

Random Set Methods: Estimation of Multiple Extended Objects

Karl Granström, *Member, IEEE*, Christian Lundquist,
Fredrik Gustafsson *Fellow, IEEE*, and Umut Orguner, *Member, IEEE*

Abstract—Random set based methods have provided a rigorous Bayesian framework and have been used extensively in the last decade for point object estimation. In this paper, we emphasize that the same methodology offers an equally powerful approach to estimation of so called extended objects, i.e., objects that result in multiple detections on the sensor side. Building upon the analogy between Bayesian state estimation of a single object and random finite set estimation for multiple objects, we give a tutorial on random set methods with an emphasis on multiple extended object estimation. The capabilities are illustrated on a simple yet insightful real life example with laser range data containing several occlusions.

Index Terms—Random finite set, Probability Hypothesis Density, Extended Objects.

I. INTRODUCTION

In 2007, former Microsoft CEO Bill Gates predicted that robotics would be the next hot research field, and indeed robotics research has seen a lot of activity and effort in the past decade. The world in which we live is becoming more and more automated, evidenced by the numerous robots that operate in air, on land, or in water.

One part of robotics research is multiple object estimation, defined as the processing of detections, obtained from multiple sources, in order to obtain and maintain estimates of the states of the objects of interest. In robotics, this applies to among other things both the multiple object tracking (MOT) problem, and the mapping part of the simultaneous localization and mapping (SLAM) problem. There is also the joint problem, called simultaneous localization, mapping and moving object tracking (SLAM-MOT). In SLAM-MOT the objective is to solve the SLAM problem while simultaneously keeping track of any moving objects. Note that in SLAM and SLAM-MOT the localization part is of equal importance. However, mobile robot localization falls outside the scope of this paper.

The word robot encompasses many different types of agents, with varying degrees of autonomy. In this paper we will use as a running example the scene depicted in Fig. 1, i.e., the robot

Karl Granström, Christian Lundquist and Fredrik Gustafsson are with the Department of Electrical Engineering, Division of Automatic Control, Linköping University, Linköping, SE-581 83, Sweden, e-mail: {karl, lundquist, fredrik}@isy.liu.se.

Umut Orguner is with the Department of Electrical and Electronics Engineering, Middle East Technical University, 06531, Ankara, Turkey, e-mail: umut@eee.metu.edu.tr.

The authors would like to thank the Linnaeus research environment CADICS and the frame project grant Extended Target Tracking (621-2010-4301), both funded by the Swedish Research Council, as well as the project Collaborative Unmanned Aircraft Systems (CUAS), funded by the Swedish Foundation for Strategic Research (SSF), for financial support.

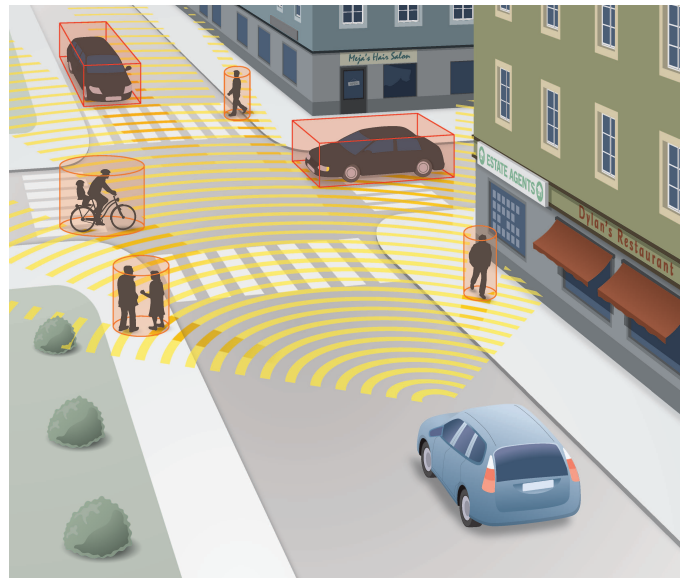


Fig. 1. Urban scene with multiple moving objects. The robot (blue vehicle) must keep track of all the moving objects (pedestrians, cars, bicycles) in order to operate safely. Illustration by Per Thorneus.

is an autonomous vehicle working in a urban environment. The robot's surroundings contain stationary objects, such as houses and trees, as well as moving objects, such as pedestrians, bicycles, and other vehicles. In MOT, the objects of interest are objects that are moving around in the vicinity of the robot. In Fig. 1 this corresponds to the pedestrians, the cars and the bicycle. In SLAM the objects of interest are stationary objects, often called landmarks, which the robot uses in the mapping and localization process. This corresponds to the buildings and the small bushes in Fig. 1.

To estimate the states of these stationary and moving objects means to estimate their locations, their speed and direction of motion, and possibly also which shape and size they have. For a robot working in an urban environment, having such estimates is necessary, e.g., for path-planning, collision avoidance, and classification of the objects into different types.

Both the MOT and the SLAM problem have received considerable research attention over the past decades, see e.g., [1] and [2] for MOT and SLAM respectively. With the advent of the random set methods [3], where the objects are not seen as individual objects, but as members of an object set, new statistical tools appeared that allowed MOT [4]–[6], SLAM [7] and SLAM-MOT [8] to be formulated as random set estimation problems. The random set methods not only enable a rigorous

TABLE I
OBJECT ESTIMATION SCENARIOS

Setup: A robot (blue rectangle) tracks moving objects (red rectangles) using a sensor with limited surveillance area (yellow circle segment).

