

# Efficient Implementation of Multi-sensor Adaptive Birth Samplers for Labeled Random Finite Set Tracking

Jennifer Bondarchuk, Anthony Trezza, Donald J. Bucci Jr.,

*Advanced Technology Laboratories*

*Lockheed Martin*

Cherry Hill, NJ, USA

{jennifer.a.bondarchuk, anthony.t.trezza, donald.j.bucci.jr}@lmco.com

**Abstract**—Adaptive track initiation remains a crucial component of many modern multi-target tracking systems. For labeled random finite sets multi-object filters, prior work has been established to construct a labeled multi-object birth density using measurements from multiple sensors. A naive construction of this adaptive birth set density results in an exponential number of newborn components in the number of sensors. A truncation procedure was provided that leverages a Gibbs sampler to truncate the birth density, reducing the complexity to quadratic in the number of sensors. However, only a limited discussion has been provided on additional algorithmic techniques that can be employed to substantially reduce the complexity in practical tracking applications. In this paper, we propose five efficiency enhancements for the labeled random finite sets multi-sensor adaptive birth procedure. Simulation results are provided to demonstrate their computational benefits and show that they result in a negligible change to the multi-target tracking performance.

**Index Terms**—Random finite sets, Target tracking, Measurement adaptive birth

## I. INTRODUCTION

The objective of a Multi-target Tracking (MTT) algorithm is to jointly estimate the number of objects in the scene and their trajectories from noisy measurements observed at one or more sensors over time. MTT solutions are often required in a variety of applications domains such as defense, automotive, and maritime traffic monitoring systems [1]. Many approaches have been proposed including Global Nearest Neighbor (GNN) techniques [1], Joint Probabilistic Data Association (JPDA) [2], Multiple Hypothesis Tracking (MHT) [3], Belief Propagation (BP) [4], and Random Finite Sets (RFS) [5], [6]. We direct the reader to [7] for a detailed survey of the field, recent advances, and example applications. In this work, we specifically focus on the labeled RFS problem formulation of the  $\delta$ -Generalized Labeled Multi-Bernoulli (GLMB) and Labeled Multi-Bernoulli (LMB) filters [8]–[11].

It has been shown that the  $\delta$ -GLMB labeled RFS density is a conjugate prior with itself under the multi-object Bayes filtering equations [8], [9], [12], resulting in a tractable solution to the MTT problem. In the  $\delta$ -GLMB filter, the predicted labeled multi-object state is modeled as the superposition of surviving and newborn targets. Under the assumption that targets move,

appear, and die independently of each other when conditioned on the prior state, the transition density for a labeled multi-object state becomes the product of a transition density over the surviving targets and a prior density over the newborn targets in the predicted state. An inaccurate model for the newborn target density can result in missing targets, increasing the number of ghost tracks, increasing the runtime complexity, and decreasing the tracker accuracy.

In an adaptive birth strategy, the newborn target density is constructed adaptively from the measurements at each time step [13]. This approach is effective in many applications since minimal prior information is known about where and how objects can appear in the surveillance volume. Care must be taken when designing these strategies to ensure that tracker performance is maintained without adversely affecting computational complexity. Single-sensor adaptive birth techniques for labeled RFS were introduced in [10] for the LMB filter and [14] for the  $\delta$ -GLMB filter. These approaches follow a similar concept as presented in [13] by constructing a component of an LMB RFS from every single-sensor measurement observed at the previous time step. The probabilities of birth for each of these LMB components were set proportionally to a maximum birth rate parameter. The proportionality factor was based on how much a measurement did not associate with the persisting targets from the last time step. In contrast, the multi-sensor adaptive birth regime carries the additional complication that there are exponentially many possible measurement tuples that could be used to generate newborn targets. In our previous work [15], we proposed a stochastic Gibbs sampling approach for truncating multi-sensor measurement tuples that would likely be pruned in the subsequent  $\delta$ -GLMB update step. The proposed Gibbs sampler exhibited quadratic complexity in the number of sensors, and both an Sequential Monte Carlo (SMC) approximation and a closed form Gaussian solution were provided. In [16], we proposed an alternative Gibbs sampling architecture and two early termination criteria that can reduce the number of Markov chain observations that the Gibbs sampler must consider, but focused the analysis primarily on the  $\delta$ -GLMB ranked assignment problem.

As discussed in [16], despite the approach in [15] having quadratic scaling complexity, each Gibbs iteration can contribute to a non-negligible amount of the filter's runtime. For example, in our simulations presented in Section IV, if no efficient methods are used, the adaptive birth procedure in [15] accounts for 98% of the simulation's runtime. Hence, any runtime improvements made to adaptive birth contribute similarly to the overall runtime reduction.

In this paper, we expand the work of [15] by proposing five practical techniques to improve the runtime efficiency, including pre-pruning, gating, memoization, missed detection sample skipping, post-pruning and capping. This paper is organized as follows. Section II, background on adaptive birth. Section III details the proposed efficiency methods. Section IV provides simulation results for each proposed method, followed by conclusions in Section V.

