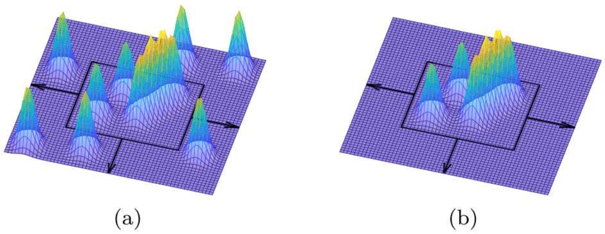


Figure 3. The double counting phenomenon in a two-sensor network. S1 and S2 denote Sensor One and Sensor Two, respectively. The letters represent distinct track data. Data is shared and combined by the sensors at each time step, and both sensors develop duplicates of the original track data as a result.

Label Consistency and Object Birth Models

One of the strongest assumptions of current direct GMD fusion methods is that all sensors share an identical object label space [25]. The most straightforward, although impractical, approach to guarantee an identical label space is to enforce a single, global, predefined object birth model across all sensors. In theory, the use of an identical birth model across all sensors ensures that the same potential birth labels are added to each sensor's local distribution at each time step. Unfortunately, the use of a global birth density across all sensor nodes is practical only for the simplest systems due to issues related to network size/geographic scalability, label-to-data assignment consistency, and systems interoperability.

Increases in geographic coverage generally result in increased birth object distribution complexity, as shown in Figure 4a. The birth distribution can remain unchanged as coverage increases, as in Figure 4b, but this restricts the formation of new tracks to a specific region of the state space and is not well-suited for many tracking applications. Increased birth distribution complexity is typically not a concern for centralized filtering, but becomes problematic for distributed systems that require all sensors to propagate and maintain the distribution. Clearly it is a waste of resources to propagate and maintain birth distribution components that lack statistical relevance to a sensor's field of view (FoV) or field of regard (FoR). A more sensible and scalable approach is for individual sensors to only model object birth in their local sensing region. However, label inconsistencies inevitably develop when different birth models are used across a network. Moreover, the supports of the multi-object densities of different sensors are inherently dissimilar when the regions in which those sensors model birth are dissimilar. Distribution support diversity introduces complications beyond label consistency, and is discussed later in this section.


(a)

(b)

Figure 4. Illustration of a birth distribution either increasing in complexity (a) or remaining unchanged (b) as geographic coverage increases.

For systems where geographic coverage is not a concern, object birth can still present issues if the birth distribution is highly complex within nodes' local sensing regions. Scalability in such scenarios is achieved in centralized systems through use of measurement-driven birth models [15], [14], [16], [17]. This is because the scalability of such models is generally limited by the number of object births and clutter returns each frame, not birth density complexity. However, these methods inevitably lead to label inconsistencies between nodes due to their adaptivity to local data.

Even when using a unified birth model with consistent labels, there is no guarantee that labels will be assigned to objects in a consistent fashion. At a local level, many options exist for assigning available birth object labels to local measurement data. To illustrate this, consider the scenario depicted in Figure 5, in which each node measures two potential birth objects in the Rio Grande river. The birth object density is assumed to

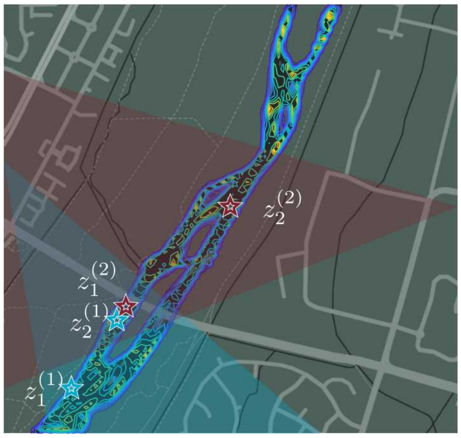


Figure 5. Example scenario using two sensors to track debris in the Rio Grande river. Sensor 1's FoV and detections are represented by the blue shaded region and blue stars, respectively. Sensor 2's FoV and detections are represented by the red shaded region and red stars, respectively. The single-object density $p_B(x, \ell)$ is depicted by the contour lines and is assumed identical for all birth object labels $\mathbb{B}_k = \{(k, 1), (k, 2), (k, 3)\}$.

be LMB with a label space $\mathbb{B}_k = \{(k, 1), (k, 2), (k, 3)\}$ and $(r_B^{(k,1)}, p_B^{(k,1)}) = (r_B^{(k,2)}, p_B^{(k,2)}) = (r_B^{(k,3)}, p_B^{(k,3)})$. Assuming no clutter and no prior objects, any given measurement is equally probable to have originated from each birth object label $(k, 1)$, $(k, 2)$, and $(k, 3)$. In order to guarantee that densities with consistent labeling are defined, a total of thirteen label-to-measurement assignment possibilities must be considered by each sensor. This number grows significantly when more object births must be considered and/or more measurements are received. In fact, for the case of no prior objects and no clutter, the number of unique assignments is given by

$$N_{\text{possible}} = 1 + \sum_{M=\max(|\mathbb{B}|-|Z|,0)}^{|\mathbb{B}|-1} \frac{|\mathbb{B}|!|Z|!}{M!(|\mathbb{B}|-M)!(|Z|-|\mathbb{B}|+M)!}. \quad (4)$$