		FIELD OF VIEW	
		Narrow: With a narrow FOW the robot can essentially only see straight ahead and only detect a single object.	Wide: If the FOW is wide the sensor can see to the sides as well, and it becomes possible to detect multiple objects.
RESOLUTION (rel. object size)	Low	Single point object: A single object that generates at most a single detection. This is sufficient for certain applications, e.g., distance keeping.	Multiple point objects: Multiple objects may be detected, and any object within the sensor's range may generate at most a single detection.
	High	Single extended object: A single object that may generate multiple detections. Possible to estimate the width of the leading vehicle, in addition to the distance.	Multiple extended objects: Multiple objects may be detected, and any object within the sensor's RNG may generate multiple detections.

generalization of the single object Bayes filter to multiple objects, but also provide the tools and approximations that are necessary to make implementations feasible.

In this paper we give a tutorial style overview of random set methods for multiple object estimation, by making an analogy between ideal single object estimation and multiple object estimation. The emphasis in the paper is on estimation of so called extended objects. Extended objects appeared in the MOT literature as a generalization of the point object concept. In extended object estimation the point object assumption (i.e., at most one sensor detection per object) is relaxed, and multiple sensor detections per object is allowed. For the sake of simplicity, we will limit the scope of the paper to only cover extended object tracking, however note that the presented methods equally well can be applied to SLAM with landmarks that are extended objects. A list of concepts that are central to multiple extended object estimation is given in Table II.

In Section II we will give background and motivation to the extended object estimation problem. The section is followed by a series of sections which aim at explaining the random set estimation problem. Beginning with the core and most ideal problem of estimating the state of a single object in Section III, the series is continued by explaining the assumptions and models necessary when extending the problem to multiple object estimation in Section IV, and finally ends by explaining the random set estimation problem in Section V. In these sections we would like to stress the similarity between the ideal single object estimation and the random set estimation. Computationally tractable filters are described in Section VI, some practical examples are shown in Section VII and the article is finalized with a summary in Section VIII.

II. BACKGROUND AND MOTIVATION

In order to estimate the states of these multiple objects, the robot must be equipped with one or several exteroceptive sensors that allow it to perceive the world, e.g., laser range sensors, radar sensors, or cameras. To keep things simple, in this paper we assume that a single sensor is used, and we assume that this sensor is of a type similar to laser range sensors and radar sensors. Note however that the material

TABLE II
MULTIPLE OBJECT ESTIMATION CONCEPTS

Kinematic state: The part of an object's state that contains information about the kinematics, e.g., position, velocity, acceleration, heading and turn-rate.
Extension state: The part of an object's state that contains information about the spatial extension, e.g., shape, size and orientation.
Point object: An object that causes at most one detection per sensor scan. A single detection per scan is sufficient to estimate the object's kinematic state. The name point object is derived from the fact that the detection is a single point in the detection space.
Extended object: An object that may cause more than one detection per sensor scan. Multiple detections per scan make it possible to estimate both the object's kinematic state and its extension state. The name extended object is derived from this possibility to estimate the extension state.
Random finite set: A set with a finite number of elements. Each element is a random variable, and the number of elements is also a random variable.
Probability Hypothesis Density (PHD): First order moment of a multiple object pdf. The PHD is to an RFS as the expected value is to a random variable.
PHD filter: Multiple object filter that propagates the object set's PHD in time.
CPHD filter: Multiple object filter that propagates the object set's PHD in time as well as the entire cardinality distribution. The CPHD filter's cardinality estimate has lower variance than the PHD filter's.
Partition: A division of a set into non-empty subsets, called cells. The union of all cells is equal to the original set. Partitioning of the set of detections is important in multiple extended object estimation using PHD or CPHD filters.

presented here easily can be generalized to other sensor types, e.g., cameras.

The sensor has a field of view (FOW) and a range (RNG), that together define the sensor's surveillance area. Both the FOW and RNG can be described further in terms of their respective resolution (RES). With time, technology development is moving towards an increase in all these properties, i.e., wider FOW, longer RNG, and higher RES.

Consider the FOW and its RES. Depending on whether the FOW is narrow or wide, and whether the resolution is low or high (relative to the size of the objects), four different kinds of object estimation problems arise, as shown in Table I. At this point it becomes necessary to distinguish between objects that may cause only a single detection each, and objects that

may cause multiple detections each. These types of objects are called point objects and extended objects, respectively, see Table II.

In the extended object case, depending on the type of sensor used the multiple detections will either be spread across the object's surface (e.g., when an airborne radar is used to track ground objects), or be spread along the edge of the object's shape (e.g., when laser is used to track vehicles and persons). In extended object estimation it is generally not of interest to estimate the locations of the points that cause the detections, because these points usually change fast with varying sensor-to-object geometry. Instead it is the principal extended object as a whole, i.e., its position, shape, size and orientation, that is of interest. Having estimates of the objects' extensions in addition to estimates of the kinematic states is useful for different robotics applications, e.g.,

- Path-planning and collision avoidance. When a robot moves it must plan its path such that it only traverses open area, because trivially it cannot go from one room to another by going through a wall. Further, in a crowded scene it must pass by both stationary and moving objects without hitting them. To succeed in both these tasks, it is necessary to know not only the location of the objects, but also their spatial extent.
- Classification of objects into different object types, e.g., car, bicycle or pedestrian in an urban environment. This is needed, e.g., for the robot to be able to interact with the objects in a correct manner.

