

The Role of Bounded Fields-of-View and Negative Information in Finite Set Statistics (FISST)

Keith LeGrand  and Silvia Ferrari 

Abstract—The role of negative information is particularly important to search-detect-track problems in which the number of objects is unknown *a priori*, and the size of the sensor field-of-view is far smaller than that of the region of interest. This paper presents an approach for systematically incorporating knowledge of the field-of-view geometry and position and object inclusion/exclusion evidence into object state densities and random finite set multi-object cardinality distributions. The approach is derived for a representative set of multi-object distributions and demonstrated through a sensor planning problem involving a multi-Bernoulli process with up to one-hundred potential targets.

Index Terms—Bounded field-of-view, Gaussian mixtures, Gaussian splitting, random finite set theory

I. INTRODUCTION

Random finite set (RFS) theory has been proven a highly effective framework for developing and analyzing tracking and sensor planning algorithms in applications involving an unknown number of multiple targets (objects) [1]–[7]. To date, however, little attention has been given to the role that bounded fields-of-view (FoVs) and negative information play in the finite set statistics (FISST) recursive updates for assimilating measurements, or lack thereof, into multi-object probability distributions. Existing algorithms typically terminate object tracks after the object is believed to have left the sensor FoV. While this approach is suitable when the FoV doubles as the tracking region of interest (ROI), it is inapplicable when the sensor FoV is much smaller than the ROI and, thus, must be moved or positioned so as to maximize information value [8]–[11].

Knowledge of object presence inside the FoV is powerful evidence that can be incorporated to update the object probability density function (pdf) in a Bayesian framework. For example, the absence of detections, referred to as *negative information* may suggest that the object state resides outside the FoV [12], [13]. In contrast, binary-type sensors may indicate that the object is inside the sensor FoV but provide no further localization information. Particle-based filtering algorithms can accommodate such measurements but require a large number of particles and are computationally expensive. Another approach [13] uses Gaussian mixtures (GMs) to model both the object pdf and the state-dependent probability of detection function. Though GMs efficiently model some detection probability functions, other simple functions, such

Keith LeGrand and Silvia Ferrari are with the Laboratory for Intelligent Systems and Controls (LISC), Sibley School of Mechanical and Aerospace Engineering, Cornell University, Ithaca, New York, United States. This work was supported in part by Office of Naval Research Grant N0014-19-1-2266

as uniform probability over a 3D FoV, require problematically large numbers of components. Recently, a method based on intermediate particle representation and the expectation maximization (EM) algorithm has been proposed for forming GMs inside and outside the FoV [14].

This paper presents relevant bounded FoV statistics both in the form of state densities and cardinality probability mass functions (pmfs). Section III presents a deterministic method that partitions a GM state density along FoV bounds through recursive Gaussian splitting. In Section IV, FoV object cardinality pmfs are derived for some of the most commonly encountered RFS distributions. Section V presents an application of bounded FoV statistics to a sensor placement problem, and conclusions are made in Section VI.

II. PROBLEM FORMULATION AND ASSUMPTIONS

This paper considers the incorporation of bounded FoV information into algorithms for (multi-)object tracking and sensor planning when the number of objects is unknown and time-varying. As shown in [15], the sensor FoV can be defined as the compact subset $\mathcal{S}(\mathbf{q}) \subset \mathbb{X}_p$, where \mathbb{X}_p is a subspace of the single-object state space \mathbb{X} . Typically, \mathbb{X}_p represents the object position space and, thus, vectors and densities associated with \mathbb{X}_p are referred to as “position” quantities in this paper. However, the methods described in the following sections are applicable to any arbitrary subspace of \mathbb{X} . In general, the FoV is a function of the sensor state \mathbf{q} , which, for example, may consist of the sensor position, orientation, and zoom level. However, for notational simplicity this dependence is omitted in the remainder of this paper.

Now, let the object state \mathbf{x} consist of the kinematic variables that are to be estimated from data via filtering, such as the object position, velocity, turn rate, etc. Then, the single-object pdf is denoted by $p(\mathbf{x})$. Letting $\mathbf{x}_p = \text{proj}_{\mathbb{X}_p} \mathbf{x}$ denote the state elements that correspond to \mathbb{X}_p , an object’s presence inside the FoV can be expressed by the generalized indicator function

$$1_{\mathcal{S}}(\mathbf{x}) = \begin{cases} 1, & \text{if } \mathbf{x}_p \in \mathcal{S} \\ 0, & \text{otherwise} \end{cases}$$

The number of objects and their kinematic states are unknown *a priori*, but can be assumed to consist of discrete and continuous variables, respectively. The collection of object states is modeled as an RFS X or labeled random finite set (LRFS) \tilde{X} , where the single-object labeled state $\tilde{x} = (\mathbf{x}, \ell) \in \mathbb{X} \times \mathbb{L}$ consists of a kinematic state vector \mathbf{x} and unique discrete label ℓ . It is assumed that the prior multi-object distribution is known, e.g., from the output of a multi-object filter, and

modeled using either the RFS density $f(X)$ or LRFS density $\mathring{f}(\mathring{X})$.

Throughout this paper, single-object states are represented by lowercase letters (e.g. \mathbf{x} , $\hat{\mathbf{x}}$), while multi-object states are represented by italic uppercase letters (e.g. X , \mathring{X}). Bold lowercase letters are used to denote vectors (e.g. \mathbf{x} , \mathbf{z}) and bold uppercase letters are used to denote matrices (e.g. \mathbf{P} , $\mathbf{\Lambda}$). The accent “ $\mathring{\cdot}$ ” is used to distinguish labeled states and functions (e.g. \mathring{f} , $\mathring{\hat{\mathbf{x}}}$, \mathring{X}) from their unlabeled equivalents. Spaces are represented by blackboard bold symbols (e.g. \mathbb{X} , \mathbb{L}).

Knowledge of object presence inside the FoV is powerful evidence that can be used to update the object state pdf in a Bayesian framework. As an example, the single-object state pdf conditioned on its presence inside the FoV can be expressed as

$$p(\mathbf{x} | \mathcal{S}) \propto 1_{\mathcal{S}}(\mathbf{x})p(\mathbf{x}) \triangleq p_{\mathcal{S}}(\mathbf{x}) \quad (1)$$

Similarly, knowledge of object presence outside of the FoV or equivalently, in the complement set $\mathcal{C}(\mathcal{S}) = \mathbb{X}_p \setminus \mathcal{S}$, can be incorporated, such that

$$p(\mathbf{x} | \mathcal{C}(\mathcal{S})) \propto (1 - 1_{\mathcal{S}}(\mathbf{x}))p(\mathbf{x}) \triangleq p_{\mathcal{C}(\mathcal{S})}(\mathbf{x}) \quad (2)$$

Equation (1) can be used to model occupancy measurements (e.g. the object was detected somewhere in the FoV), and when integrated with respect to \mathbf{x} , gives the probability that the object is inside the FoV. Equation (2) is required to properly incorporate the negative information that an object is not inside \mathcal{S} . For mathematical conciseness, throughout this paper, object *presence* and *absence* are considered rather than object *detection* and *non-detection*. However, Equations (1) and (2) are easily modified to account for detection probabilities. For example, the event “object not detected,” denoted by $\neg D$, is incorporated through the application of Bayes’ rule, such that

$$p(\mathbf{x} | \neg D) \propto (1 - 1_{\mathcal{S}}(\mathbf{x})p_D(\mathbf{x}))p(\mathbf{x})$$

where $p_D(\mathbf{x})$ is the state-dependent probability of detection.