## II. BACKGROUND

We adopt the notation of [8], [12], [15] as follows. Lowercase letters denote vectors (e.g.,  $x, \mathbf{x}$ ), whereas uppercase letters denote finite sets (e.g.,  $X, \mathbf{X}$ ). Bold letters denote labeled states and their distributions (e.g.,  $\mathbf{x}, \mathbf{X}, \boldsymbol{\pi}$ ) and black-board bold letters denote spaces (e.g.,  $\mathbb{X}, \mathbb{Z}, \mathbb{L}, \mathbb{R}$ ). Variable sequences  $X_i, X_{i+1}, \dots, X_j$  are abbreviated  $X_{i:j}$ . The inner product  $\int f(x)g(x)dx$  is written as  $\langle f, g \rangle$ . Finally, we drop the subscript notation for current time step  $k$  and use subscript '+' to indicate the next time step ( $k+1$ ).

We assume the standard point-object models described in [8] that result in a closed-form  $\delta$ -GLMB filtering recursion. In this formalism, the newborn birth density is modeled as a labeled RFS density.

### A. Multi-sensor Multi-target Adaptive Birth Model

Our objective is to adaptively construct an LMB RFS density  $\mathbf{f}_{B,+}$  given the multi-object measurement sets  $Z^{(s)} \in \mathbb{Z}^{(s)}$  from sensors  $s \in \{1, \dots, V\}$ . Let  $m^{(s)} = |Z^{(s)}|$  and  $\mathbb{J}^{(s)} \triangleq \{1, \dots, m^{(s)}\}$  be an index set into  $Z^{(s)}$ . Let  $\mathbb{J}_0^{(s)} \triangleq \mathbb{J}^{(s)} \cup \{0\}$ , where the entry 0 represents a miss-detection. The tuple  $J \triangleq (j^{(1)}, \dots, j^{(V)}) \in \mathbb{J}_0^{(1)} \times \dots \times \mathbb{J}_0^{(V)}$  is a multi-sensor measurement index with  $Z_J \triangleq (z_{j^{(1)}}^{(1)}, \dots, z_{j^{(V)}}^{(V)})$  as the corresponding tuple of multi-sensor measurements.

As suggested in [15], the birth LMB RFS density used at time  $(k+1)$  is modeled as

$$\mathbf{f}_{B,+} = \{(r_{B,+}(l_+), p_{B,+}(\cdot, l_+ | Z_J))\}_{l_+ \in \mathbb{B}_+}, \quad (1)$$

where  $l_+ \in \mathbb{B}_+ = \{(k+1, J) : \forall J \in \mathbb{J}_0\}$  is the newborn object's label having birth probability  $r_{B,+}(l_+)$ . The quantity  $p_{B,+}(\cdot, l_+ | Z_J)$  is the state distribution of label  $l_+$  conditioned on the measurement tuple  $Z_J$ . The state distribution is given by the Chapman-Kolmogorov equation [5]

$$p_{B,+}(x_+, l_+ | Z_J) = \int f_+(x_+ | x, l) p_B(x, l_+ | Z_J) dx, \quad (2)$$

where,

$$p_B(x, l_+ | Z_J) = \frac{p_B(x, l_+) \psi_Z^J(x, l_+)}{\bar{\psi}_Z^J(l_+)} \quad (3)$$

$$\bar{\psi}_Z^J(l_+) = \langle p_B(\cdot, l_+), \psi_Z^J(\cdot, l_+) \rangle. \quad (4)$$

Here  $f_+(x_+ | x, l)$  is the Markov transition density modeling the dynamics of the object state,  $\psi_Z^J(x, l_+)$  is the multi-sensor measurement pseudolikelihood function, and  $p_B(x, l_+)$  is the prior distribution on the newborn object's state. Assuming conditional independence of the sensor measurements, the multi-sensor measurement pseudolikelihood function is [12],

$$\psi_Z^J(\mathbf{x}) \triangleq \prod_{s=1}^V \psi_{Z^{(s)}}^{s, j^{(s)}}(\mathbf{x}), \quad (5)$$

where  $\psi_{Z^{(s)}}^{s, j^{(s)}}(\mathbf{x})$  is the per-object, per-sensor measurement pseudolikelihood [7], [8],

$$\psi_{Z^{(s)}}^{s, j^{(s)}}(\mathbf{x}) = \begin{cases} \frac{p_D^{(s)}(\mathbf{x}) g^{(s)}(z_{j^{(s)}}^{(s)} | \mathbf{x})}{\kappa^{(s)}(z_{j^{(s)}}^{(s)})} & j^{(s)} \in \mathbb{J}^{(s)} \\ 1 - p_D^{(s)}(\mathbf{x}) & j^{(s)} = 0 \end{cases} \quad (6)$$

and  $p_D^{(s)}(\mathbf{x})$  models the probability that sensor  $s$  detected the target state  $\mathbf{x}$ ,  $g^{(s)}(z_{j^{(s)}}^{(s)} | x)$  is the single-sensor likelihood function and  $\kappa^{(s)}(z_{j^{(s)}}^{(s)})$  is the Poisson RFS intensity function of the clutter process.