Let us return to the sensor's properties. The FOW and the RNG of the sensor are what determine the sensor's surveillance area. Typically both the FOW and RNG are limited, and thus the surveillance area is limited. Trivially, it cannot be known *a priori* how many objects there are inside the sensors surveillance area, and during the course of operation objects might exit the surveillance area and new objects might enter it. Objects that are inside the surveillance area may also be invisible to the sensor due to occlusion from other objects, and false detections of non existing objects may complicate things further. Thus, the number of visible objects is both time-varying and unknown.

An example of the multiple extended object scenario is given in Fig. 2. The detections display a large degree of structure, especially the *L*-shaped cluster caused by a car, and it is therefore suitable to estimate the shape and size of the objects, in addition to their positions. The sensor's FOW is 180 degrees with a resolution of 0.5 degrees and the maximum RNG is 13 meters, which gives a semi-circular surveillance area. Existing objects would disappear if they move outside the semi-circle, and new objects would appear along the edge of the semi-circle.

In the next section we will overview Bayesian estimation of a single and ideal object's state, using a sequence of sensor detections. The single object case will then be generalized to multiple objects, and we will show how random set methods can be used to derive a multiple object analog to single state Bayesian estimation.

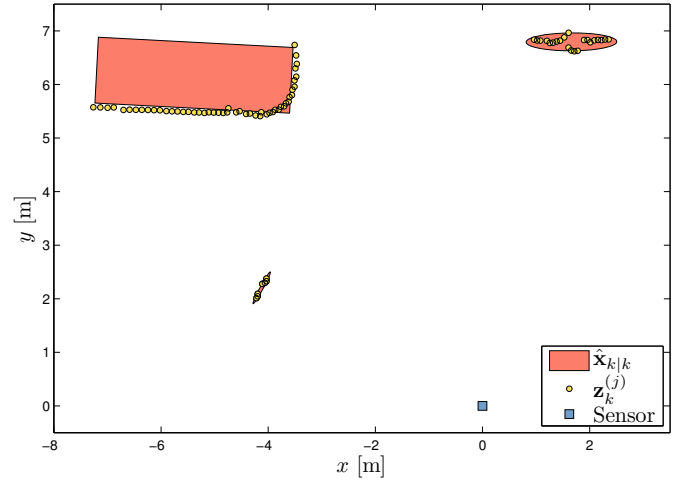


Fig. 2. Laser range sensor detections (yellow dots) caused by a car, a bicycle and a pedestrian. The sensor is located in the origin (blue square), object estimates are shown in red.

III. BAYESIAN STATE ESTIMATION

To be able to estimate the state of multiple objects, one must be able to estimate the state of a single object. In this estimation problem, the single state, denoted \mathbf{x}_k , generates a single detection, denoted \mathbf{z}_k in each discrete time step k . A Bayesian state estimation algorithm is made up of two steps: the time update and the measurement update.

A. Time update

The time update consists of predicting the motion that the object performs in between detections. In the general case the motion is not known, and therefore only simple assumptions can be made about the type of motion that the object is performing. To facilitate this a motion model of the form $\mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{v}_k)$ can be used, where $f(\cdot)$ is typically a non-linear function. Random process noise \mathbf{v}_k is included to handle uncertainties and imperfect modeling. Note that even for the SLAM problem, where the landmark objects can be assumed stationary, the time evolution of the objects' states must be modeled. In this case it is typically sufficient to model the state as being (approximately) constant over time.

B. Measurement update

The measurement update consists of using the sensor detections to update the object estimate, which requires a measurement model $\mathbf{z}_k = h(\mathbf{x}_k, \mathbf{e}_k)$, where $h(\cdot)$ is typically a non-linear algebraic function. A fundamental characteristic here is that each detection is corrupted by noise \mathbf{e}_k , i.e., the state of the object cannot be found by simply taking the inverse of the measurement model $\mathbf{x}_k = h^{-1}(\mathbf{z}_k)$.

C. Bayesian recursion

In each time step the sensor gives a detection \mathbf{z}_k . Let \mathbf{z}^k be all detections from time 1 to time k , i.e., $\mathbf{z}^k = \{\mathbf{z}_i\}_{i=1}^k$. The objective of single object estimation is to use \mathbf{z}^k to estimate the object state \mathbf{x}_k . This is illustrated in Fig. 3a.

$$\begin{array}{ccccccc}
\cdots & p(\mathbf{x}_k | \mathbf{z}^k) & \rightarrow & \boxed{\text{Time update}} & \rightarrow & p(\mathbf{x}_{k+1} | \mathbf{z}^k) & \rightarrow & \boxed{\text{Measurement update}} & \rightarrow & p(\mathbf{x}_{k+1} | \mathbf{z}^{k+1}) & \cdots \\
& & & \uparrow & & & & \uparrow & & & \\
& & & \text{Transition density} & & & & \text{Likelihood} & & & \\
& & & p(\mathbf{x}_{k+1} | \mathbf{x}_k) & & & & p(\mathbf{z}_{k+1} | \mathbf{x}_{k+1}) & & & \\
& & & \uparrow & & & & \uparrow & & & \\
& & & \text{Motion model} & & & & \text{Measurement model} & & & \\
& & & \mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{v}_k) & & & & \mathbf{z}_{k+1} = h(\mathbf{x}_{k+1}, \mathbf{e}_{k+1}) & & & \\
\end{array} \tag{1}$$

Because of the uncertainties involved (process and measurement noise and unknown initial state) the knowledge of the object's state \mathbf{x} is often described using probability distributions $p_{\mathbf{x}}(\mathbf{x}_k | \mathbf{z}^k)$. With the help of a motion model $f(\cdot)$ and a measurement model $h(\cdot)$ the time evolution of the distribution of the state \mathbf{x} can be described in a recursive Bayesian framework, as outlined in (1).