In RFS-based tracking, single-object densities are, in fact, parameters of the higher-dimensional multi-object density. Non-Gaussian single-object state densities are often modeled using GMs because they admit closed-form approximations to the multi-object Bayes recursion under certain conditions [2], [16]. Therefore, in this paper, it is assumed that single-object densities (which are parameters of the higher dimensional multi-object density) are parameterized as

$$p(\mathbf{x}) = \sum_{\ell=1}^L w^{(\ell)} \mathcal{N}(\mathbf{x}; \mathbf{m}^{(\ell)}, \mathbf{P}^{(\ell)})$$

where L is the number of GM components and $w^{(\ell)}$, $\mathbf{m}^{(\ell)}$, and $\mathbf{P}^{(\ell)}$ are the weight, mean, and covariance matrix of the ℓ^{th} component, respectively.

The multi-object exponential notation,

$$h^A \triangleq \prod_{a \in A} h(a)$$

where $h^{\emptyset} \triangleq 1$, is adopted throughout. For multivariate functions, the dot (\cdot) denotes the argument of the multi-object exponential, e.g.:

$$[g(a, \cdot, c)]^B \triangleq \prod_{b \in B} g(a, b, c)$$

The exponential notation is used to denote the product space, $\mathbb{X}^n = \prod^n(\mathbb{X} \times)$. Exponents of RFSs are used to denote RFSs of a given cardinality, e.g. $|X^n| = n$, where n is the cardinality. The operator $\text{diag}(\cdot)$ places its input on the diagonal of the zero matrix. The Kronecker delta function is defined as

$$\delta_{\mathbf{a}}(\mathbf{b}) \triangleq \begin{cases} 1, & \text{if } \mathbf{b} = \mathbf{a} \\ 0, & \text{otherwise} \end{cases}$$

for any two arbitrary vectors $\mathbf{a}, \mathbf{b} \in \mathbb{R}^n$. The inner product of two integrable functions $f(\cdot)$ and $g(\cdot)$ is denoted by

$$\langle f, g \rangle = \int f(\mathbf{x})g(\mathbf{x})d\mathbf{x}$$

III. GM APPROXIMATION OF FOV-PARTITIONED DENSITIES

This section presents a method for partitioning the object pdf into truncated densities $p_{\mathcal{S}}(\mathbf{x})$ and $p_{\mathcal{C}(\mathcal{S})}(\mathbf{x})$, with supports $\mathbb{X} \setminus \mathcal{C}(\mathcal{S})$ and $\mathbb{X} \setminus \mathcal{S}$, respectively. Focus is given to the single-object state density with the awareness that the method is naturally extended to RFS multi-object densities and algorithms that use GM parameterization. Consider the single-object density $p(\mathbf{x})$ parameterized by an L -component GM, as follows:

$$p(\mathbf{x}) = p_{\mathcal{S}}(\mathbf{x}) + p_{\mathcal{C}(\mathcal{S})}(\mathbf{x}) = \sum_{\ell=1}^L w^{(\ell)} \mathcal{N}(\mathbf{x}; \mathbf{m}^{(\ell)}, \mathbf{P}^{(\ell)})$$

One simple approximation of densities partitioned according to the discrete FoV geometry, referred to as FoV-partitioned densities hereon, is found by evaluating the indicator function at the component means [17], i.e.:

$$p_{\mathcal{S}}(\mathbf{x}) \approx \sum_{\ell=1}^L w^{(\ell)} 1_{\mathcal{S}}(\mathbf{m}^{(\ell)}) \mathcal{N}(\mathbf{x}; \mathbf{m}^{(\ell)}, \mathbf{P}^{(\ell)}) \quad (3)$$

$$p_{\mathcal{C}(\mathcal{S})}(\mathbf{x}) \approx \sum_{\ell=1}^L w^{(\ell)} (1 - 1_{\mathcal{S}}(\mathbf{m}^{(\ell)})) \mathcal{N}(\mathbf{x}; \mathbf{m}^{(\ell)}, \mathbf{P}^{(\ell)}) \quad (4)$$

By this approach, components whose means lie inside (outside) the FoV are preserved (pruned), or vice versa.

The accuracy of this mean-based partition approximation depends strongly on the resolution of the GM near the geometric boundaries of the FoV. Even though the mean of a given component lies inside (outside) the FoV, a considerable proportion of the probability mass may lie outside (inside) the FoV, as is illustrated in Figure 1a. Therefore, the amount of FoV overlap, along with the weight of the component, determines the accuracy of the approximations (Eq. (3)-(4)). To that end, the algorithm presented in the following subsection iteratively resolves the GM near FoV bounds by recursively splitting Gaussian components that overlap the FoV bounds.

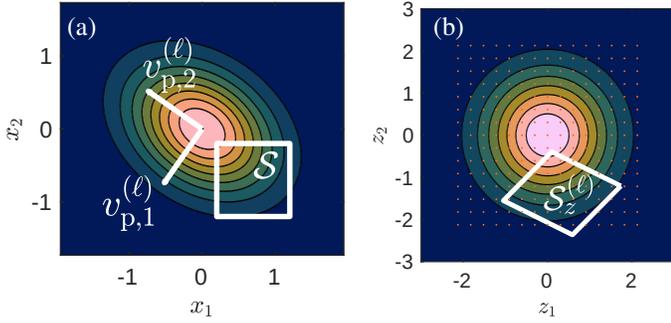


Fig. 1: Original component density and FoV with covariance eigenvectors overlaid (a), and same component density and FoV after change of variables (b).

A. Gaussian Splitting Algorithm

The Gaussian splitting algorithm presented in this subsection forms an FoV-partitioned GM approximation of the original GM by using a higher number of components near the FoV boundaries, ∂S , so as to improve the accuracy of the mean-based partition.

Consider for simplicity a two-dimensional example in which the original GM, $p(\mathbf{x})$, has a single component whose mean lies outside the FoV, as shown in Figure 1a. The algorithm first applies a change of variables $\mathbf{x} \mapsto \mathbf{z}$ such that $p(\mathbf{z})$ is symmetric, has zero mean and unit variance. The basis vectors of the space $\mathbb{Z} \ni \mathbf{z}$ correspond to the principal directions of the component's positional covariance. The same change of variables is applied to the FoV bounds (Figure 1b).

A pre-computed point grid is then tested for inclusion in the transformed FoV in order to decide whether to split the component, and if so, along which principal direction. For each new split component, the process is repeated—if a new component significantly overlaps the FoV boundaries, it may be further split into several smaller components, as illustrated in Figure 2b. This process is repeated until stopping criteria are satisfied. After the GM splitting terminates, $p_S(\mathbf{x})$ and $p_{C(S)}(\mathbf{x})$ are approximated by the mean-based partition, as illustrated in Figure 3.

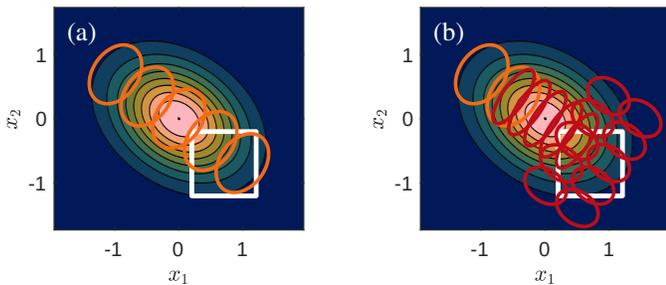


Fig. 2: 1σ contours of components after first split operation (a), and second split operation (b), where components formed in the second operation are shown in red.