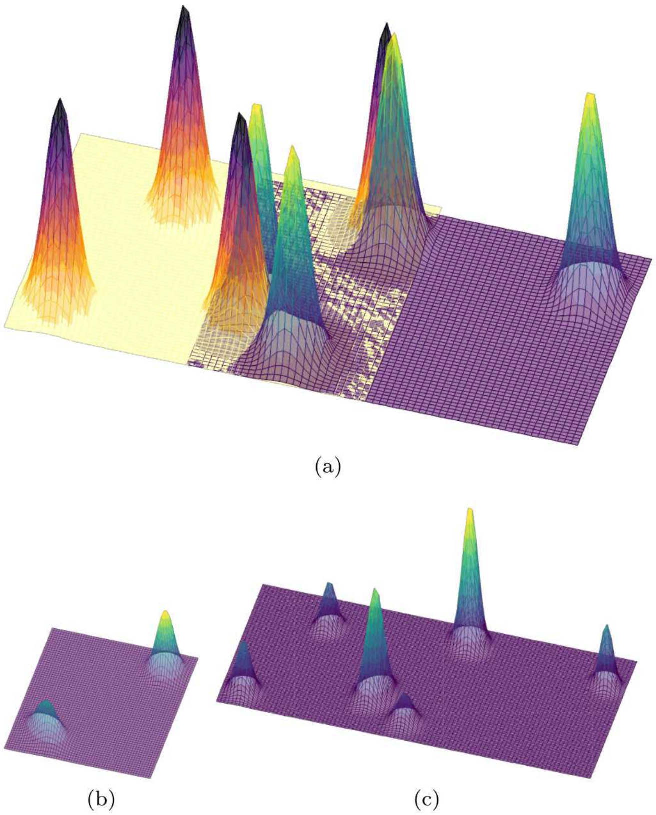
Equation (4) clearly demonstrates the lack of scalability of this approach. For instance, for $|Z| = |\mathbb{B}| = 7$, there are $130,922$ birth label assignment possibilities that must be considered to guarantee labeling consistency with other sensors, and this is further complicated when clutter and persistent objects exist.

The global birth density requirement needed to satisfy label space consistency is specifically challenging for ad-hoc sensor networks. The ability for sensing systems to dynamically form, task, and fuse disjoint sensor data is critical for responding to new mission priorities, exercising alternate sensing phenomenologies against new object types, and incrementally increasing the sensor network performance. Systems engineering focus has been on open systems architectures, which define the physical and data interfaces between sensing nodes within an arbitrary network. A single global birth model may not be defined for all mission phases of a given group of sensing systems. Fusion algorithms that are agnostic to the filtering recursions should be used to enable interoperability and label consensus across these ad-hoc RFS-based sensing networks.

Distribution Support Diversity

For practical reasons, single-sensor tracking systems do not typically maintain track solutions outside their FoV or FoR. In a computationally-limited system, it is a natural decision to prioritize maintenance of object tracks that can be sensed and limit others through measurement-driven birth models and track termination of objects leaving a sensor's FoR. Employed across a distributed network of sensors with different and potentially time-varying FoRs, this practice presents challenges to RFS fusion, as the multi-object density supports will inevitably vary across sensors. Simply put, Sensor A's solution's may imply zero probability of object existence in Sensor B's FoR simply because Sensor A has never accessed that area. This diversity in density support is problematic for both GMD and AMD fusion methods, as illustrated in Figure 6.

As shown in Figure 6b, GMD fusion results in a density where all tracks that fall outside the shared support region are dropped. AMD fusion, on the other hand, preserves these tracks, yet penalizes them in the same manner it would for missed detections, as shown in Figure 6c.



(a)

(b)

(c)

Figure 6. Illustration of fusion using two distributions (a) with different supports. Using GMD fusion (b), object tracks not common to both sensors are discarded. Using AMD fusion (c), object tracks not common to both sensors are significantly down-weighted.

Detection and Clutter Variation

Due to stochastic variations in detection and clutter, a track associated with a given object may be produced before or after the object appears, as demonstrated in Figure 7. If an object is misdetected on its first appearance, its corresponding track initialization will be delayed and its label will reflect a later time of birth. If a clutter measurement is received prior to a true object appearance that statistically agrees to the object's trajectory, a track may be initialized with a label that reflects an earlier time of birth. In Figure 7, a single object is tracked with three distinct labels by different sensors as a result of differing times of first detection and clutter. In distributed systems, these variations between nodes result in tracks nominally associated with the same object having different labels [11]. This phenomenon exacerbates the issue of support diversity and can result in significant cardinality errors when GMD or AMD fusion methods are used directly.