Propagation of the full distribution can sometimes be unnecessary complex from a computational perspective. A simpler approach is to only propagate the first order moment of the single object state, called the expected value,

$$\cdots \quad \mathbb{E}[\mathbf{x}_{k|k}] \xrightarrow{\text{t.u.}} \mathbb{E}[\mathbf{x}_{k+1|k}] \xrightarrow{\text{m.u.}} \mathbb{E}[\mathbf{x}_{k+1|k+1}] \quad \cdots \tag{2}$$

where m.u. and t.u. are abbreviations for measurement update and time update. Propagating the expected value (2) is, compared to a full Bayes recursion (1), both simpler and computationally cheaper.

IV. MULTIPLE OBJECT STATE ESTIMATION

Multiple object estimation is a joint estimation problem that consists of estimating the number of objects, and estimating the state of each object. Just as in single object estimation, in multiple object estimation all of the detections are corrupted by noise. However, there are additional characteristics that further complicate the estimation problem.

For each object the probability of detection is less than one. In practice, this means that we cannot know for certain whether or not an object caused any detection. There may also be false alarms, also called clutter detections. Furthermore, the detection origin is unknown: we do not know which detections are clutter and which are caused by actual objects; and we do not know which object caused which detection. A final characteristic of multiple object estimation is that we do not know how many objects there are inside the sensor's surveillance area.

Let $N_{x,k}$ denote the unknown number of objects present at time k , and let $\mathbf{x}_k^{(i)}$ denote the state of object i at time k . At time k the set \mathbf{X}_k of all present objects is given by

$$\mathbf{X}_k = \left\{ \mathbf{x}_k^{(i)} \right\}_{i=1}^{N_{x,k}} \tag{3}$$

Each sensor scan gives $N_{z,k}$ detections $\mathbf{z}_k^{(j)}$. Let the set of detections at time k be denoted

$$\mathbf{Z}_k = \left\{ \mathbf{z}_k^{(j)} \right\}_{j=1}^{N_{z,k}} \tag{4}$$

and let \mathbf{Z}^k be all sets of measurements from time 1 to time k , i.e., $\mathbf{Z}^k = \{\mathbf{Z}_i\}_{i=1}^k$. The objective of multiple object tracking

is to estimate \mathbf{X}_k given \mathbf{Z}^k , i.e., to determine how many objects there are, and for the objects that are present, to estimate the object states $\mathbf{x}_k^{(i)}$. This is illustrated in Fig. 3b.

Just as in single object estimation, each individual estimate needs to be both time updated and measurement updated. In addition to this, classical approaches to MOT have included data handling and data association. In the next two subsections we will briefly review these two problems.

A. Data handling

Data handling means to handle the fact that the number of objects in the sensor's surveillance area changes over time. This includes new targets appearing and old targets disappearing, in MOT literature also called object birth and object death. Approaches to handle birth and death include M/N logic and the Score based approach, see e.g., [1].

B. Data association

Using multiple detections to estimate the states of multiple objects requires an approach to the data association, or correspondence, problem. Data association means to associate each detection to one of the detection generating sources, i.e., either to an object or a clutter source. Correct data association is very important, because an incorrect association solution can result in disastrous estimation performance. In each time step, each detection is assumed to be either clutter, or generated by an object. For the detections that are generated by objects, a decision has to be made as to which measurements belong to already existing objects, and which measurements belong to newly appeared objects. Note that if it is known that there can be at most one object, cf. the single object cases in Table I, the data association problem becomes somewhat easier.

In a multiple point object scenario at most one detection can be caused by each source, since a point object by definition may cause at most one detection. For multiple point object estimation, solutions to the data association problem in MOT literature include global nearest neighbor (GNN), joint probabilistic data association (JPDA), and multiple hypothesis tracking (MHT). A comprehensive overview of these methods is given in [1]. In SLAM literature the joint compatibility branch and bound (JCBB) algorithm can be found [9].

In contrast to point objects, an extended object by definition may cause more than one detection, and thus in a multiple extended target scenario at most one *cell* of detections can be caused by each source. In this context a cell of detections is a non-empty subset of the full set of detections. A *partition* of

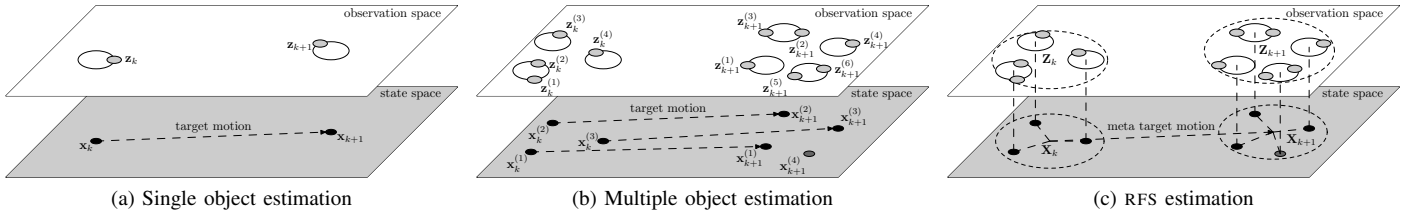


Fig. 3. The objects' states \mathbf{x} belong to a state space, and the detections \mathbf{z} are generated in an observation space. (a) There is a one-to-one correspondence between the detection and the state. (b) Generalization of the single object case where the detection-to-state correspondence is unknown. (c) The multiple objects are treated as an RFS whose set members all belong to the same state space, and the detections are treated as an RFS whose set members all belong to the same observation space. There is a one-to-one correspondence between the object set \mathbf{X} and the detection set \mathbf{Z} .

the set of detections is a set of cells, such that the union of the cells is equal to the full set of detections. In the measurement update each partition must have a likelihood that corresponds to how probable the partition is.