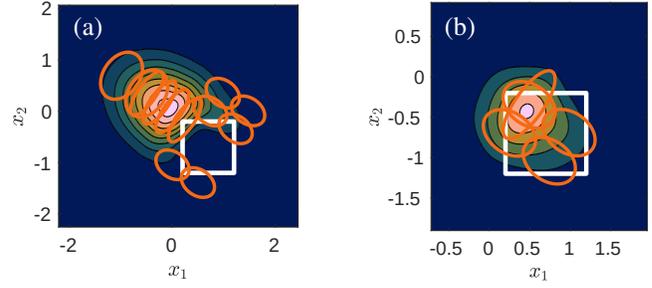


Fig. 3: The GM approximations to densities $p_{C(S)}(\mathbf{x})$ (a), and $p_S(\mathbf{x})$ (b) after two iterations of splitting.

B. Univariate Splitting Library

Splitting is performed efficiently by utilizing a pre-generated library of optimal split parameters for the univariate standard Gaussian $q(x)$, as first proposed in [18] and later generalized in [19]. The univariate split parameters are retrieved at runtime and applied to arbitrary multivariate Gaussian densities via scaling, shifting, and covariance diagonalization.

Generation of the univariate split library is performed by minimizing the cost function

$$J = L_2(q||\tilde{q}) + \lambda \tilde{\sigma}^2 \quad \text{s.t.} \quad \sum_{j=1}^R \tilde{w}^{(j)} = 1$$

where

$$\tilde{q}(x) = \sum_{j=1}^R \tilde{w}^{(j)} \mathcal{N}(x; \tilde{m}^{(j)}, \tilde{\sigma}^2)$$

for different parameter values R, λ . The regularization term λ balances the importance of using smaller standard deviations $\tilde{\sigma}$ with the minimization of the L_2 distance. While other distance measures may be used, the L_2 distance is attractive because it can be computed in closed form for GMs [19].

C. Change of Variables

The determination of which components should be split, and if so, along which direction, is simplified by first establishing a change of variables. For each component with index ℓ , the change of variables $\mathbf{h}^{(\ell)} : \mathbb{X}_p \mapsto \mathbb{Z}$ is applied as follows:

$$\mathbf{z} = \mathbf{h}^{(\ell)}(\mathbf{x}_p^{(\ell)}; \mathbf{m}_p^{(\ell)}, \mathbf{P}_p^{(\ell)}) \triangleq (\mathbf{\Lambda}_p^{(\ell)})^{-1/2} \mathbf{V}_p^{(\ell)T} (\mathbf{x}_p - \mathbf{m}_p^{(\ell)}) \quad (5)$$

where

$$\mathbf{V}_p^{(\ell)} = [\mathbf{v}_{p,1}^{(\ell)} \quad \cdots \quad \mathbf{v}_{p,n_p}^{(\ell)}]$$

$$(\mathbf{\Lambda}_p^{(\ell)})^{-1/2} = \text{diag} \left(\left[\frac{1}{\sqrt{\lambda_{p,1}^{(\ell)}}} \quad \cdots \quad \frac{1}{\sqrt{\lambda_{p,n_p}^{(\ell)}}} \right] \right)$$

and $\mathbf{m}_p^{(\ell)}$ is the n_p -element position portion of the full-state mean, and the columns of $\mathbf{V}_p^{(\ell)}$ are the normalized eigenvectors of the position-marginal covariance $\mathbf{P}_p^{(\ell)}$, with $\mathbf{v}_{p,i}^{(\ell)}$

corresponding to the i^{th} eigenvalue $\lambda_{p,i}^{(\ell)}$. In the transformed space,

$$p_z(\mathbf{z}) = \mathcal{N}(\mathbf{z}; \mathbf{0}, \mathbf{I})$$

Note that, in defining the transformation over \mathbb{X}_p , the same transformation can be applied to the FoV, such that

$$\mathcal{S}_z^{(\ell)} = \{\mathbf{h}^{(\ell)}(\mathbf{x}_p; \mathbf{m}_p^{(\ell)}, \mathbf{P}_p^{(\ell)}) : \mathbf{x}_p \in \mathcal{S}\} \quad (6)$$

In \mathbb{Z} , the Euclidean distances to boundary points of $\mathcal{S}_z^{(\ell)}$ can be interpreted as probabilistically normalized distances. In fact, the Euclidean distance of a point \mathbf{z} from the origin in \mathbb{Z} corresponds exactly to the Mahalanobis distance between the corresponding point \mathbf{x}_p and the original position-marginal component.

D. Component Selection and Collocation Points

Components are selected for splitting if they have sufficient weight and significant statistical overlap of the FoV boundaries ($\partial\mathcal{S}$). For components of sufficient weight, the change of variables is applied to the FoV to obtain $\mathcal{S}_z^{(\ell)}$ per Equation (6). The overlap of the original component on \mathcal{S} is then equivalent to the overlap of the standard Gaussian distribution on $\mathcal{S}_z^{(\ell)}$, which is quantified using a grid of collocation points on \mathbb{Z} . Define a uniform grid of collocation points $\{\bar{\mathbf{z}}_{i_1, \dots, i_{n_p}}\}$ on \mathbb{Z} such that

$$\begin{aligned} \bar{\mathbf{z}}_{i_1, \dots, i_{n_p}} &= [\bar{z}_1(i_1) \dots \bar{z}_{n_p}(i_{n_p})]^T \\ \bar{z}_j(i_j) &= -\zeta + 2\zeta \left(\frac{i_j - 1}{N - 1} \right), \quad i_j = 1, \dots, N \end{aligned}$$

where ζ is a user-specified bound for the grid and N is the number of points per dimension. An inclusion variable is defined as

$$d_{i_1, \dots, i_{n_p}}^{(\ell)} \triangleq 1_{\mathcal{S}_z^{(\ell)}}(\bar{\mathbf{z}}_{i_1, \dots, i_{n_p}})$$

A function $s_{\mathcal{S}_z^{(\ell)}}(\cdot)$ is established to mark total inclusion or total exclusion as

$$s_{\mathcal{S}_z^{(\ell)}}(\bar{\mathbf{z}}_{i_1, \dots, i_{n_p}}) = \prod_{i_1, \dots, i_{n_p}} \delta_{d_{i_1, \dots, 1}^{(\ell)}}(d_{i_1, \dots, i_{n_p}}^{(\ell)})$$

which is equal to unity if all grid points lie inside of $\mathcal{S}_z^{(\ell)}$ or all grid points lie outside of $\mathcal{S}_z^{(\ell)}$, and is zero otherwise. If either all or no points are included, no splitting is required. Otherwise, the component is marked for splitting.

E. Positional Split Direction

Rather than split a component along each of its principal directions, a more judicious selection can be made by limiting split operations to a single direction (per component) per recursion. Thus, by performing one split per component per recursion, the component selection criteria are re-evaluated, reducing the overall number of components generated. In the aforementioned two-dimensional example, only a subset of new components generated from the first split are selected for further splitting as shown in Figure 2b.

The split direction is chosen based on the relative geometry of the FoV, and thus positional vectors are of interest. Choosing the best positional split direction is a challenging problem. Ideally, splitting along the chosen direction should minimize the number of splits required in the next iteration as well as improve the accuracy of the partition approximation applied after the final iteration. The computational complexity of exhaustive optimization of the split direction would likely negate the computational efficiency of the overall algorithm. Instead, to minimize the number of splits required in the next iteration, the positional split direction is chosen as the direction that is orthogonal to the most grid planes of consistent inclusion/exclusion. The plane of constant $z_j = \bar{z}_j(i_j)$ is consistently inside or consistently outside if

$$s_j^{(\ell)}(i_j) = \prod_{i_1, \dots, i_{j-1}, i_{j+1}, i_{n_p}} \delta_{d_{i_1, \dots, i_j, \dots, 1}^{(\ell)}}(d_{i_1, \dots, i_j, \dots, i_{n_p}}^{(\ell)})$$

is equal to unity. The optimal positional split direction is then given by the eigenvector \mathbf{v}_{p,j^*} , where the optimal eigenvector index is found as

$$j^* = \arg \max_j \left(\sum_{i_j} s_j^{(\ell)}(i_j) \right) \quad (7)$$

For notational simplicity, the implicit dependence of j^* on the component index ℓ is omitted. For example, referring back to the two-dimensional example and Figure 1b, there are more rows than columns that are consistently inside or outside the transformed FoV, and thus $j^* = 2$ is chosen as the desired positional split direction index. In the case where multiple maxima exist, the eigenvector with largest eigenvalue is selected, which corresponds to the direction of largest variance among the maximizing eigenvectors.