The birth probability of label  $l_+$  is modeled in [15] as

$$r_{B,+}(l_+) = \min(r_{B,\max}, \hat{r}_{B,+}(l_+) \lambda_{B,+}), \quad (7)$$

where  $r_{B,\max} \in [0, 1]$  is the maximum existence probability of a newborn target and  $\lambda_{B,+}$  is the expected number of target births at time step  $k+1$ . The effective birth probability  $\hat{r}_{B,+}(l_+)$  is given as

$$\hat{r}_{B,+}(l_+) = \frac{r_U(J) \bar{\psi}_Z^J(l_+)}{\sum_{J' \in \mathbb{J}_0} r_U(J') \bar{\psi}_Z^{J'}(l_+)}, \quad (8)$$

where  $r_U(J)$  is the probability that the multi-sensor measurement tuple  $J$  does not associate with any existing targets. This non-association probability is approximated as

$$r_U(J) \propto \prod_{j \in J} (1 - r_A(j)), \quad (9)$$

where  $r_A(j)$  is the probability that a single-sensor's measurement,  $z_{j^{(s)}}^{(s)}$ , is associated with existing targets in the  $\delta$ -GLMB posterior density. For more information on these quantities, we direct the reader to [15].

### B. Truncating the Multi-sensor Adaptive Birth LMB

The number of newborn labels  $\mathbb{B}_+$  is driven by the cardinality of  $\mathbb{J}_0$  which has  $O(m^V)$  elements, where  $m$  is the worst case number of measurements. It is therefore not practical to enumerate every possible entry at each time step. In the spirit of [17], a technique was proposed in [15] to sample labels  $\mathbb{B}'_+ \subset \mathbb{B}_+$  that have high  $\hat{r}_{B,+}(l_+)$  and hence are likely

to persist through future measurement updates. The sampling distribution is modeled as

$$p(l_+) \propto \hat{r}_{B,+}(l_+) \propto r_U(J) \bar{\psi}_Z^J(l_+). \quad (10)$$

Directly sampling from the categorical distribution  $p(l_+)$  requires evaluation of an exponential number of possible birth labels. By construction of the birth set, sampling from  $p(l_+)$  is equivalent to sampling multi-sensor measurement tuples from the joint distribution,  $p(J) = p(j^{(1)}, \dots, j^{(V)})$ . A Gibbs sampler was proposed in [15] to achieve efficient sampling from Equation (10) using the Markov transition probabilities

$$p(j^{(s)} | J^{-s}) \propto (1 - r_A(j^{(s)})) \bar{\psi}_Z^J(l_+), \quad (11)$$

where  $J^{-s} = (j^{(1)}, \dots, j^{(s-1)}, j^{(s+1)}, \dots, j^{(V)})$ .

The stochastic Gibbs sampler that generates the measurement tuples (i.e., birth components) with significant birth probabilities is given in [15, Algorithm 1]. Similar to [17], it exhibits an exponential convergence rate and does not require a burn-in period since unique solutions can be directly used in construction of the birth set. An SMC method was provided in [15, Section VI] for evaluating the transition probabilities and generating the state distributions. Closed-form solutions for these quantities in the Gaussian case can be found in [15, Section VII].

### III. EFFICIENCY IMPROVEMENTS

In this section we discuss our proposed methods to reduce the runtime of the adaptive birth procedure. We describe the motivation for each method as well as our implementation.

#### A. Associated Measurement Pre-pruning

To pre-prune measurements, we exploit the fact that each measurement's association probability must be computed at the previous time step and provide an efficient check to remove measurements from the sample space that are likely to have been associated with existing targets.

The association probability,  $r_A(j^{(s)})$ , is computed for every measurement and provided as an input to the multi-sensor adaptive birth procedure. As seen in Equation (9), the joint unassociation probability assumes conditional independence between sensors. If any measurement  $z_{j^{(s)}}^{(s)}$  in the measurement tuple  $j^{(s)} \in J$  has a high association probability, then  $r_U(J)$  will be low. By Equation (8), a low value for  $r_U(J)$  results in a low value for the birth probability in Equation (7). This is intuitive as a low unassociation probability indicates that at least one measurement in the measurement tuple is likely to have associated with an existing target and therefore should not be used to birth a newborn target.

This is accounted for by construction of the Markov transition density in Equation (11) such that if  $r_A(j^{(s)})$  is high, then it is unlikely that  $j^{(s)}$  will be transitioned to and therefore will be unlikely to be sampled. However, in the current implementation, the value for  $\bar{\psi}_Z^J$  in Equation (4) still needs to be computed or approximated.

In practice, it has been observed that for highly accurate sensors,  $r_A(j^{(s)})$  tends to either be very close to 0 or 1.

Additionally, in practical multi-sensor, multi-target tracking applications with many targets in the scene, it is possible that many measurements have a high value for  $r_A(j^{(s)})$  and thus results in many unnecessary and computationally costly computations of  $\bar{\psi}_Z^J$ .

Since the association probability is provided as an input, we propose a simple pre-pruning approach to remove measurements from the sample space whose association probability is above a threshold,  $r_A(j^{(s)}) > \tau_A$  with  $\tau_A \in [0, 1]$ .

#### B. Measurement Gating

In Equation (5), the multi-sensor measurement pseudolikelihood assumes conditional independence between sensors resulting in a product likelihood form. This means that if any subset of measurements in the measurement tuple result in a low joint pseudolikelihood when paired together, then the multi-sensor measurement pseudolikelihood for the measurement tuple will also be low. This is intuitive as it means that a subset of measurements in the measurement tuple are unlikely to have been generated from the same target.