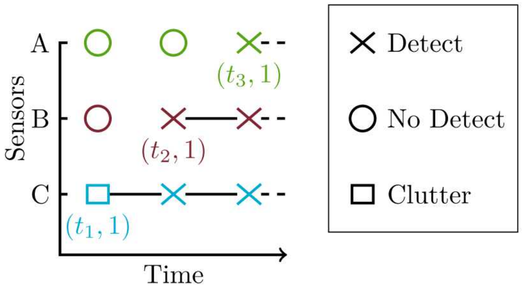


Figure 7. Illustration of detection and clutter variation causing different labels to be developed by different sensors for the same object. The object appears at time t_2. Sensor A first detects the object after its appearance due to missed detection, Sensor B detects the object at its time of appearance, and Sensor C has a clutter detection near the object's birth location immediately prior to its appearance.

In GLMB filtering, the number of possible data association histories grows exponentially in time. To maintain computational tractability, the GLMB density is truncated at each time step by dropping the lowest-weight hypotheses and renormalizing. This technique creates additional challenges in multi-sensor fusion due to variations in object detections between sensors. Sensors that detect different sets of birth objects are likely to discard hypotheses on undetected birth objects and keep hypotheses containing only those birth objects that were detected. As a result, those sensors end up maintaining hypotheses whose label sets differ only by birth track labels. Similar to the issue of support diversity, under these conditions neither direct GMD nor AMD fusion can produce a hypothesis containing all detected birth tracks, as these methods only combine hypotheses with matching label sets.

Bandwidth Constraints

Two of the key considerations in the design of multi-sensor architectures are whether to share track data or measurement data, and how often to share these data. Track data representations are naturally compressive: a density function can capture the information content of thousands of observations in a compact form. However, some number of measurements must be processed before the track data size becomes smaller than the composite size of those measurement data [34]. Due the complex relationship between multi-object density data size and object cardinality, state dimension, clutter intensity, detection probability, etc., an exhaustive study of this question is beyond the scope of this paper. Instead, a few numerical examples are presented under simplifying assumptions in order to demonstrate general trends.

The driving consideration is the amount of data needed to communicate multi-object densities versus that needed to communicate measurements. It would be difficult to fairly compare these quantities for hypothesis-oriented labeled RFS distributions, because of variability in the number of hypotheses kept and the variable object cardinalities within those hypotheses. For ease of exposition, only LMB densities are considered because of their simplified data structure. Furthermore, it is assumed that single-object densities are represented by Gaussian mixtures (GMs). Each component in an LMB density consists of a probability of existence, a label, and a GM with weights, means, and covariance matrices. Assuming single-precision floating point representations of these data, the size in bytes of a LMB distribution on the labels $\mathbb{L}$ is

$$N^{(\mathrm{LMB})} = 4|\mathbb{L}|\left(3 + n_{\mathrm{GM}}\left(1 + n_x + \frac{n_x^2 + n_x}{2}\right)\right), \quad (5)$$

where n_{GM} is the number of GM components, n_x is the dimension of the single-object state vector, and the three accounts for two-component labels and probabilities of existence. The measurement information, assuming a linear measurement model, zero-bias Gaussian white noise, and state-independent clutter and detection, would consist of the set of measurement vectors, the measurement noise covariances, and clutter and missed-detection probabilities. Again assuming single-precision floating point representations, the size in bytes of the measurement information for a set of measurements Z is

$$N^{(\mathrm{meas})} = 4(2 + n_z + n_z|Z|)\,, \quad (6)$$

where n_z is the dimension of the single-object measurement vector and the two accounts for the probabilities of clutter and missed detection.

A comparison of Equation (5) and Equation (6) for a variable number of objects is presented in Figure 8. The parameters of the comparison are that single-object densities are comprised of five Gaussian components, the single-object state vector dimension is six, and the single-object measurement vector dimension is three. The assumptions made are that the number of Bernoulli components of the LMB matches the number of objects, i.e. $|\mathbb{L}| = |X|$, and there is no clutter or missed detections. From Figure 8, it is clear that sharing measurement data is more efficient than sharing full posterior LMB distributions, and this would be considerably worse for hypothesis-oriented labeled RFS distributions or more complex single-object distributions.