F. Multivariate Split of Full-state Component

Gaussian splitting must be performed along the principal directions of the full-state covariance. The general multivariate split approximation, splitting along the k^{th} eigenvector $\mathbf{v}_k^{(\ell)}$ is given by [19]

$$w^{(\ell)} \mathcal{N}(\mathbf{x}; \mathbf{m}^{(\ell)}, \mathbf{P}^{(\ell)}) \approx \sum_{j=1}^R w^{(\ell,j)} \mathcal{N}(\mathbf{x}; \mathbf{m}^{(\ell,j)}, \mathbf{P}^{(\ell,j)}) \quad (8)$$

where

$$\begin{aligned} w^{(\ell,j)} &= \tilde{w}^{(j)} w^{(\ell)}, \quad \mathbf{m}^{(\ell,j)} = \mathbf{m}^{(\ell)} + \sqrt{\lambda_k^{(\ell)}} \tilde{m}^{(j)} \mathbf{v}_k^{(\ell)} \\ \mathbf{P}^{(\ell,j)} &= \mathbf{V}^{(\ell)} \mathbf{\Lambda}^{(\ell)} \mathbf{V}^{(\ell)T}, \quad \mathbf{\Lambda}^{(\ell)} = \text{diag}([\lambda_1 \dots \tilde{\sigma}^2 \lambda_k \dots \lambda_n]) \end{aligned}$$

and the optimal univariate split parameters $\tilde{w}^{(j)}$, $\tilde{m}^{(j)}$, and $\tilde{\sigma}$ are found from the pre-computed split library given the number of split components R and regularization parameter λ . In general, the positional components of the full-state eigenvectors will not perfectly match the desired positional split vector due to correlations between the states. Rather,

the actual full-state split is performed along $\mathbf{v}_{k^*}^{(\ell)}$, where the optimal eigenvector index is found according to

$$k^* = \arg \max_k |[\mathbf{v}_{p,j^*}^{(\ell)T} \mathbf{0}^T] \mathbf{v}_k^{(\ell)}| \quad (9)$$

where, without loss of generality, a specific state convention is assumed such that position states are first in element order.

G. Recursion and Role of Negative Information

The splitting procedure is applied recursively, as detailed in Algorithm 1. The recursion is terminated when no remaining components satisfy the criteria for splitting. Each recursion further refines the GM near the FoV bounds to improve the approximations of Equations (3)-(4). However, because a Gaussian component's split approximation (Eq. 8) does not perfectly replicate the original component, a small error is induced with each split. Given enough recursions, this error may become dominant. In the authors' experience, the recursion is terminated well before the cumulative split approximation error dominates.

Algorithm 1: `split_for_fov` ($\{w^{(\ell)}, \mathbf{m}^{(\ell)}, \mathbf{P}^{(\ell)}\}_{\ell=1}^L$, w_{\min} , \mathcal{S} , R , λ)

```

split ← {}, no_split ← {}
if  $L = 0$  then
  return split
end if
for  $\ell = 1, \dots, L$  do
  if  $w^{(\ell)} < w_{\min}$  then
    add  $\{w^{(\ell)}, \mathbf{m}^{(\ell)}, \mathbf{P}^{(\ell)}\}$  to no_split
    continue
  end if
  Compute  $\mathcal{S}_z^{(\ell)}$  accord. to Eq. (6)
  if  $s_{\mathcal{S}_z^{(\ell)}}(\bar{\mathbf{z}}_{i_1, \dots, i_{n_p}}) = 1$  then
    add  $\{w^{(\ell)}, \mathbf{m}^{(\ell)}, \mathbf{P}^{(\ell)}\}$  to no_split
  else
     $j^* \leftarrow$  Eq. (7),  $k^* \leftarrow$  Eq. (9)
     $\{w^{(\ell,j)}, \mathbf{m}^{(\ell,j)}, \mathbf{P}^{(\ell,j)}\}_{j=1}^R \leftarrow$  Eq. (8) with  $k = k^*$ 
    add  $\{w^{(\ell,j)}, \mathbf{m}^{(\ell,j)}, \mathbf{P}^{(\ell,j)}\}_{j=1}^R$  to split
  end if
end for
split ← split_for_fov(split,  $w_{\min}$ ,  $\mathcal{S}$ ,  $R$ ,  $\lambda$ )
return split  $\cup$  no_split

```

One of the many potential applications of the recursive algorithm presented in this section involves incorporating the evidence of non-detections, or negative information, in single- or multi-object filtering. To demonstrate, a single-object filtering problem with a bounded square FoV is considered where, in three subsequent sensor reports, no object is detected. The true object position and constant velocity are unknown but are distributed according to a known GM pdf at the first time step. As the initial pdf is propagated over time, the position-marginal pdf travels from left to right, as pictured in Figure 4. For simplicity, the probability of detection inside the FoV is

assumed equal to one. At each time step, the GM is refined by Algorithm 1 using $w_{\min} = 0.01$, $R = 3$, and $\lambda = 0.001$. Then, the mean-based partition approximation is applied (Eq. 4) and the updated filtering density is found (Eq. 2). By this approach, the number of components may increase over time but can be reduced as needed through component merging and pruning.

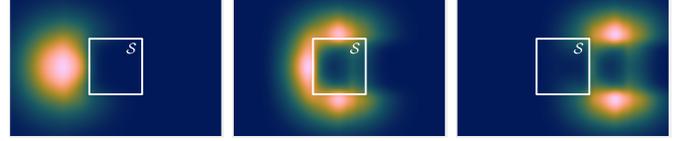


Fig. 4: Negative information, comprised of absence of detections inside the sensor FoV \mathcal{S} , is used to update the object pdf as the object moves across the ROI.

IV. FOV CARDINALITY DISTRIBUTION

This section presents pmfs for the cardinality of objects inside a bounded FoV \mathcal{S} given different multi-object workspace densities $f(\cdot)$. The Poisson, independent identically distributed cluster (i.i.d.c.), multi-Bernoulli (MB), and generalized labeled multi-Bernoulli (GLMB) distributions are considered in Subsections IV-A, IV-B, IV-C, and IV-D, respectively.

The probability of n objects existing inside FoV \mathcal{S} conditioned on X can be written in terms of the indicator function as

$$\rho_{\mathcal{S}}(n | X) = \sum_{X^n \subseteq X} [1_{\mathcal{S}}(\cdot)]^{X^n} [1 - 1_{\mathcal{S}}(\cdot)]^{X \setminus X^n} \quad (10)$$

where the summation is taken over all subsets $X^n \subseteq X$ with cardinality n . Given the RFS density $f(X)$, the FoV cardinality distribution is obtained via the set integral as

$$\rho_{\mathcal{S}}(n) = \int \rho_{\mathcal{S}}(n | X) f(X) \delta X$$

Expanding the integral,

$$\rho_{\mathcal{S}}(n) = \sum_{m=n}^{\infty} \frac{1}{m!} \int_{\mathbb{X}^m} \rho_{\mathcal{S}}(n | \{\mathbf{x}_1, \dots, \mathbf{x}_m\}) f(\{\mathbf{x}_1, \dots, \mathbf{x}_m\}) d\mathbf{x}_1 \cdots d\mathbf{x}_m \quad (11)$$

Remark: The results presented in this section can be trivially extended to express the predicted cardinality of object-originated *detections* Z (excluding false alarms) by noting that

$$\rho_{\mathcal{S}}(n_Z | X) = \sum_{X^n \subseteq X} [p_D(\cdot) 1_{\mathcal{S}}(\cdot)]^{X^n} [1 - p_D(\cdot) 1_{\mathcal{S}}(\cdot)]^{X \setminus X^n}$$

where $n_Z = |Z|$.