Naturally there are multiple ways to form such partitions. For Bayes optimality it is necessary to consider all possible partitions, but from a practical perspective considering all possible partitions is computationally intractable. Consider the laser detections in Fig. 2: there are almost 10^{87} different ways to partition the 80 detections. However, an absolute majority of the possible partitions can be discarded on the basis of being highly unlikely. One such example is a partition that deems the small cluster around $(x, y) = (-4, 2)$ and the elongated cluster around $(x, y) = (2, 7)$ to be from the same target. The intuition here is that detections are from the same object if they are clustered together, where the clusters may or may not have a spatial structure.

Central here is that only the most likely partitions are considered, while remaining partitions are discarded on the basis of being too improbable. Note that care must also be taken in ambiguous cases where a cluster of detections may have been caused by a single object, or by multiple smaller objects. One such example can be found in Fig. 2, where it is not immediately obvious if the elongated detections around $(x, y) = (2, 7)$ were caused by one object, e.g., a bike, or by two objects, e.g., two pedestrians. This ambiguity is handled by letting the estimation algorithm consider both partitions, with appropriate likelihoods for each partition. Algorithms for limiting the number of partitions to the most likely ones, without sacrificing estimation performance, are given in [10], [11].

An alternative way to approach multiple object estimation is to use random set methods [3], which relaxes the need for solving the data association problem, and the need for data handling. This is the topic of the next section.

V. MULTIPLE OBJECT BAYESIAN STATE ESTIMATION

In this section we will overview multiple object estimation under random set models. A random finite set is defined as follows [3].

Definition 1 (Random finite set (RFS)): A random variable Υ that draws its instantiations $\Upsilon = \mathbf{Y}$ from the hyperspace \mathcal{Y} of all finite subsets \mathbf{Y} (the null set \emptyset included) of some underlying space \mathcal{Y}_0 . \square

While the definition may seem daunting at first, the reality is that the RFS concept is quite intuitive. In MOT and SLAM, the underlying space \mathcal{Y}_0 is typically a Euclidean vector space, i.e., $\mathcal{Y}_0 = \mathbb{R}^{n_y}$. In SLAM, this would mean that the landmarks' states are vectors that typically define the landmarks' positions, and possibly also their orientations. In MOT, the moving objects' states are vectors that define the objects' positions, and their kinematic properties such as velocity, acceleration, heading, and turn-rate. In case the objects are extended, the state vectors also include any parameters that govern the shape, size and orientation of the object's extension.

If $\mathcal{Y}_0 = \mathbb{R}^{n_y}$ then the instantiations, or realizations, are sets $\mathbf{Y}_k = \left\{ \mathbf{y}_k^{(j)} \right\}_{j=1}^{N_{y,k}}$, where the set cardinality $N_{y,k}$ is a discrete random variable, and each set member $\mathbf{y}_k^{(j)} \in \mathcal{Y}_0$ is a random vector. Note here that \mathbf{Y}_k is without order, i.e., $\left\{ \mathbf{y}_k^{(1)}, \mathbf{y}_k^{(2)} \right\} = \left\{ \mathbf{y}_k^{(2)}, \mathbf{y}_k^{(1)} \right\}$. This observation may seem trivial, however it is important to the multiple object filters presented below because it disposes of the need for book-keeping of the order of the object estimates.

Considering the characteristics of multiple object estimation, the number of objects/detections, and each individual object/detection, can all be modeled as random variables. Thus, the RFS concept is suitable for modeling multiple object estimation, both for MOT and for mapping in SLAM. In practice this means that we see the set of objects (3) and the set of detections (4) as RFS:s, and as stated in the previous section the purpose is to estimate \mathbf{X}_k given \mathbf{Z}^k . To this end, both a time update and a measurement update is needed.

A. Time update

The RFS motion model captures the following:

- 1) The time evolution of objects that are already located in the surveillance area, modeled by the RFS $\mathbb{F}(\mathbf{X}_k)$. This includes objects that move such that they are still located in the surveillance area in the next time step, and it also includes the disappearance of existing objects.

The former is typically achieved by assuming that all objects move independently of each other, see e.g., [1], and with the same type of motion model $f(\cdot)$. However, note that the motion parameters are estimated individually for the objects, i.e., one velocity is estimated for each object. Note also that the same type of motion models can generally be used regardless of whether a moving point object or a moving extended object is estimated.

The disappearance of objects is captured by a state dependent probability of survival $p_S(\mathbf{x})$ which assigns a non-zero probability to the event that an object with state \mathbf{x} will not be present in the next time step.

- 2) The appearance of new objects in the surveillance area. This can either be an object that spawns from an already existing object, modeled by the RFS $\mathbb{S}(\mathbf{X}_k)$, or it could be an entirely new object, modeled by the RFS $\mathbf{\Gamma}_k$.

It should be noted that under the assumption of independent motion it is not possible to model interactive motion between multiple objects.

B. Measurement update

The RFS measurement model captures the following:

- 1) How many detections an object will generate, and how each detection relates to the object's state, modeled by the RFS $\mathbb{H}(\mathbf{X}_k)$. A typical choice is to model the number of detections as Poisson distributed, with a state dependent rate $\gamma(\mathbf{x}_k)$, see e.g., [12]. Given the number of detections, how the set of detections relates to the object's state is described with a measurement model $h(\cdot)$. Missed detection and occlusion by other objects is captured by a state dependent probability of detection $p_D(\mathbf{x}_k)$ which assigns a non-zero probability to the event that an object with state \mathbf{x}_k causes no detection at all.
- 2) How many false detections there are, and how these are distributed, modeled by the RFS \mathbf{K}_k .