For determination of the frequency at which posterior distribution sharing and fusion should be performed, it is useful to consider how many measurements must accumulate before distribution communication is more efficient. Using the scenario and assumptions of Figure 8, it is clear that the size of the LMB distribution would

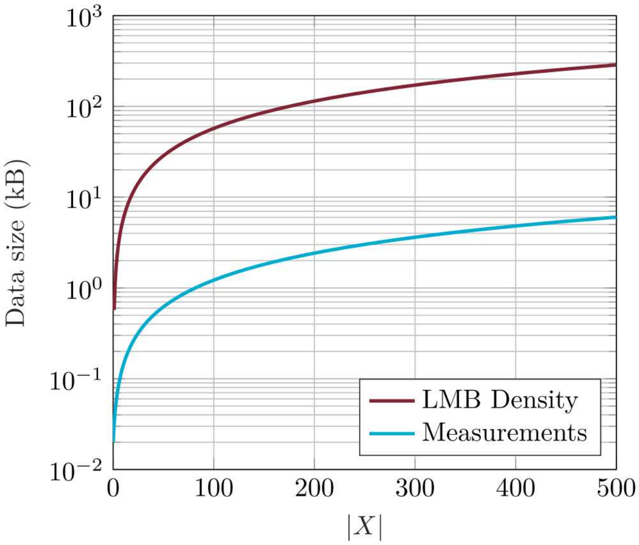


Figure 8. Plot of data packet size for variable object cardinality. Values do not consider message headers. The parameters are $n_{\mathrm{GM}} = 5$, $n_x = 6$, and $n_z = 3$. The assumptions are $|\mathbb{L}| = |X|$ and no clutter or missed detections.

not change as more measurements are collected. The distribution's data size would also not change following fusion, assuming that GM component pruning and merging is performed. In contrast, the cumulative size of the measurement data set grows as new measurements are collected and as new measurement data sets are received from other sensors. To capture these effects, Equation (6) is redefined, assuming constant measurement noise covariance and constant, state-independent probabilities of clutter and missed detection, as

$$N^{(\mathrm{meas})} = 4\sum_{s\in\mathcal{S}}\left(2+n_z+\sum_{k\in T^{(s)}}\left(1+n_z|Z_k^{(s)}|\right)\right), \tag{7}$$

where $\mathcal{S}$ is the set of indices s of sensors that have contributed to the measurement data set, $T^{(s)}$ is the set of time indices k of the measurement sets of sensor s, and $Z_k^{(s)}$ is the set of measurements at time k of sensor s. Note that the one in the second summand accounts for the time corresponding to a measurement set $Z_k^{(s)}$.

Figure 9 compares the amount of data accumulated over successive observations of a constant number of objects for density and measurement fusion. All other parameters and assumptions of this comparison are the same as those of the comparison in Figure 8. Figure 9 suggests that measurement communication and fusion will almost always be more efficient than density communication and fusion for a single sensor or single fusion iteration. If observations were taken at a frequency of 1Hz a system would have to wait about 45s to accumulate enough measurement data for posterior density communication to be more efficient. Such a low frequency of density fusion would undermine the benefits of distributed tracking in most practical systems. In general, the severe extent to which measurement communication is more efficient than density communication calls into question the utility of density fusion in bandwidth restricted systems.

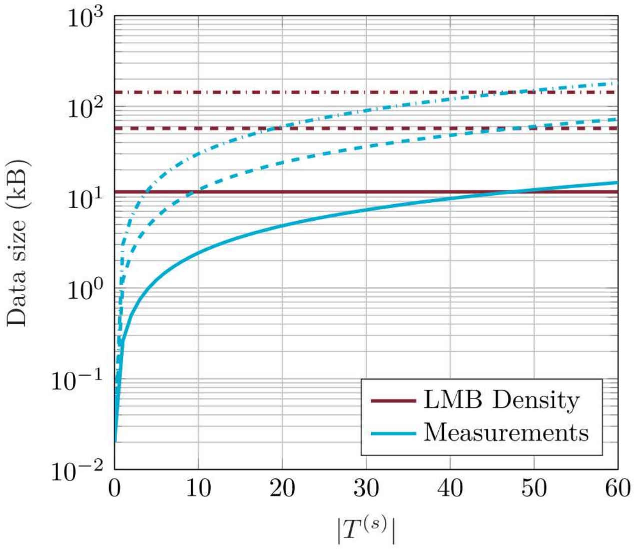


Figure 9. Plot of data packet size accumulated over time. Values do not consider message headers. The solid line is for $|X| = 20$, the dashed line is for $|X| = 100$, and the dot-dash line is for $|X| = 250$. The parameters are $n_{\mathrm{GM}} = 5$, $n_x = 6$, $|\mathcal{S}| = 1$, and $n_z = 3$. The assumptions are $|\mathbb{L}| = |X|$, constant object cardinality, and no clutter or missed detections.

The effects of the assumptions of this comparison must be considered before definitive conclusions can be drawn. On the side of density communication and fusion, the size the density is a highly conservative approximation in this scenario. Typically, the number of Bernoulli components of a LMB density exceeds the number of objects actually present due to clutter and object birth, and a larger, more realistic component count would increase the density's size considerably. If hypothesis-oriented labeled RFS densities were used in place of LMB densities, the data size of each density component would be significantly greater. Similarly, more complex single-object densities would produce larger data sizes.