A. Poisson Distribution

The density of a Poisson-distributed RFS is

$$f(X) = e^{-N_X} [D]^X \quad (12)$$

where N_X is the global cardinality mean, and $D(\mathbf{x})$ is the probability hypothesis density (PHD), or intensity function,

of X , which is defined on the single-object space \mathbb{X} . One important property of the PHD is that its integral over a closed set on \mathbb{X} yields the expected number of objects within that set, i.e.

$$E[|X \cap T|] = \int_T D(\mathbf{x}) d\mathbf{x} \quad (13)$$

Proposition 1: Given a Poisson-distributed RFS with PHD $D(\mathbf{x})$ and global cardinality mean N_X , the cardinality of objects inside the field of view $\mathcal{S} \subseteq \mathbb{X}$ is distributed according to

$$\rho_{\mathcal{S}}(n) = \sum_{m=n}^{\infty} \frac{e^{-N_X}}{n!(m-n)!} \langle 1_{\mathcal{S}}, D \rangle^n \langle 1 - 1_{\mathcal{S}}, D \rangle^{m-n} \quad (14)$$

Proof: Substituting Equation (12) into Equation (11),

$$\rho_{\mathcal{S}}(n) = \sum_{m=n}^{\infty} \frac{1}{m!} e^{-N_X} \int_{\mathbb{X}^m} \sum_{X^n \subseteq X} [1_{\mathcal{S}}(\cdot)D(\cdot)]^{X^n} \cdot [(1 - 1_{\mathcal{S}}(\cdot))D(\cdot)]^{X \setminus X^n} d\mathbf{x}_1 \cdots d\mathbf{x}_m \quad (15)$$

The nested integrals of Equation (15) can be distributed, rewriting the second sum over n -cardinality index sets \mathcal{I}^n as

$$\rho_{\mathcal{S}}(n) = \sum_{m=n}^{\infty} \frac{1}{m!} e^{-N_X} \sum_{\mathcal{I}^n \subseteq \{1..m\}} \left[\int 1_{\mathcal{S}}(\mathbf{x}_{(\cdot)}) D(\mathbf{x}_{(\cdot)}) d\mathbf{x}_{(\cdot)} \right]^{\mathcal{I}^n} \cdot \left[\int (1 - 1_{\mathcal{S}}(\mathbf{x}_{(\cdot)})) D(\mathbf{x}_{(\cdot)}) \right]^{\{1..m\} \setminus \mathcal{I}^n}$$

where the shorthand $\{1..m\}$ is used to denote the set of integers $\{1, \dots, m\}$. Note that the value of the integrals is independent of the variable index, and thus

$$\rho_{\mathcal{S}}(n) = \sum_{m=n}^{\infty} e^{-N_X} \frac{1}{m!} \frac{m!}{n!(m-n)!} \langle 1_{\mathcal{S}}, D \rangle^n \langle 1 - 1_{\mathcal{S}}, D \rangle^{m-n}$$

from which Equation (14) follows. \square

Remark: Computation of Equation (14) requires only one integral computation; namely $\langle 1_{\mathcal{S}}, D \rangle$, which can be found either by summing the weights of Equation (3) or through Monte Carlo integration. Using the integral property of the PHD (Eq. 13), the integral

$$\langle 1 - 1_{\mathcal{S}}, D \rangle = N_X - \langle 1_{\mathcal{S}}, D \rangle$$

Furthermore, for $m \gg N_X$, the summand of Equation (14) is negligible, and the infinite sum can be safely truncated at an appropriately chosen $m = m_{\max}(N_X)$.

B. Independent Identically Distributed Cluster Distribution

The density of an i.i.d.c. RFS is

$$f(X) = |X|! \cdot \rho(|X|)[p]^X, \quad (16)$$

where $\rho(n)$ is the cardinality pmf and $p(\mathbf{x})$ is the single-object state pdf.

Proposition 2: Given an i.i.d.c.-distributed RFS with cardinality pmf $\rho(\cdot)$ and state density $p(\cdot)$, the cardinality of objects inside the FoV \mathcal{S} is distributed according to

$$\rho_{\mathcal{S}}(n) = \sum_{m=n}^{\infty} \rho(m) \binom{m}{n} \langle 1_{\mathcal{S}}, p \rangle^n \langle 1 - 1_{\mathcal{S}}, p \rangle^{m-n} \quad (17)$$

where $\binom{m}{n}$ is the binomial coefficient.

Proof: Substituting Equation (16) into Equation (11),

$$\rho_{\mathcal{S}}(n) = \sum_{m=n}^{\infty} \frac{1}{m!} m! \rho(m) \int_{\mathbb{X}^m} \sum_{X^n \subseteq X} [1_{\mathcal{S}}(\cdot)p(\cdot)]^{X^n} [(1 - 1_{\mathcal{S}}(\cdot))p(\cdot)]^{X \setminus X^n} d\mathbf{x}_1 \cdots d\mathbf{x}_m$$

The integral can be moved inside the products so that

$$\rho_{\mathcal{S}}(n) = \sum_{m=n}^{\infty} \rho(m) \sum_{\mathcal{I}^n \subseteq \{1..m\}} \left[\int 1_{\mathcal{S}}(\mathbf{x}_{(\cdot)}) p(\mathbf{x}_{(\cdot)}) d\mathbf{x}_{(\cdot)} \right]^{\mathcal{I}^n} \cdot \left[\int (1 - 1_{\mathcal{S}}(\mathbf{x}_{(\cdot)})) p(\mathbf{x}_{(\cdot)}) d\mathbf{x}_{(\cdot)} \right]^{\{1..m\} \setminus \mathcal{I}^n} \quad (18)$$

Equation (17) follows from Equation (18) by noting that there are $\binom{m}{n}$ unique unordered n -cardinality index subsets of $\{1, \dots, m\}$. \square

C. Multi-Bernoulli Distribution

The density of a MB distribution is [20, p. 102]

$$f(X) = \left[(1 - r^{(\cdot)}) \right]^{\{1..M\}} \sum_{1 \leq i_1 \neq \dots \neq i_n \leq M} \left[\frac{r^{i(\cdot)} p^{i(\cdot)}(\mathbf{x}_{(\cdot)})}{1 - r^{i(\cdot)}} \right]^{\{1..n\}} \quad (19)$$

where M is the number of MB components and maximum possible object cardinality, r^i is the probability that the i^{th} object exists, and $p^i(\mathbf{x})$ is the single-object state density of the i^{th} object if it exists.

Proposition 3: Given a MB density of the form of Equation (19), the cardinality of objects inside the FoV \mathcal{S} is distributed according to

$$\rho_{\mathcal{S}}(n) = \left[(1 - r^{(\cdot)}) \right]^{\{1..M\}} \cdot \sum_{\mathcal{I}_1 \uplus \mathcal{I}_2 \uplus \mathcal{I}_3} \delta_n(|\mathcal{I}_1|) \left[\frac{\langle 1_{\mathcal{S}}, r^{(\cdot)} p^{(\cdot)} \rangle}{1 - r^{(\cdot)}} \right]^{\mathcal{I}_1} \left[\frac{\langle 1 - 1_{\mathcal{S}}, r^{(\cdot)} p^{(\cdot)} \rangle}{1 - r^{(\cdot)}} \right]^{\mathcal{I}_2} \quad (20)$$

where the summation is taken over all mutually exclusive index partitions $\mathcal{I}_1 \uplus \mathcal{I}_2 \uplus \mathcal{I}_3 = \{1..M\}$.