We propose a gating procedure by pre-computing (or approximating) the value for  $\bar{\psi}_Z^{(j^{(a)}, j^{(b)})}$  for every 2-pair measurement combination of sensors  $a$  and  $b$ . The value of the 2-pair average pseudolikelihood can be checked against a gating threshold  $\tau_G$  and stored in a  $4 - D$  matrix that is indexed by the pair of sensors and measurement indices,  $M[a, b, j^{(a)}, j^{(b)}] = (\bar{\psi}_Z^{(j^{(a)}, j^{(b)})} < \tau_G)$ .

When computing the transition density for  $j^{(a)}$ , an efficient check can be done to see if  $M[a, b, j^{(a)}, j^{(b)}] = 0$  for any  $j^{(b)} \in J^{-a}$ . If this occurs, then the transition to  $j^{(a)}$  can be skipped by setting  $p(j^{(a)} | J^{-a}) \approx 0$  without computing  $\bar{\psi}_Z^J$ .

The gating check is an  $O(V)$  procedure, but maintains the overall complexity as quadratic in the number of sensors for both the Monte Carlo and Gaussian solutions in [15]. In the Monte Carlo approach, the value  $\bar{\psi}_Z^{(j^{(a)}, j^{(b)})}$  can be pre-computed in  $O(N_p)$  time resulting in a complexity of  $O(N_p V^2 m^2)$  to make the gate matrix  $M$ .

Alternatively, a Mahalanobis or Euclidean distance gating procedure between the maximum likelihood estimates  $\hat{\mathbf{x}}^{(a)}$  and  $\hat{\mathbf{x}}^{(b)}$  of  $g^{(a)}(z_{j^{(a)}}^{(a)} | \mathbf{x})$  and  $g^{(b)}(z_{j^{(b)}}^{(b)} | \mathbf{x})$  could be used as an approximation. The Mahalanobis gate is constructed as  $M[a, b, j^{(a)}, j^{(b)}] = (d_M^{2,(a)}(h^{(a)}(\hat{\mathbf{x}}^{(b)})) < \tau_{G,a}) \vee (d_M^{2,(b)}(h^{(b)}(\hat{\mathbf{x}}^{(a)})) < \tau_{G,b})$  where  $d_M^{2,(a)}$  is the squared Mahalanobis distance for the distribution  $g^{(a)}(h^{(a)}(\hat{\mathbf{x}}^{(b)}))$  and similar for  $d_M^{2,(b)}$ . If  $g^{(s)}(\cdot)$  is Gaussian, then  $d_M^{2,(s)}$  follows a chi-squared distribution with  $n_Z^{(s)}$  degrees of freedom and the gate threshold  $\tau_{G,s}$  can be set to achieve a particular gating probability using the cumulative chi-squared distribution. The Euclidean distance gate is constructed as  $M[a, b, j^{(a)}, j^{(b)}] = \|\hat{\mathbf{x}}^{(a)} - \hat{\mathbf{x}}^{(b)}\|_2 < \tau_G$ . These simple checks often prove to be useful in many practical applications, especially when the measurement uncertainty is low.

#### C. Memoization

In memoization, we exploit the fact that many computations are repeated during the Gibbs sampling procedure and the

results can be efficiently stored and accessed when needed.

A Gibbs sampler is a Markov chain Monte Carlo method that indirectly generates samples from a target distribution by simulating state observations of a Markov chain according to a stationary transition density. This approach can be seen as a type of random walk through sample space, generating samples as it goes along. At every step in the Markov chain simulation, the transition probabilities must be computed so the next chain observation can be sampled. As the Markov chain simulation progresses, it is possible (and often likely) that the same state may be reached multiple times. When this happens, the transition probabilities are unnecessarily recomputed each time, since the transition density is stationary.

In the multi-sensor adaptive birth sampler, recomputing the transition density can be computationally expensive. Specifically, by Equation (10), the average pseudolikelihood  $\bar{\psi}_Z^J$  must be recalculated. Note the same average pseudolikelihood will need to be computed every time a state wants to consider transitioning to the state  $J$ , even if  $J$  is never sampled. This means that not only is the entire transition density for a given state unnecessarily recomputed, but the average pseudolikelihood will be unnecessarily recomputed by any state that considers transitioning into a state for which we've already computed  $\bar{\psi}_Z^J$ . This can be costly, especially if this happens many times throughout the sampling procedure.

Further, after the Gibbs sampling truncation procedure is completed and a truncated label space is sampled, a component in the LMB is then constructed for every newborn label. By Equation (3) and Equation (8), to construct the labeled Bernoulli component the average pseudolikelihood (which was already calculated in the sampling procedure) must again be recalculated two more times.

As is commonly done in dynamic programming algorithms, we can avoid these redundant calculations by caching the value of  $\bar{\psi}_Z^J$  once we compute it, and access it if it is needed again. This can easily be implemented via a hash map keyed on the measurement tuple  $J$ , resulting in an  $O(1)$  access operation.