On the other hand, the measurement set data size is also a conservative estimate. The use of complex measurement models or time-dependent error covariances would increase measurement set data size, as model parameters and/or a set of error covariances would need to be communicated. More significantly, state-dependent clutter or missed detection probabilities would require the communication of full probability hypothesis densitys (PHDs) or, if there is time dependence in addition to state dependence, sets of PHDs. The addition of these densities to the communicated measurement data set would drastically increase the set's size. Lastly, communication of single-sensor data sets would only propagate information between network neighbors. To fully exploit the advantages of distributed sensing, systems performing measurement fusion would have to aggregate and communicate sets of measurements from many sensors, which essentially scales the size of the collective measurement data set according to the number of contributing sensors.

It is beyond the scope of this work to explore every intricacy of the relationship between measurement and density communication efficiency in full, but its importance to future research in both labeled and unlabeled distributed RFS filtering is paramount. The choice of fusing measurements or fusing posterior densities and the frequencies at which each should be done would be among the central informing decisions in the design of any real-world system. Considering this importance, more in depth study of this relationship is required.

Track Management

Given finite local resources, any node in a distributed sensing system has a finite number of tracks it can manage simultaneously in real time, where track management refers to the propagation and maintenance of track data. Considering that each node has some state space region it is interested in, e.g. its FoR, it is reasonable for a node whose maintenance limits are stressed by the number of tracks to prioritize maintenance of tracks that are in its region of interest and drop tracks that are not. Alternatively, local track maintenance limits may be improved through the use of approximations that reduce the burdens of track management, e.g. GM component merging and pruning. However, the extent of such improvements is limited, as extensive approximation will invariably degrade tracking performance below acceptable levels. In distributed systems, sensors will be forced to drop tracks to suit local resources as global object cardinality grows. As a result of dropped tracks, label consistency is violated and support diversity develops, which prohibits use of direct AMD and GMD fusion as discussed.

This concern is illustrated in Figure 10, where two satellites observe objects on the ground. Consider the case where both satellites are tasked with tracking objects they pass over, but neither has the resources necessary to manage all tracks generated by the satellite pair. The satellites would need to make intelligent decisions on which tracks to manage. One approach might be to drop or reject for fusion tracks that could not be observed by a satellite in a reasonable time frame, such as the next orbit, because the satellite would be unable to provide useful new information on the track. The particular configuration of the satellites in Figure 10 demonstrates this point, as the equator orbiting satellite would be unable to observe objects at the poles, and it would thus be unreasonable and unhelpful for that satellite to maintain tracks of those objects.

Sensor Trust

It is standard practice in existing distributed sensing systems to weight local track data during fusion according to the relative accuracy of the systems and processes that produced them. Commonly, this is referred to as trust-based weighting [23]. A key difference between trust-based weighting in distributed labeled RFS systems and traditional systems stems from the lack of mathematically rigorous representations of uncertainty in the multi-object sense, e.g. data association and existence uncertainty, in the track data of traditional systems. Trust-weights in traditional track-to-track fusion systems are often assigned to account for this lack of principled uncertainty data. For example, such systems may down-weight the data produced by a sensor tracking a large number of objects or which has a high clutter rate, because these conditions cause greater data association uncertainty. In contrast, labeled RFS densities directly and accurately capture multi-object uncertainties. It would therefore be erroneous to consider multi-object uncertainties during weight assignment in distributed labeled RFS tracking systems. Issues such as this highlight the importance of establishing the use-cases and potential hazards of trust-based weighting in distributed labeled RFS systems.

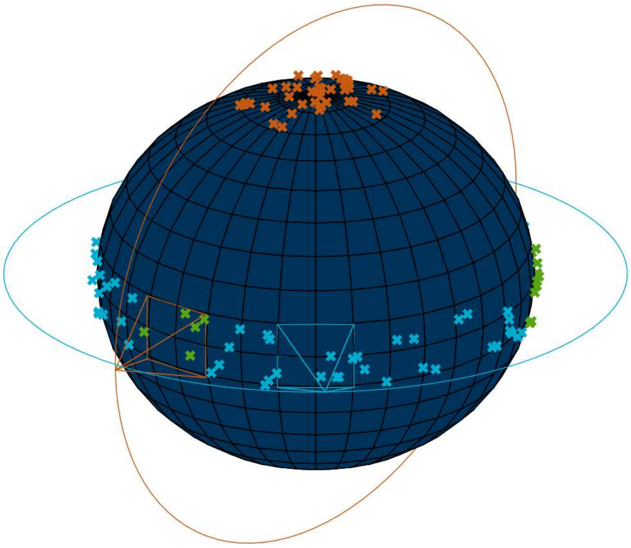


Figure 10. Illustration of track management concerns. Two satellites observe ground objects, represented by x's, and perform fusion when access is available. Objects share the color of the satellite that may observe them in its next orbit. Objects observable by both satellites in the next orbit are colored green.