Proof of Proposition 3 is given in Appendix A. Following the same procedure, similar results for the labeled multi-Bernoulli (LMB) [3] and multi-Bernoulli mixture (MBM) [21] RFS distributions may be obtained.

Direct computation of Equation (20) is only feasible for small M due to the sum over all permutations $\mathcal{I}_1 \uplus \mathcal{I}_2 \uplus \mathcal{I}_3$. For large M , a stochastic approximation may be used, as detailed in Algorithm 2 and summarized as follows. For each

MB component, the integral $\langle 1_{\mathcal{S}}, p^{(i)} \rangle$ is computed either by summing the weights of the partitioned GM or by Monte Carlo integration. Using the integral results, the probability of object i existing inside the FoV is found as

$$r_S^{(i)} = r^{(i)} \langle 1_{\mathcal{S}}, p^{(i)} \rangle$$

These probabilities are then sampled in N_s Monte Carlo trials to randomly generate $\bar{n}_{i,j}$ which is unity if the object i is in \mathcal{S} in the j^{th} trial and zero otherwise. The cardinality of each random trial is tallied, and the probability of n objects existing inside the FoV is given by the proportion of the number of trials with n objects with respect to the total number of trials.

Algorithm 2: Stochastic MB FOV Cardinality

```

for  $i = 1, \dots, M$  do
   $r_S^{(i)} \leftarrow r^{(i)} \langle 1_{\mathcal{S}}, p^{(i)} \rangle$ 
end for
for  $j = 1, \dots, N_s$  do
  for  $i = 1, \dots, M$  do
     $u \sim \text{Uniform}[0, 1]$ 
     $\bar{n}_{i,j} \leftarrow r_S^{(i)} \geq u$ 
  end for
   $\bar{n}_j \leftarrow \sum_{i=1}^M \bar{n}_{i,j}$ 
end for
 $\rho_S(n) \leftarrow \frac{1}{N_s} \sum_{j=1}^{N_s} \delta_n(\bar{n}_j)$ 

```

D. Generalized Labeled Multi-Bernoulli Distribution

The density of a GLMB distribution is given by [2]

$$\mathring{f}(\mathring{X}) = \Delta(\mathring{X}) \sum_{\xi \in \Xi} w^{(\xi)}(\mathcal{L}(\mathring{X})) [p^{(\xi)}]^{\mathring{X}}, \quad (21)$$

where each $\xi \in \Xi$ represents a history of measurement association maps, each $p^{(\xi)}(\cdot, \ell)$ is a probability density on \mathbb{X} , and each weight $w^{(\xi)}$ is non-negative with $\sum_{(I, \xi) \in \mathcal{F}(\mathbb{L}) \times \Xi} w^{(\xi)}(I) = 1$. The label of a labeled state \hat{x} is recovered by $\mathcal{L}(\hat{x})$, where $\mathcal{L} : \mathbb{X} \times \mathbb{L} \mapsto \mathbb{L}$ is the projection defined by $\mathcal{L}((\mathbf{x}, \ell)) \triangleq \ell$. Similarly, for LRFSSs, $\mathcal{L}(\mathring{X}) \triangleq \{\mathcal{L}(\hat{x}) : \hat{x} \in \mathring{X}\}$. The distinct label indicator $\Delta(\mathring{X}) = \delta_{\{|\mathring{X}|\}}(|\mathcal{L}(\mathring{X})|)$ ensures that only sets with distinct labels are considered.

Proposition 4: Given a GLMB density $\mathring{f}(\mathring{X})$ of the form of Equation (21), the cardinality of objects inside a bounded FoV \mathcal{S} is distributed according to

$$\rho_S(n) = \sum_{(\xi, \mathcal{I}_1 \uplus \mathcal{I}_2) \in \Xi \times \mathcal{F}(\mathbb{L})} w^{(\xi)}(I) \delta_n(|\mathcal{I}_1|) \langle 1_{\mathcal{S}}, p \rangle^{\mathcal{I}_1} \langle 1 - 1_{\mathcal{S}}, p \rangle^{\mathcal{I}_2} \quad (22)$$

Proof: Equation (10) can be rewritten to accommodate the labeled RFS as

$$\rho_S(n | \mathring{X}) = \sum_{\mathring{X}^n \subseteq \mathring{X}} [1_{\mathcal{S}}(\cdot)]^{\mathring{X}^n} [1 - 1_{\mathcal{S}}(\cdot)]^{\mathring{X} \setminus \mathring{X}^n} \quad (23)$$

If \mathring{X} is distributed according to the LRFS density $\mathring{f}(\mathring{X})$, the FoV cardinality distribution is obtained via the set integral

$$\rho_S(n) = \int \rho_S(n | \mathring{X}) \mathring{f}(\mathring{X}) \delta \mathring{X}$$

Expanding the integral,

$$\begin{aligned} \rho_S(n) &= \sum_{m=n}^{\infty} \frac{1}{m!} \sum_{(\ell_1, \dots, \ell_m) \in \mathbb{L}^m} \int_{\mathbb{X}^m} \rho_S(n | \{(\mathbf{x}_1, \ell_1), \dots, (\mathbf{x}_m, \ell_m)\}) \\ &\quad \cdot \mathring{f}(\{(\mathbf{x}_1, \ell_1), \dots, (\mathbf{x}_m, \ell_m)\}) d\mathbf{x}_1 \cdots d\mathbf{x}_m \end{aligned}$$

Defining $p^{(\xi, \ell)}(x) \triangleq p^{(\xi)}(x, \ell)$, substitution of Equations (21) and (23) yields

$$\begin{aligned} \rho_S(n) &= \sum_{m=n}^{\infty} \frac{1}{m!} m! \sum_{\{\ell_1, \dots, \ell_m\} \in \mathbb{L}^m} \sum_{\xi \in \Xi} w^{(\xi)}(\{\ell_1, \dots, \ell_m\}) \\ &\quad \sum_{I^n \subseteq \{\ell_1, \dots, \ell_m\}} \langle 1_{\mathcal{S}}, p^{(\xi, \cdot)} \rangle^{I^n} \langle 1 - 1_{\mathcal{S}}, p^{(\xi, \cdot)} \rangle^{\{\ell_1, \dots, \ell_m\} \setminus I^n} \\ &= \sum_{(\xi, I) \in \Xi \times \mathcal{F}(\mathbb{L})} w^{(\xi)}(I) \sum_{I^n \subseteq I} \langle 1_{\mathcal{S}}, p^{(\xi, \cdot)} \rangle^{I^n} \langle 1 - 1_{\mathcal{S}}, p^{(\xi, \cdot)} \rangle^{I \setminus I^n} \end{aligned}$$

from which Equation (22) follows. \square

Remark: Substitution of $n = 0$ in Equation 22 gives the GLMB void probability functional [6, Eq. 22], which, while less general, has theoretical significance and practical applications in sensor management.

V. SENSOR PLACEMENT EXAMPLE

The FoV statistics developed in this paper are demonstrated through a sensor placement optimization problem subject to multi-object uncertainty. For brevity, the workspace distribution is assumed MB-distributed. Numerical simulation is performed for the case of 100 MB components, with probabilities of existence randomly chosen between 0.35 and 1. Each MB component has a Gaussian density and randomly chosen mean and covariance. To visualize the workspace distribution, the PHD is shown in Figure 5.