#### D. Prune and Cap Birth LMB

Although the Gibbs sampling procedure is designed to truncate many of the components with low birth probability, it is not guaranteed that all labels in the truncated LMB will have a high birth probability. Low birth probability labels may be sampled at any point, but will commonly occur in Markov Chain Monte Carlo methods while the sampler is converging to the target distribution. In the  $\delta$ -GLMB and LMB filtering recursions, it is standard practice to prune and cap the belief state to remove low existence probability labels and cap the number of tracks kept alive to a manageable number [10]. We can apply the same prune and cap procedure to the birth LMB before providing them to the filter to reduce unnecessary computations.

#### E. Missed Detection Sample Skipping

In missed detection sample skipping, we show that in many practical applications, such as those with ambiguous sensing

modalities, if a newborn label is constructed by a multi-sensor measurement tuple with many missed detections, then it can lead to ambiguities in the downstream filtering recursion. To prevent that, these sampled labels can often be skipped to reduce several unnecessary computations in the downstream filtering recursion.

Grouping the missed detected and detected elements of  $J$  in  $\psi_Z^J$  results in,

$$\psi_Z^J(\mathbf{x}) = \left[ \prod_{\substack{s'=1 \\ j^{(s')}=0}}^V (1 - p_D^{(s')}(\mathbf{x})) \right] \times \left[ \prod_{\substack{s'=1 \\ j^{(s')}>0}}^V \frac{p_D^{(s')}(\mathbf{x}) g^{(s')}(z_{j^{(s')}}^{(s')} | x)}{\kappa^{(s)}(z_{j^{(s')}}^{(s')})} \right]. \quad (12)$$

When  $p_B(x, l_+)$  and  $p_D^{(s)}(\mathbf{x})$  are not a function of  $x$ , then by Equation (3) the spatial distribution  $p_B(x, l_+ | Z_J)$  constructed from a tuple  $J$  with many missed detections is often uninformative. This is specifically important when fusing ambiguous sensor measurements without an invertible observation function such as bistatic range, angle-only, time difference of arrival (TDOA) and frequency difference of arrival (FDOA) [18]. In these applications, a minimum number of observations from disperse sensors are often required to unambiguously compute an intersection point in cartesian space (e.g., angle triangulation). For example in a multi-sensor 2D bearing-only sensing application, at least 2 sensors need to observe a target for it to be triangulated since range is unobserved.

In [15], [18], procedures such as triangulation do not explicitly occur as the ambiguity naturally resolves through the multi-sensor pseudolikelihood function in Equation (3). However, if the number of non-missed detection observations in  $J$  is below the minimum number of observations required to resolve this ambiguity, the spatial distribution will likely be uninformative and may falsely associate with measurements in the  $\delta$ -GLMB update procedure. For these types of applications, it is often useful to make an explicit rule on the minimum number of non-missed detection measurements in  $J$  for a newborn component to be constructed.

Similarly, in applications containing one or more sensors with a high clutter rate, it is likely that a state containing a clutter measurement may be transitioned to during the Gibbs sampling procedure. Since a target did not generate the clutter measurement it is often most probable that the remaining sensors missed the detection, resulting in sampling the all-but-one missed detection measurement tuple  $J = (-1, \dots, j^{(s)}, \dots, -1)$ . In this case, a labeled Bernoulli component in the LMB would be constructed from this clutter observation.

In both cases, the resulting birth probability of the constructed labeled Bernoulli component will likely be low if one or more sensors that missed the detection have a high

detection probability. However, as discussed in Section III-D, by adding this component in the birth LMB the onus is now shifted to the  $\delta$ -GLMB to truncate any hypotheses containing this label, resulting in many unnecessary computations that could be avoided by simply pruning components from the birth LMB that were constructed with a tuple containing more than a maximum number of missed detections. This procedure will often coincide with the pruning procedure discussed in Section III-D but can be valuable to check explicitly.

#### IV. SIMULATIONS

In this section, we used a similar simulation as Scenario 1 from [15, Section VIII.A] with the particle LMB filter to highlight the benefits of the proposed efficiency improvements. The target and sensor positions of the simulation are shown in Figure 1. The simulations were implemented in Python and run on an AMD EPYC 7742 64-Core Processor with 16GB of RAM. The runtime results include other inefficiencies such as logging, but are held constant throughout the comparison. Additional runtime improvements would be expected if the approaches were implemented in a compiled language such as C/C++, but the relative differences between the techniques would be expected to be similar.

The scenario tracks a time varying, randomly generated number of targets, using 8 bearing-range sensors. The targets are 2D planar position and velocity,  $x = [p_x, \dot{p}_x, p_y, \dot{p}_y]^T$ . Bearing-range measurements,  $z^{(s)} = [\alpha^{(s)}, r^{(s)}]^T$ , were observed according to the single-target measurement likelihood  $g(z^{(s)}|x) = \mathcal{N}(z^{(s)}; h^{(s)}(z^{(s)}, x^{(s)}), R^{(s)})$ , where  $x^{(s)} = [p_x^{(s)}, p_y^{(s)}]^T$ ,  $R^{(s)} = \text{diag}(0.25^2, 10^2)$ , and

$$h_\alpha^{(s)}(x, x^{(s)}) = \arctan\left(\frac{p_x^{(s)} - p_x}{p_y^{(s)} - p_y}\right) \quad (13)$$

$$h_r^{(s)}(x, x^{(s)}) = \sqrt{(p_x^{(s)} - p_x)^2 + (p_y^{(s)} - p_y)^2}. \quad (14)$$

The number of targets birthed was sampled uniformly every 5 seconds, 0 – 3 targets were selected to be born. Target positions were uniformly sampled over the domain  $[0, 10000]$   $m$  and target speed was fixed at 50  $m/s$ . Target velocity heading was uniformly sampled in the domain  $[-\pi, \pi]$  radians.