The most readily identifiable use-case of trust-based weighting in labeled RFS systems is that of networks comprised of heterogeneous nodes. Resource differences across such nodes can cause modeling and filtering fidelity to vary significantly, and this variation may change dynamically over time. Consider a sensor performing labeled RFS tracking with a multiple-motion model. Such a system requires more computational resources to accommodate the additional uncertainty of object motion mode. In moments of increased object numbers, a resource-limited sensor might reduce the motion-model fidelity of the tracker in order to accommodate the higher number of objects. Such variations in fidelity cannot be captured in the uncertainty of local filtering densities. As a result, the globally convergent tracking solution can become biased and corrupted. By down-weighting those densities produced by heavier approximation, these effects can be reduced or effectively eliminated.

An issue involving trust-based weighting that has, to the authors' knowledge, not been discussed in the context of distributed labeled RFS fusion is that of *bottlenecking*. This term is used to describe the phenomena in which severely down-weighting a node's density with respect to those of its neighbors can slow or effectively stop the flow of information through a network. Consider a bandwidth restricted system that heavily truncates its filtering density to maintain real-time operation of a consensus-based sensing network. Assuming weights are kept constant during fusion, if this system's density is too heavily down-weighted, the densities of neighboring nodes with higher weights will overshadow the low-weight density during fusion. As a result of this overshadowing, little or essentially no information will be fused from the low-weight node into the densities of its neighbors. This phenomenon is illustrated in Figure 11 by the proxy example of a distributed network that iteratively takes the weighted average of nodes' local values. As in Figure 11, if a low-weight node provides the only link between two sections of a network graph, those sections effectively cannot communicate.

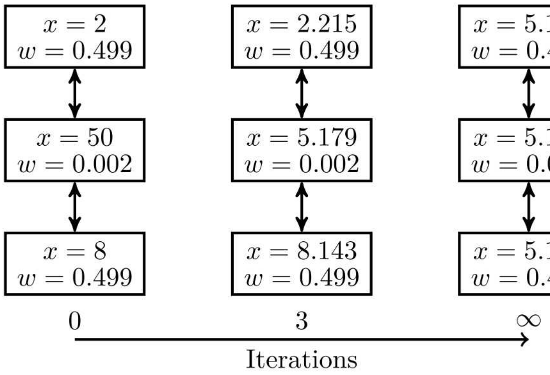


Figure 11. Illustration of the bottlenecking effect in a weighted averaging network. A node's value is denoted x and that value's weight in each averaging is denoted w. The central (bottleneck) node almost immediately averages to the convergent solution, but the outer nodes do not reach 95% accuracy until over 600 iterations have passed.

The development of bottlenecks in a network may be unavoidable due to environmental constraints and platform dynamics. However, it may be possible to mitigate their effects. Just as nodes neighboring a bottleneck overshadow the bottleneck's density with their own during fusion, the bottleneck has its local density overshadowed by those of its neighbors after some number of fusion iterations. Therefore, trust in the bottleneck's density can be increased with each fusion, as its original low-trust information has a diminishingly small presence relative to the high-trust information of its neighbors. As the weight increases, the neighboring nodes will increasingly incorporate information from the bottleneck. Thus, information from either side of the bottleneck can efficiently disseminate to the entire network, while information originating from the bottleneck maintains a reduced presence. Figure 12 illustrates this approach applied to the proxy example of Figure 11.

Any number of use-cases may exist for trust-based weighting in distributed labeled RFS systems, and this work makes no attempt to exhaustively enumerate them. In identifying use cases, it is necessary to consider the behavior and needs of a particular system to determine the most effective weighting strategy, while keeping in mind the unique attributes of multi-object densities as compared to traditional track data representations. Similarly, the pitfalls of trust-based weighting in these systems are likely many and mostly unknown. The response of a system's behavior to a particular weighting strategy must be analyzed to determine if the strategy produces the intended results while maintaining satisfactory overall system performance.

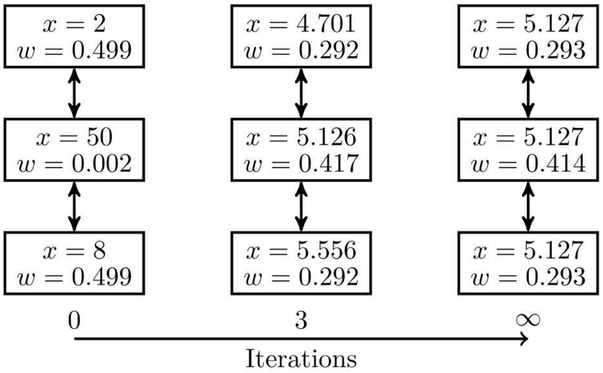


Figure 12. Illustration of reduction of the bottlenecking effect in a weighted averaging network. A node's value is denoted x and that value's weight in each averaging is denoted w. Weights are averaged with neighboring weights and re-normalized after each iteration. All nodes reach 95% accuracy by the 4th iteration. The convergent solution is altered by weight averaging, but this error can be reduced by delaying when that averaging is started.