This is typically modeled under the assumption that the objects generate detections independently of each other, and that multiple detections from the same object are generated independently of each other. For the extended object case, different alternatives for the measurement model $h(\cdot)$ exist, ranging from the simpler to the more advanced.

- The simplest case is to assume that all objects' extensions are equal and constant over time. In this case the measurement uncertainty represents both the effect of the measurement noise, and the object's extension. The measurement model can be similar to the models used for point objects, with the addition of multiple detections per object, see e.g., [10]. Conditioned on the number of detections, the detection set likelihood is the product of the individual detection likelihoods.
- A straightforward way to model the object extension is to assume a specific geometric shape, e.g., a rectangle or an ellipse. A popular model for elliptically shaped extended objects is the random matrix model [13]. This model decomposes the object state \mathbf{x}_k into a random vector x_k that represents the kinematic state and a random matrix X_k that represents the elliptical extension, abbreviated as $\mathbf{x}_k = (x_k, X_k)$. Assuming a linear Gaussian measurement model $\mathbf{z}_k = h(\mathbf{x}_k, \mathbf{e}_k) = Hx_k + \mathbf{e}_k$, $\mathbf{e}_k \sim \mathcal{N}(\mathbf{0}, X_k)$, the measurement update is linear.
- A more general model can be obtained by giving the shape a parametrization that allows for, under certain conditions, an arbitrary shape, see e.g., [14]. Another general alternative is to assume that there are reflection points across the object's surface, see e.g., [15] where

the reflection points act as point objects and generate the detections. The shape of the object is then given by the spatial structure of these reflection points. In this case the reflection points are used as an intermediary in the measurement model to describe the relation between the detections and the object's state.

C. Bayesian recursion

In Section III a single object Bayes recursion was introduced. With the help of the random finite set concept introduced above, the single object Bayes recursion can be generalized to the multiple object case. This requires us to model both the set of detections (4) and the set of objects (3) as RFS:s. This is illustrated in Fig. 3c, where the objects are members of the object set \mathbf{X}_k , and the detections are members of the detection set \mathbf{Z}_k . Comparing Fig. 3a and Fig. 3c, we see that there is a one-to-one correspondence in both cases: between the state \mathbf{x}_k and the detection \mathbf{z}_k , and between the object set \mathbf{X}_k and the detection set \mathbf{Z}_k .

Modeling the objects and detections as RFS:s is fundamental to allowing us to cast multiple object estimation, in the presence of false detections and with uncertain detection origin, in a Bayesian framework. In (5) a random set equivalent to the single object Bayes recursion (1) is given. Similarly to the single object case, both motion and measurement models are needed. If we compare single object estimation, (1) and Fig. 3a, to the RFS formulation of multiple object estimation, (5) and Fig. 3c, we see that there are conceptual similarities between the two. However, while the single object Bayes filter is computationally tractable, the multiple object generalization is typically not [3]. To alleviate the large computational demands approximations are needed, and in the next section we will review two approximations to the full multiple object Bayes recursion.

VI. COMPUTATIONALLY TRACTABLE RFS FILTERS

In Section III-C propagating the expected value of the object's state (2) was proposed as a computationally cheap alternative to the full Bayesian recursion (1). In this section we will overview the multiple object equivalents to (2), i.e., we will show how computational tractability of (5) can be achieved. The corresponding filters are abbreviated PHD and CPHD filters.¹

A. The PHD filter

The first order moment of a single object pdf is the expected value, and a simplified version of the single object Bayes recursion (1) is to propagate only the first order moment (2).

The multiple object equivalent to the expected value, i.e., the first order moment of a multiple object pdf, is called the *probability hypothesis density* (PHD). The PHD is an intensity function $D_{k|k}(\mathbf{x})$ that is defined on single target states $\mathbf{x} \in \mathcal{X}_0$ [3], and whose local maxima correspond to object locations. It

¹MATLAB implementations of extended object PHD and CPHD filters are freely available for research purposes. They can be acquired by contacting the authors.

$$\begin{array}{ccccccc}
\cdots & p(\mathbf{X}_k | \mathbf{Z}^k) & \rightarrow & \boxed{\text{Time update}} & \rightarrow & p(\mathbf{X}_{k+1} | \mathbf{Z}^k) & \rightarrow & \boxed{\text{Measurement update}} & \rightarrow & p(\mathbf{X}_{k+1} | \mathbf{Z}^{k+1}) & \cdots \\
& & & \uparrow & & & & \uparrow & & & \\
& & & \text{Set transition density} & & & & \text{Set likelihood} & & & \\
& & & p(\mathbf{X}_{k+1} | \mathbf{X}_k) & & & & p(\mathbf{Z}_{k+1} | \mathbf{X}_{k+1}) & & & (5) \\
& & & \uparrow & & & & \uparrow & & & \\
& & & \text{Set motion model} & & & & \text{Set measurement model} & & & \\
& & & \mathbf{X}_{k+1} = \mathbb{F}(\mathbf{X}_k) \cup \mathbb{S}(\mathbf{X}_k) \cup \Gamma_k & & & & \mathbf{Z}_{k+1} = \mathbb{H}(\mathbf{X}_{k+1}) \cup \mathbf{K}_{k+1} & & &
\end{array}$$

is uniquely determined by the property that, given any region S in single target state space \mathcal{X}_0 (i.e., $S \subseteq \mathcal{X}_0$) the integral

$$N_{k|k}^S = \int_S D_{k|k}(\mathbf{x}) d\mathbf{x} \quad (6)$$

is the expected number of targets in S [3]. Especially, if $S = \mathcal{X}_0$ is the entire state space then $N_{k|k}^{\mathcal{X}_0}$ is the expected total number of targets [3]. The practical implications of the PHD can be summarized briefly as follows.