The target dynamics followed a nearly constant velocity transition model  $f_+(x_+|x) = \mathcal{N}(x_+; (I_2 \otimes F)x, (I_2 \otimes G)w)$  with  $I_2$  as the  $2 \times 2$  identity matrix,  $\otimes$  being the Kronecker product, and

$$F = \begin{bmatrix} 1 & \Delta_t \\ 0 & 1 \end{bmatrix}, \quad G = \begin{bmatrix} \frac{\Delta_t^2}{2} \\ \Delta_t \end{bmatrix},$$

where  $\Delta_t$  was the discrete-time sampling interval [19]. The acceleration white noise was  $w = [5, 5]^T m/s^2$ . The simulations ran for 100 seconds with  $\Delta_t = 1$  second. Each target had a survival probability of  $p_s(x, l) = 0.99$ . The sensors used a constant detection probability of  $p_D^{(s)}(x, l) = 0.95$ . Clutter was modeled as Poisson distributed with intensity  $\kappa^{(s)}(\mathbb{Z}^{(s)}) = \lambda_c^{(s)}\mathcal{U}(\mathbb{Z}^{(s)})$  where  $\mathcal{U}(\mathbb{Z}^{(s)})$  is the uniform distribution over  $\mathbb{Z}^{(s)}$ , and  $\lambda_c^{(s)} = 10$ .

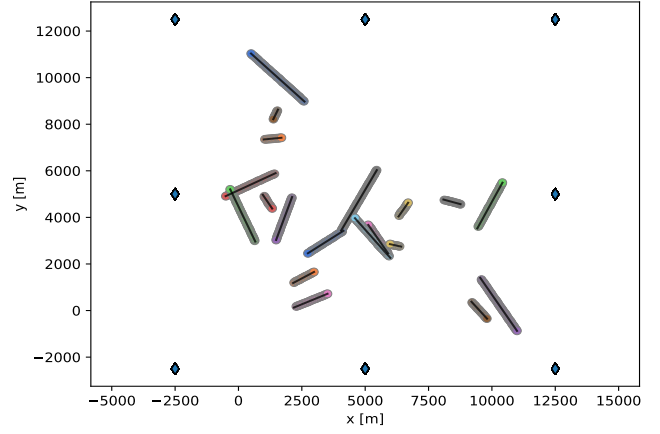


Fig. 1. Single observation of target trajectories (black lines), and an example labeled state estimate result (colored circles). Diamonds represent the locations of the bearing-range sensors.

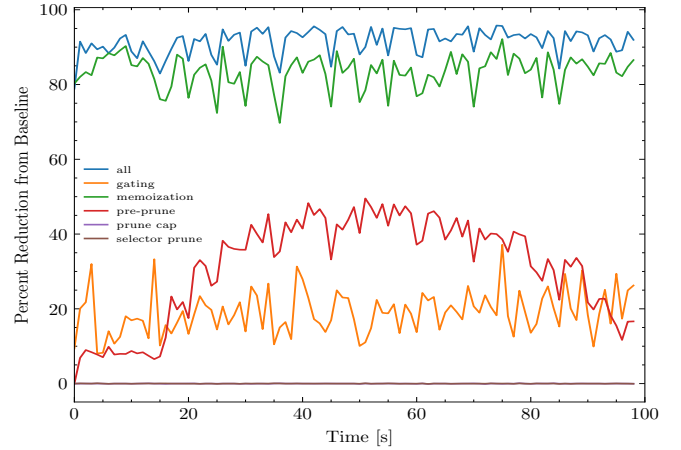


Fig. 2. Percent change (decrease) in number of evaluations per method compared to baseline.

The adaptive birth Gibbs sampler used the multi-short run sample generation architecture as described in [16], with 20 chains and a chain length of 5. The initial condition for all chain simulations was the all-miss-detected measurement tuple. The maximum birth probability was  $r_{B,max} = 1.0$ , and the expected birth rate was  $\lambda_{B,+} = 0.5$ . The maximum association probability was  $\tau = 0.01$ .

#### A. Results

The overall runtime reductions are shown in Table I. Each method was run individually and compared to the baseline run time, and one simulation was included where all efficiency improvements were combined and compared to the baseline. Run times were recorded using wall clock time. Each percent reduction was calculated as  $\frac{r_1 - r_2}{r_1} * 100$ , where  $r_1$  is the baseline simulation's runtime and  $r_2$  is runtime of interest.

1) *Associated Measurement Pre-pruning*: In this simulation, we used a small association threshold of  $\tau_A = 0.0001$

TABLE I  
PERCENT REDUCTION IN RUNTIME COMPARED TO BASELINE.

| Feature         | Reduction (%) |
|-----------------|---------------|
| Pre-prune       | 33.45         |
| Gating          | 11.17         |
| Memoization     | 83.82         |
| Prune and Cap   | 2.09          |
| Sample skipping | 0.90          |
| All on          | 91.57         |

to demonstrate the effectiveness of the approach when only using measurements that are highly likely to be unassociated with existing targets. In practice, this value can be increased to allow for birthing from more ambiguous measurements if necessary.