6. Summary

The principled foundations upon which labeled random finite set (RFS) tracking is built allows for its application to highly complex problems that other multi-object tracking (MOT) approaches struggle or fail to address. Among these is the problem of large scale distributed MOT of large numbers of objects. Successful application of labeled RFS tracking to this problem requires principled multi-sensor fusion techniques, which can be accomplished within the RFS framework through either measurement fusion or multi-object posterior density fusion. These methods are particularly appealing as their results can be proven to lower estimation error when compared to single-sensor solutions. However, many practical considerations need to be addressed for these techniques to be useful in real world systems.

In this work, several of the unique challenges that the distributed MOT problem presents were explored within the context of labeled RFS methods. The label consistency assumption of the popular geometric mean density (GMD) fusion strategy proves to be very problematic due to issues related to object birth models, distribution support diversity, geographic scalability, systems interoperability, and stochastic variations in clutter and detections. It was demonstrated that it is infeasible to ensure consistent labeling of new objects in distributed systems of practical scale. Thus, GMD fusion cannot be directly applied in many cases and alternatives or adaptations need to be used. The development of tracking methods that are robust to label inconsistencies is an important area for future distributed labeled RFS tracking research.

Additionally, several other aspects of distributed labeled RFS tracking have been examined. Concerns such as bandwidth constraints, track management, and sensor trust were discussed and characterized. It was shown that the utility of posterior density fusion is disputable from the bandwidth perspective, as communication of raw measurements will almost always be more efficient than communication of posterior densities. How these efficiencies evolve over time and with propagation of data through a network is highly complex and becomes increasingly difficult to analyze as simplifying assumptions are relaxed. Further study of this problem is necessary to show that posterior density fusion is a practical option for bandwidth-limited systems and, in particular, experimental implementations are needed to draw definitive conclusions.

7. Acknowledgments

This work was supported by the Laboratory Directed Research and Development program at Sandia National Laboratories, a multi-mission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

References

[1] M. Üney, D. E. Clark, and 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.

[2] R. P. Mahler, *Advances in Statistical Multisource-Multitarget Information Fusion.* Artech House, 2014.

[3] F. Papi, "Multi-sensor δ-GLMB filter for multi-target tracking using Doppler only measurements," in *2015 European Intelligence and Security Informatics Conference*, pp. 83–89, 2015.

[4] B.-N. Vo and B.-T. Vo, "Multi-sensor multi-object tracking with the generalized labeled multi-Bernoulli filter," 2017.

[5] M. Üney, J. Houssineau, E. Delande, S. J. Julier, and D. E. Clark, "Fusion of finite set distributions: Pointwise consistency and global cardinality," *arXiv preprint arXiv:1802.06220*, 2018.

[6] M. Liggins, Chee-Yee Chong, I. Kadar, M. Alford, V. Vannicola, and S. Thomopoulos, "Distributed fusion architectures and algorithms for target tracking," *Proceedings of the IEEE*, vol. 85, no. 1, pp. 95–107, 1997.

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

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

[9] Y. Bar-Shalom, P. K. Willett, and X. Tian, *Tracking and Data Fusion.* YBS publishing, 2011.

[10] B.-N. Vo and B.-T. Vo, "An implementation of the multi-sensor generalized labeled multi-Bernoulli filter via Gibbs sampling," in *Information Fusion (Fusion), 2017 20th International Conference on*, pp. 1–8, IEEE, 2017.

[11] S. Li, W. Yi, R. Hoseinnezhad, G. Battistelli, B. Wang, and L. Kong, "Robust distributed fusion with labeled random finite sets," *IEEE Transactions on Signal Processing*, vol. 66, no. 2, pp. 278–293, 2018.

[12] B.-T. Vo and B.-N. Vo, "Labeled random finite sets and multi-object conjugate priors," *IEEE Transactions on Signal Processing*, vol. 61, no. 13, pp. 3460–3475, 2013.

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

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

[15] K. A. LeGrand and K. J. DeMars, "The data-driven delta-generalized labeled multi-Bernoulli tracker for automatic birth initialization," in *Signal Processing, Sensor/Information Fusion, and Target Recognition XXVII*, vol. 10646, p. 1064606, International Society for Optics and Photonics, 2018.

[16] B. Ristic, D. Clark, B.-N. Vo, and B.-T. Vo, "Adaptive target birth intensity for PHD and CPHD filters," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 2, pp. 1656–1668, 2012.

[17] S. Reuter, D. Meissner, B. Wilking, and K. Dietmayer, "Cardinality balanced multi-target multi-Bernoulli filtering using adaptive birth distributions," in *Information Fusion (FUSION), 2013 16th International Conference on*, pp. 1608–1615, IEEE, 2013.

[18] B. Wei and B. Nener, "Distributed space debris tracking with consensus labeled random finite set filtering," *Sensors*, vol. 18, no. 9, p. 3005, 2018.