- 1) Given an estimated PHD, a cardinality estimate $\hat{N}_{k|k}^{\mathcal{X}_0}$ is obtained by taking the integral of the PHD over the entire state space.
- 2) Given a cardinality estimate, individual object estimates can be acquired by taking the $\hat{N}_{k|k}^{\mathcal{X}_0}$ largest local maxima of the intensity function.

The so called PHD filter propagates the object set's PHD in time [3],

$$\cdots D_{k|k}(\mathbf{x}) \xrightarrow{\text{t.u.}} D_{k+1|k}(\mathbf{x}) \xrightarrow{\text{m.u.}} D_{k+1|k+1}(\mathbf{x}) \cdots \quad (7)$$

and represents a computationally tractable approximation to full multiple object Bayes filter (5). The PHD filter can be interpreted as an RFS equivalent to propagating the expected value of a state vector (2). In principle it is possible to derive a second order multiple object filter, however such a filter is considered unlikely to be computationally tractable [3].

Practical implementations of the PHD filter require approximations of the PHD intensity, either using Monte Carlo samples (a.k.a. particle filters), or using a distribution mixture

$$D_{k|k}(\mathbf{x}) = \sum_{j=1}^{J_{k|k}} w_{k|k}^{(j)} p(\mathbf{x}; \zeta_{k|k}^{(j)}), \quad (8)$$

where $\zeta_{k|k}^{(j)}$ denotes the parameters for the j th component of the distribution mixture. In MOT and SLAM applications, the Gaussian distribution prevails as a popular choice for modeling the object state distribution, as well as the transition density and likelihood. This is the case also for PHD filters.

For the multiple point target case, a Gaussian mixture (GM) PHD filter is given in [16]. In this filter, the object state is a point that can be seen as representing the object's center of mass. For the case of multiple extended objects, GM-PHD filters are given in [10], [17]. In the former, the object state is a point that represents the object's center of mass [10]; in the latter the object state also contains parameters for the spatial extension of the object [17]. Under the random matrix extended object model [13] a PHD filter for multiple extended objects is given in [11].

A known drawback of the PHD filter is that its cardinality estimate has high variance, a problem that is manifested e.g., when there are missed detections. The result is typically that the cardinality is underestimated. The cause of the high variance is the approximation of the full cardinality distribution with a Poisson distribution, which has a single parameter corresponding to the mean. Because the Poisson distribution has equal mean and variance, when the true cardinality is high, the corresponding estimate is high and thus also has a high variance.

To improve upon the PHD filter's cardinality estimate, the *cardinalized probability hypothesis density* (CPHD) filter was introduced [3]. The CPHD filter is the topic of the next subsection.

B. The CPHD filter

In addition to propagating the PHD in time (like the PHD filter does), the CPHD filter also propagates the full cardinality distribution $P_{k|k}(n)$,

$$\cdots \begin{Bmatrix} D_{k|k}(\mathbf{x}) \\ P_{k|k}(n) \end{Bmatrix} \xrightarrow{\text{t.u.}} \begin{Bmatrix} D_{k+1|k}(\mathbf{x}) \\ P_{k+1|k}(n) \end{Bmatrix} \xrightarrow{\text{m.u.}} \begin{Bmatrix} D_{k+1|k+1}(\mathbf{x}) \\ P_{k+1|k+1}(n) \end{Bmatrix} \cdots \quad (9)$$

Just as in the PHD filter, in the CPHD filter the PHD intensity must be approximated, either using Monte Carlo samples or distribution mixtures. In addition, the cardinality distribution is typically propagated as a truncated version of the full distribution. In practice this means that the probability $P_{k|k}(n)$ of n targets is only computed for $n \in [0, N_{max}]$. Here N_{max} must be chosen such that it is larger than the largest number of objects that it is believed will appear in the surveillance area at any one time.

Using a GM approximation for the PHD intensity, a CPHD filter for point objects is given in [18]. An extended object CPHD filter implementation is presented in [19].

VII. EXPERIMENTAL RESULTS

In this section we will illustrate object estimation with an experiment in which pedestrians are tracked using a laser range sensor. Modern laser range sensors have a wide FOW with high resolution, the RNG varies from being on the order of 10's of meter and upwards, with high resolution. An example of laser range data is given in Fig. 2. The laser range sensor falls into the the multiple extended object category, see Table I. Pedestrian tracking using a laser sensor is a suitable introductory example of multiple extended object estimation due to its relative simplicity.