The objective of the sensor control problem is to place the FoV, comprised of a square of 1×1 dimensions, in the ROI (Figure 5) such that the variance of object cardinality inside the FoV is maximized. This objective can be interpreted as placing the FoV in a region of the workspace where the object cardinality is most uncertain. A related objective which minimizes the variance of the *global* cardinality using cardinality-balanced multi-Bernoulli (CB-MeMBer) predictions was first proposed in [5]. For each candidate FoV placement, the FoV cardinality pmf is given by Equation (20) and is efficiently approximated using Algorithm 2. The variance of the resulting pmf is shown as a function of the FoV center location in Figure 6. The optimal FoV center location is found to be $(-0.8, -1.25)$.

A compelling result is that, by virtue of the bounded FoV geometry, spatial information is encoded in the FoV cardinality

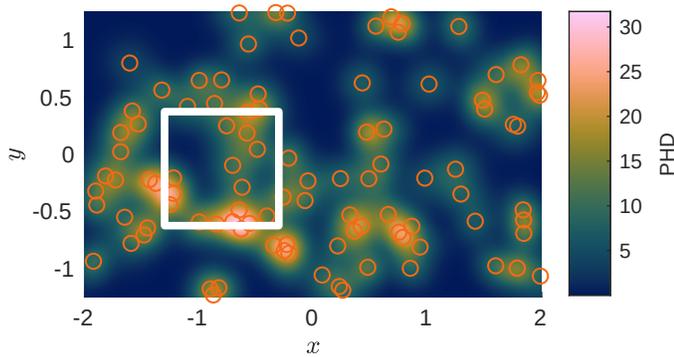


Fig. 5: PHD of MB workspace distribution with 100 potential objects, where object means are represented by orange circles and the bounds of the FoV that maximizes the FoV cardinality variance are shown in white.

pmf. It can be seen that the optimal FoV (Fig. 5) has boundary segments (lower half of left boundary and right half of lower boundary) that bisect clusters of MB components. These boundary segments divide the components' single-object densities such that significant mass appears inside and outside the FoV, increasing the overall FoV cardinality variance.

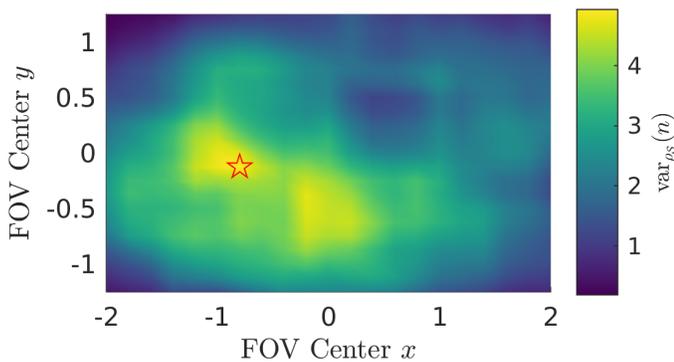


Fig. 6: FoV cardinality variance as a function of FoV center location, where the red star denotes the maximum variance point.

VI. CONCLUSIONS

This paper presents an approach for incorporating bounded field-of-view (FoV) geometry into state density updates and object cardinality predictions via finite set statistics (FISST). Negative information is processed in state density updates via a novel Gaussian splitting algorithm that recursively refines a Gaussian mixture approximation near the boundaries of the discrete FoV geometry. Using FISST, cardinality probability mass functions that describe the probability that a given number of targets exist inside the FoV are derived. The approach is presented for representative random finite set distributions and, thus, is applicable to a wide range of tracking, perception, and sensor planning problems.

REFERENCES

- [1] R. P. Mahler, *Statistical Multisource-Multitarget Information Fusion*. Artech House Boston, 2007.
- [2] B.-N. Vo, B.-T. Vo, and D. Phung, "Labeled random finite sets and the Bayes multi-target tracking filter," *IEEE Transactions on Signal Processing*, vol. 62, no. 24, pp. 6554–6567, 2014.
- [3] S. Reuter, B. T. Vo, B. N. Vo, and K. Dietmayer, "The labeled multi-Bernoulli filter," *IEEE Transactions on Signal Processing*, vol. 62, no. 12, pp. 3246–3260, 2014.
- [4] Á. F. García-Fernández, Y. Xia, K. Granström, L. Svensson, and J. L. Williams, "Gaussian implementation of the multi-Bernoulli mixture filter," in *2019 22nd International Conference on Information Fusion (FUSION)*, 2019.
- [5] H. G. Hoang and B. T. Vo, "Sensor management for multi-target tracking via multi-Bernoulli filtering," *Automatica*, vol. 50, no. 4, pp. 1135–1142, 2014. [Online]. Available: <http://dx.doi.org/10.1016/j.automatica.2014.02.007>
- [6] M. Beard, B. T. Vo, B. N. Vo, and S. Arulampalam, "Void probabilities and Cauchy-Schwarz divergence for generalized labeled multi-Bernoulli models," *IEEE Transactions on Signal Processing*, vol. 65, no. 19, pp. 5047–5061, 2017.
- [7] X. Wang, R. Hoseinnezhad, A. K. Gostar, T. Rathnayake, B. Xu, and A. Bab-hadiashar, "Multi-sensor control for multi-object Bayes filters," *Signal Processing*, vol. 142, pp. 260–270, 2018. [Online]. Available: <http://dx.doi.org/10.1016/j.sigpro.2017.07.031>
- [8] S. Ferrari, R. Fierro, B. Perteet, C. Cai, and K. Baumgartner, "A geometric optimization approach to detecting and intercepting dynamic targets using a mobile sensor network," *SIAM Journal on Control and Optimization*, vol. 48, no. 1, pp. 292–320, January 2009. [Online]. Available: <https://doi.org/10.1137/07067934X>
- [9] H. Wei and S. Ferrari, "A geometric transversals approach to sensor motion planning for tracking maneuvering targets," *IEEE Transactions on Automatic Control*, vol. 60, no. 10, pp. 2773–2778, 2015.
- [10] S. Gehly, B. A. Jones, and P. Axelrad, "Search-detect-track sensor allocation for geosynchronous space objects," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 54, no. 6, pp. 2788–2808, 2018.
- [11] A. Buonviri, M. York, K. A. LeGrand, and J. Meub, "Survey of challenges in labeled random finite set based distributed multi-sensor multi-object tracking," in *2019 IEEE Aerospace Conference*, 2019.
- [12] W. Koch, "On exploiting 'negative' sensor evidence for target tracking and sensor data fusion," *Information Fusion*, 2007.
- [13] T. L. Song, D. Musicki, and K. D. Sol, "Target tracking with target state dependent detection," *IEEE Transactions on Signal Processing*, vol. 59, no. 3, pp. 1063–1074, 2011.
- [14] B. Wei and B. Nener, "Distributed space debris tracking with consensus labeled random finite set filtering," *Sensors*, vol. 18, pp. 1–26, 2018.
- [15] K. A. C. Baumgartner, S. Ferrari, and A. V. Rao, "Optimal control of an underwater sensor network for cooperative target tracking," *IEEE Journal of Oceanic Engineering*, vol. 34, no. 4, pp. 678–697, 2009.
- [16] B.-N. Vo and W.-K. Ma, "The Gaussian mixture probability hypothesis density filter," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4091–4104, Nov 2006.
- [17] K. A. LeGrand and K. J. DeMars, "Relative multiple space object tracking using intensity filters," in *18th International Conference on Information Fusion (FUSION)*. IEEE, 2015, pp. 1253–1261.
- [18] M. F. Huber, T. Bailey, H. Durrant-Whyte, and U. D. Hanebeck, "On entropy approximation for Gaussian mixture random vectors," *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, pp. 181–188, 2008.
- [19] K. J. DeMars, R. H. Bishop, and M. K. Jah, "Entropy-based approach for uncertainty propagation of nonlinear dynamical systems," *Journal of Guidance, Control, and Dynamics*, vol. 36, no. 4, pp. 1047–1057, 2013. [Online]. Available: <http://arc.aiaa.org/doi/10.2514/1.58987>
- [20] R. P. Mahler, *Advances in Statistical Multisource-Multitarget Information Fusion*. Artech House, 2014.
- [21] J. L. Williams, "Marginal multi-Bernoulli filters: RFS derivation of MHT, JIPDA, and association-based MeMBer," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, no. 3, pp. 1664–1687, 2015.
- [22] 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.