The measurement pre-pruning method prunes already associated measurements, once track labels are established. This resulted in an average of 4,530 values of  $\bar{\psi}_Z^J$  per timestep that did not need to be computed, which is a 31.4% average reduction in number of evaluations, as shown in Figure 2. Overall, this translated to a runtime reduction of 33.45%, as shown in Table I.

2) *Measurement Gating*: For this simulation, we used a very simple Euclidean distance coarse gate with  $\tau_G = 500m$  to highlight the benefit of this procedure. On average, this resulted in 2,588 values of  $\bar{\psi}_Z^J$  that did not need to be computed at each timestep, an average reduction in the number of evaluations by 19.06%, as shown in Figure 2. Including the time it took to construct the gate matrix, this translated to a total runtime reduction of 11.17%, as shown in Table I.

3) *Memoization*: Memoization resulted in the most significant reduction in runtime from any single proposed method. The birth Gibbs sampler explores many repeated solutions, so saving these results instead of recalculating them produces an 83.82% reduction in runtime, as shown in Table I. For this simulation, an average of 11,275  $\bar{\psi}_Z^J$  evaluations were saved per timestep, an 83.53% reduction in total  $\bar{\psi}_Z^J$  evaluations, as shown in Figure 2.

4) *Prune and Cap Birth LMB*: To prune, we drop any track labels with an  $r_{1+} < 0.001$ . We cap our track label cardinality to 100.

Pruning and capping resulted in a 2.09% reduction in runtime, as shown in Table I. This shows there are some savings in this scenario, but we would see a more significant runtime reduction in a scenario that require more Gibbs samples.

5) *Missed Detection Sample Skipping*: In this simulation, a maximum number of missed detections was set to 4.

As seen in Figure 3, a significant number of measurement tuples were skipped even for this sensing application with unambiguous measurement inverse functions and a relatively low clutter rate. Despite the large number of skipped components, this only translated to a total runtime reduction of 0.90%, as shown in Table I. This is because the measurement tuples themselves were still considered in the sampling procedure, meaning  $\bar{\psi}_Z^J$  was computed every time it was evaluated by the Gibbs sampler.

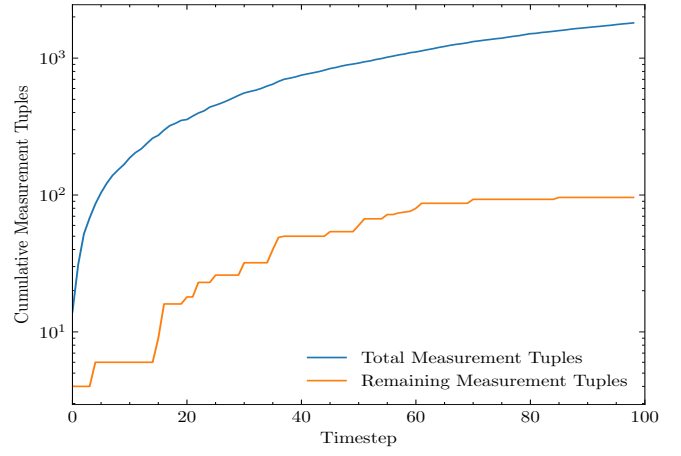


Fig. 3. Measurement tuples skipped when in labeled Bernoulli components construction.

6) *Overall Reduction*: This paper focuses on adaptive birth efficiency improvements, in our baseline simulation the birth procedure was 98% of the runtime. After combining the above techniques to improve the runtime of the birth procedure, we see an overall simulation runtime reduction of 91.57%, with an average of 91.54% reduction in evaluations, as shown in Figure 2.

To verify these efficiency improvements don't affect tracking performance, we use the Optimal Subpattern Assignment (OSPA)(2) [20] to compare results, as shown in Figure 4. The OSPA(2) metric was computed using a distance cutoff of 200, a distance order of 1.0, a sliding window length of 5 and an expanding window weight power of 0. There is some small variation in tracking performance with each proposed method, but this is expected because each method modifies or truncates different components of the birth procedure and this is not a Monte Carlo analysis. The overall tracking performance stayed consistent, while the runtime is significantly reduced.

## V. CONCLUSION

This paper provided five methods for improving the runtime of the multi-sensor measurement adaptive birth procedure. There are two main components of the adaptive birth procedure where we focused on reducing runtime. The first was the Gibbs sampler, which is a large driver of the high runtime. The second is during construction of the truncated birth LMB, which is a smaller proportion of computations, but can still result in reducing overall runtime. Some methods are more application specific and would see more improvement in scenarios with less accurate sensors, or scenarios with more clutter. Overall these efficiency improvements significantly reduce runtime in a variety of scenarios. This paper provided benchmarking for what method to apply, depending on where the birth procedure bottleneck is in a given scenario.