[19] J. S. McCabe and K. J. DeMars, "Fusion methodologies for orbit determination with distributed sensor networks," in *2018 21st International Conference on Information Fusion (FUSION)*, pp. 1323–1330, IEEE, 2018.

[20] B. R. Abidi, N. R. Aragam, Y. Yao, and M. A. Abidi, "Survey and analysis of multimodal sensor planning and integration for wide area surveillance," *ACM Computing Surveys (CSUR)*, vol. 41, no. 1, p. 7, 2009.

[21] M. Nekovee, "Sensor networks on the road: the promises and challenges of vehicular ad hoc networks and grids," in *Workshop on ubiquitous computing and e-Research*, vol. 47, 2005.

[22] R. P. Mahler, "Optimal/robust distributed data fusion: a unified approach," in *Signal Processing, Sensor Fusion, and Target Recognition IX*, vol. 4052, pp. 128–139, International Society for Optics and Photonics, 2000.

[23] M. Liggins II, C.-Y. Chong, D. Hall, and J. Llinas, *Distributed data fusion for network-centric operations.* CRC Press, 2012.

[24] G. Battistelli, L. Chisci, C. Fantacci, A. Farina, and R. P. Mahler, "Distributed fusion of multitarget densities and consensus PHD/CPHD filters," in *Signal processing, sensor/information fusion, and target recognition XXIV*, vol. 9474, p. 94740E, International Society for Optics and Photonics, 2015.

[25] C. Fantacci, B. N. Vo, B.-T. Vo, G. Battistelli,

and L. Chisci, “Robust fusion for multisensor multiobject tracking,” *IEEE Signal Processing Letters*, 2018.

[26] B. Wang, W. Yi, S. Li, L. Kong, and X. Yang, “Distributed fusion with multi-Bernoulli filter based on generalized covariance intersection,” in *Radar Conference (RadarCon), 2015 IEEE*, pp. 0958–0962, IEEE, 2015.

[27] K.-C. Chang, C.-Y. Chong, and S. Mori, “Analytical and computational evaluation of scalable distributed fusion algorithms,” *IEEE transactions on Aerospace and Electronic Systems*, vol. 46, no. 4, pp. 2022–2034, 2010.

[28] N. R. Ahmed and M. Campbell, “Fast consistent Chernoff fusion of Gaussian mixtures for ad hoc sensor networks,” *IEEE Transactions on Signal Processing*, vol. 60, no. 12, pp. 6739–6745, 2012.

[29] S. J. Julier, T. Bailey, and J. K. Uhlmann, “Using exponential mixture models for suboptimal distributed data fusion,” in *Nonlinear Statistical Signal Processing Workshop, 2006 IEEE*, pp. 160–163, IEEE, 2006.

[30] T. Heskes, “Selecting weighting factors in logarithmic opinion pools,” in *Advances in neural information processing systems*, pp. 266–272, 1998.

[31] K. J. DeMars, J. S. McCabe, and J. E. Darling, “Collaborative multi-sensor tracking and data fusion,” 2015.

[32] T. Li, J. M. Corchado, and S. Sun, “Partial consensus and conservative fusion of Gaussian mixtures for distributed PHD fusion,” pp. 1–12, 2017.

[33] A.-A. Saucan and P. K. Varshney, “Distributed cross-entropy δ-GLMB filter for multi-sensor multitarget tracking,” in *2018 21st International Conference on Information Fusion (FUSION)*, pp. 1559–1566, IEEE, 2018.

[34] O. E. Drummond, “Track and tracklet fusion filtering,” in *Signal and Data Processing of Small Targets 2002*, vol. 4728, pp. 176–196, International Society for Optics and Photonics, 2002.

Biography

Augustus Buonviri is currently a year-round intern at Sandia National Laboratories. He received his B.S. degree in aerospace engineering from Mississippi State University in 2018, and will be pursuing a Ph.D. in aerospace engineering at the University of Texas at Austin starting Spring 2019. His interests include single and multiple object filtering and estimation, and spacecraft attitude determination and control.

Matthew York is currently a year-round research and development intern at Sandia National Laboratories. He is pursuing his B.S. in aerospace engineering at the Missouri University of Science & Technology. His interests include scalable multi-object tracking algorithms and remote sensing systems design.

Keith LeGrand is a Senior Member of Technical Staff at Sandia National Laboratories. Keith earned both his B.S. and M.S. degrees in aerospace engineering at the Missouri University of Science and Technology before joining Sandia full-time in 2015. His research focuses include space situational awareness, filtering and estimation, and multitarget tracking.

James Meub is a Senior Member of Technical Staff at Sandia National Laboratories. James earned his B.S. and M.S. degrees in aerospace engineering from the Missouri University of Science and Technology. His interests include remote sensing and systems design.

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy’s National Nuclear Security Administration under contract DE-NA0003525.