APPENDIX A
PROOF OF PROPOSITION 3

Equation (19) can be rewritten as

$$f(X) = \left[\left(1 - r^{(\cdot)}\right) \right]^{\{1..M\}} \sum_{(\mathcal{I}_\sigma) \uplus \mathcal{I}_3} \left[\frac{r^{i(\cdot)} p^{i(\cdot)}(\mathbf{x}_{(\cdot)})}{1 - r^{i(\cdot)}} \right]^{\{1..n\}} \quad (24)$$

where (\mathcal{I}_σ) denotes the (ordered) sequence $(i_1, \dots, i_n) = (\alpha_{\sigma(1)}, \dots, \alpha_{\sigma(n)})$, where the n -tuple index set $\{\alpha_1, \dots, \alpha_n\} \uplus \mathcal{I}_3 = \{1, \dots, M\}$ and σ is a permutation of $\{1, \dots, n\}$.

Substituting Equation (24) into Equation (11),

$$\begin{aligned} \rho_S(n) &= \left[\left(1 - r^{(\cdot)}\right) \right]^{\{1..M\}} \\ &\cdot \sum_{m=n}^M \frac{1}{m!} \int_{\mathbb{X}^m} \sum_{(\mathcal{I}_\sigma) \uplus \mathcal{I}_3} \delta_m(|\mathcal{I}_\sigma|) \left[\frac{r^{i(\cdot)} p^{i(\cdot)}(\mathbf{x}_{(\cdot)})}{1 - r^{i(\cdot)}} \right]^{\{1..m\}} \\ &\quad \sum_{X^n \subseteq X} [1_S(\cdot)]^{X^n} [1 - 1_S(\cdot)]^{X \setminus X^n} d\mathbf{x}_1 \cdots d\mathbf{x}_m \end{aligned}$$

The last sum can be written in terms of label index sets $\mathcal{I}_1 \uplus \mathcal{I}_2 = \mathcal{I}_\sigma$ as

$$\begin{aligned} \rho_S(n) &= \left[\left(1 - r^{(\cdot)}\right) \right]^{\{1..M\}} \\ &\cdot \sum_{m=n}^M \frac{1}{m!} \int_{\mathbb{X}^m} \sum_{(\mathcal{I}_\sigma) \uplus \mathcal{I}_3} \delta_m(|\mathcal{I}_\sigma|) \left[\frac{r^{i(\cdot)} p^{i(\cdot)}(\mathbf{x}_{(\cdot)})}{1 - r^{i(\cdot)}} \right]^{\{1..m\}} \\ &\cdot \sum_{\mathcal{I}_1 \uplus \mathcal{I}_2 = \mathcal{I}_\sigma} \delta_n(|\mathcal{I}_1|) [1_S(\mathbf{x}_{(\cdot)})]^{\{j:i_j \in \mathcal{I}_1\}} [1 - 1_S(\mathbf{x}_{(\cdot)})]^{\{j:i_j \in \mathcal{I}_2\}} \\ &\quad d\mathbf{x}_1 \cdots d\mathbf{x}_m \end{aligned}$$

Distributing terms from the second summation,

$$\begin{aligned} \rho_S(n) &= \left[\left(1 - r^{(\cdot)}\right) \right]^{\{1..M\}} \\ &\cdot \sum_{m=n}^M \frac{1}{m!} \int_{\mathbb{X}^m} \sum_{(\mathcal{I}_\sigma) \uplus \mathcal{I}_3} \delta_m(|\mathcal{I}_\sigma|) \sum_{\mathcal{I}_1 \uplus \mathcal{I}_2 = \mathcal{I}_\sigma} \delta_n(|\mathcal{I}_1|) \\ &\cdot \left[\frac{1_S(\mathbf{x}_{(\cdot)}) r^{i(\cdot)} p^{i(\cdot)}(\mathbf{x}_{(\cdot)})}{1 - r^{i(\cdot)}} \right]^{\{j:i_j \in \mathcal{I}_1\}} \\ &\cdot \left[\frac{[1 - 1_S(\mathbf{x}_{(\cdot)})] r^{i(\cdot)} p^{i(\cdot)}(\mathbf{x}_{(\cdot)})}{1 - r^{i(\cdot)}} \right]^{\{j:i_j \in \mathcal{I}_2\}} d\mathbf{x}_1 \cdots d\mathbf{x}_m \end{aligned}$$

Because $\mathcal{I}_1 \cap \mathcal{I}_2 = \emptyset$, then $\{\mathbf{x}_j : i_j \in \mathcal{I}_1\} \cap \{\mathbf{x}_j : i_j \in \mathcal{I}_2\} = \emptyset$ and the integral on \mathbb{X}^m becomes a product of integrals on \mathbb{X} , such that

$$\begin{aligned} \rho_S(n) &= \left[\left(1 - r^{(\cdot)}\right) \right]^{\{1..M\}} \\ &\cdot \sum_{m=n}^M \frac{1}{m!} \sum_{(\mathcal{I}_\sigma) \uplus \mathcal{I}_3} \delta_m(|\mathcal{I}_\sigma|) \sum_{\mathcal{I}_1 \uplus \mathcal{I}_2 = \mathcal{I}_\sigma} \delta_n(|\mathcal{I}_1|) \\ &\cdot \left[\frac{\langle 1_S, r^{i(\cdot)} p^{i(\cdot)} \rangle}{1 - r^{i(\cdot)}} \right]^{\{j:i_j \in \mathcal{I}_1\}} \left[\frac{\langle 1 - 1_S, r^{i(\cdot)} p^{i(\cdot)} \rangle}{1 - r^{i(\cdot)}} \right]^{\{j:i_j \in \mathcal{I}_2\}} \end{aligned}$$

Now note that the result of the innermost sum does not depend the permutation order of (\mathcal{I}_σ) . Thus the property [22, Lemma 12] that for an arbitrary symmetric function h

$$\sum_{(i_1, \dots, i_m)} h(\{i_1, \dots, i_m\}) = m! \sum_{\{i_1, \dots, i_m\}} h(\{i_1, \dots, i_m\})$$

is applied, yielding

$$\begin{aligned} \rho_S(n) &= \left[\left(1 - r^{(\cdot)}\right) \right]^{\{1, \dots, M\}} \\ &\cdot \sum_{m=n}^M \sum_{\mathcal{I}_1 \uplus \mathcal{I}_2 \uplus \mathcal{I}_3} \delta_m(|\mathcal{I}_1 \uplus \mathcal{I}_2|) \delta_n(|\mathcal{I}_1|) \\ &\cdot \left[\frac{\langle 1_S, r^{(\cdot)} p^{(\cdot)} \rangle}{1 - r^{(\cdot)}} \right]^{\mathcal{I}_1} \left[\frac{\langle 1 - 1_S, r^{(\cdot)} p^{(\cdot)} \rangle}{1 - r^{(\cdot)}} \right]^{\mathcal{I}_2} \end{aligned}$$

The term $\delta_m(|\mathcal{I}_1 \uplus \mathcal{I}_2|)$ is non-zero only when the combined cardinality of \mathcal{I}_1 and \mathcal{I}_2 is equal to m —the index of the outermost sum. Thus, the outermost sum is absorbed by the second sum to give Equation (20). \square