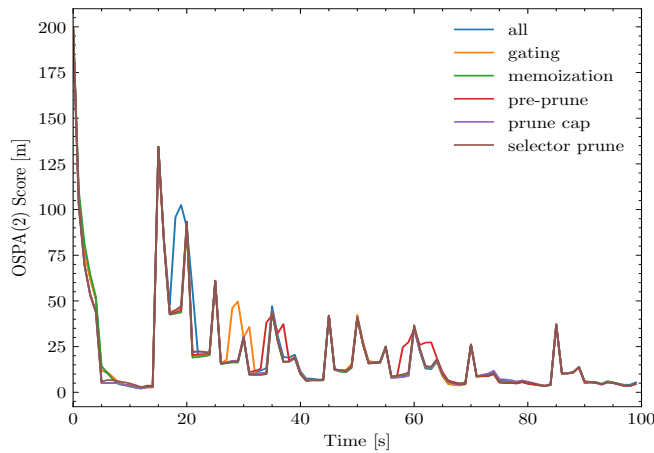


Fig. 4. Negligible variation in OSPA(2) performance with efficiency suggestions implemented.

## REFERENCES

- [1] S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Syst.* Norwood, MA, USA: Artech House, 1999.
- [2] Y. Bar-Shalom, F. Daum, and J. Huang, "The probabilistic data association filter," *IEEE Control Syst. Mag.*, vol. 29, no. 6, pp. 82–100, 2009.
- [3] S. Blackman, "Multiple hypothesis tracking for multiple target tracking," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 19, no. 1, pp. 5–18, Jan 2004.
- [4] F. Meyer, T. Kropfreiter, J. L. Williams, R. Lau, F. Hlawatsch, P. Braca, and M. Z. Win, "Message passing algorithms for scalable multitarget tracking," *Proc. IEEE*, vol. 106, no. 2, pp. 221–259, Feb 2018.
- [5] R. Mahler, *Statistical Multisource-Multitarget Information Fusion*. Norwood, MA, USA: Artech House, 2007.
- [6] —, *Advances in Statistical Multisource-Multitarget Information Fusion*. Norwood, MA, USA: Artech House, 2014.
- [7] B.-N. Vo, M. Mallick, Y. Bar-Shalom, S. Coraluppi, R. Mahler, and B.-T. Vo, "Multitarget tracking," *Wiley Encyclopedia of Elect. and Electron. Eng.*, Sep. 2015.
- [8] B.-T. Vo and B.-N. Vo, "Labeled random finite sets and multi-object conjugate priors," *IEEE Trans. Signal Process.*, vol. 61, no. 13, pp. 3460–3475, 2013.
- [9] B.-N. Vo, B.-T. Vo, and D. Phung, "Labeled random finite sets and the Bayes multi-target tracking filter," *IEEE Trans. Signal Process.*, vol. 62, no. 24, pp. 6554–6567, 2014.
- [10] S. Reuter, B.-T. Vo, B.-N. Vo, and K. Dietmayer, "The labeled multi-Bernoulli filter," *IEEE Trans. Signal Process.*, vol. 62, no. 12, pp. 3246–3260, 2014.
- [11] S. Reuter, A. Danzer, M. Stübler, A. Scheel, and K. Granström, "A fast implementation of the labeled multi-Bernoulli filter using Gibbs sampling," in *Proc. IEEE Intell. Veh. Symp.*, 2017, pp. 765–772.
- [12] B.-N. Vo, B.-T. Vo, and M. Beard, "Multi-sensor multi-object tracking with the generalized labeled multi-Bernoulli filter," *IEEE Trans. Signal Process.*, vol. 67, no. 23, pp. 5952–5967, 2019.
- [13] B. Ristic, D. Clark, B.-N. Vo, and B.-T. Vo, "Adaptive target birth intensity for PHD and CPHD filters," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 48, no. 2, pp. 1656–1668, 2012.
- [14] S. Lin, B.-T. Vo, and S. E. Nordholm, "Measurement driven birth model for the generalized labeled multi-Bernoulli filter," in *Proc. Int. Conf. Control, Automat. and Inf. Sci.*, 2016, pp. 94–99.
- [15] A. Trezza, D. J. Bucci, and P. K. Varshney, "Multi-sensor joint adaptive birth sampler for labeled random finite set tracking," *IEEE Trans. Signal Process.*, Feb. 2022.
- [16] —, "On gibbs sampling architecture for labeled random finite sets multi-object tracking," in *Proc. IEEE 26th Int. Conf. Inf. Fusion*, 2023, pp. 1–8.
- [17] B.-N. Vo, B.-T. Vo, and H. G. Hoang, "An efficient implementation of the generalized labeled multi-Bernoulli filter," *IEEE Trans. Signal Process.*, vol. 65, no. 8, pp. 1975–1987, 2016.
- [18] A. Murray, A. Withers, A. Trezza, and D. J. Bucci, "Multi-sensor adaptive birth for labeled RFS filters using bistatic range-only measurements," in *Proc. IEEE Conf. Radar*, 2023, pp. 1–6.
- [19] X. R. Li and V. P. Jilkov, "Survey of maneuvering target tracking. part I : Dynamic models," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 39, no. 4, pp. 1333–1364, 2003.
- [20] M. Beard, B.-T. Vo, and B.-N. Vo, "A solution for large-scale multi-object tracking," *IEEE Trans. Signal Process.*, vol. 68, pp. 2754–2769, 2020.