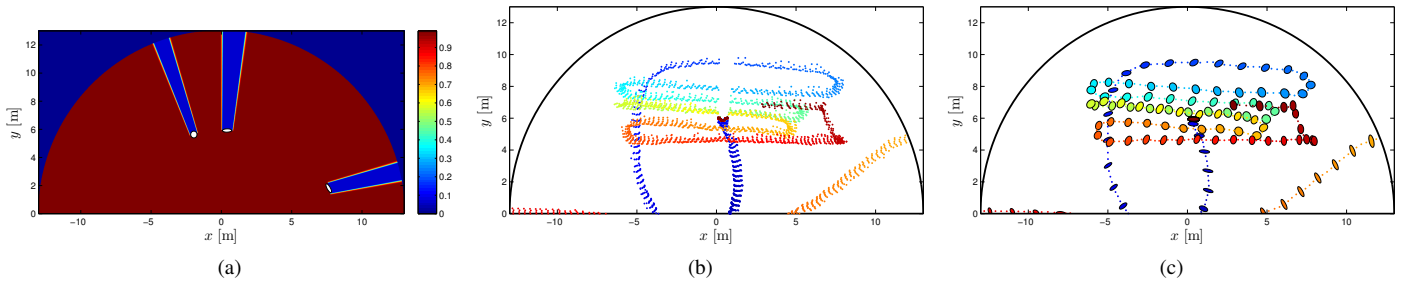


Fig. 4. Results from experiment. (a) Variable probability of detection. Behind the three extended target estimates (white ellipses) the probability of detection is lower. Image created using the method from [11]. (b) Laser range data from four pedestrians. The edge of the surveillance area is shown as the black semi-circle. The detections are shown as dots, where the colors denote different time steps. (c) Results for the data in (b). For clarity the extension estimate is only plotted every fourth time step, the small dots show only the kinematic positions.

- As a model for the shape of a pedestrian an ellipse is typically sufficient: in this particular example we will use the random matrix model [13]. In Fig. 2 this shape model has been applied to the pedestrian and bicycle detections. However, note that an ellipsoidal model is not always suitable when laser range sensors are used, e.g., not for detections from the car. In Fig. 2 the car detections have been applied to a rectangular extended object model [17].
- As a model for the motion of a pedestrian, a simple constant velocity or constant acceleration model, see e.g., [20], is typically sufficient. When humans walk around they may “turn on a dime,” and it is therefore necessary to have a sufficiently large process noise covariance.

Like many other sensors the laser range sensor is subject to the occlusion problem, i.e., the sensor cannot see an object if it is located behind another object. In an environment with several pedestrians, trivially some pedestrians will walk behind others. As they do so they will be either fully occluded (not detectable at all) or partially occluded (only parts of the object is detectable). In order to correctly estimate the number of pedestrians, and their respective locations, the estimation filter must be able to handle the occlusion problem. In the experiment presented here, occlusion is handled by modeling the probability of detection as non-homogeneous in the surveillance area. In brief, the intuition is that an object that is occluded cannot cause any detections, and thus the probability of detection for that particular object is zero. Because the true object locations are unknown, the object estimates are used instead to compute an approximate probability of detection. The idea is illustrated in Fig. 4a.

The experimental data, also used in [10], [11], is shown in Fig. 4b. The sensor was mounted at waist height, and four pedestrians walked around in the surveillance area, however at most three pedestrians were present at any given time. One of the pedestrians remained still around the position $(x, y) \approx (0.4\text{m}, 6\text{m})$ during most of the experiment, see Fig. 4b. Another pedestrian then moved behind and in front of the still pedestrian, causing multiple instances of both full and partial occlusion. Thanks to the non-homogeneous probability of detection, the filter can maintain estimates of the objects even when they move through occluded parts of the surveillance area.

The pedestrian estimates are shown in Fig. 4c. For this data there is unfortunately no ground truth available that the estimates can be compared to. However, visual inspection of the estimates in Fig. 4c, and comparison to the data in Fig. 4b, shows that the results are good. The pedestrian that repeatedly moves through an occluded area can be tracked also during the occlusions. The estimated extension ellipses are a good approximation of the size of a person measured at waist height.

Comparisons of the estimated and true number of targets can be found in [10], [11]. The results show that the estimated number of targets is more robust against errors when the size and shape of the pedestrians is estimated, compared to when the shape and size is assumed constant.

VIII. SUMMARY

Multiple object estimation is a well established research area which recently has met a considerable interest in safety and security applications, as for example the urban situation awareness problem illustrated in Fig. 1. The first generation of vehicles with situation awareness had a low resolution radar, the case in the upper left corner of Table I, which is sufficient for adaptive cruise controllers. The second generation have higher resolution, corresponding to the upper right corner in Table I, which is used for collision avoidance systems. To get fully automated vehicles, as have been demonstrated in, e.g., DARPA’s grand challenges, the laser range sensor is instrumental. Its high resolution and field of view is illustrated in the lower right corner of Table I. The main difference is that the vehicle now gets plenty of detections from each object. Fig. 2 shows a snapshot of real laser range data, where suitable extension models have been fitted to the data caused by the car, the bike and the pedestrian, respectively. To find these shapes, and to track them over time, is the goal in multiple extended object estimation. This research field has evolved over time along the following main thrusts:

- 1) Bayesian state estimation for single point objects (Fig. 3a). This includes Kalman filter approaches, as well as filter banks and particle filters, yielding a Gaussian mixture or particle representation of the state probability density function. The Bayesian filter has a very simple structure, illustrated in (1).
- 2) Bayesian state estimation for single extended objects. Using parametric extension models such as ellipses or

rectangles, the extension parameters can be augmented to the state vector, and the problem is recast to 1 above.

- 3) Bayesian state estimation for multiple point object tracking (Fig. 3b). This gives a combinatorial explosion in complexity due to the association of detections to objects and clutter as outliers. For that reason, suboptimal track handling algorithms have been developed to again recast the problem to 1 above.
- 4) Bayesian estimation of the PHD for multiple extended objects. The PHD filter is a mathematically beautiful approach, originally developed for multiple point object estimation, and later generalized to extended objects. The filter in (5) has the same simple structure as in (1).

The PHD is a representation of object existence probability. It should not be confused with the state's density function, though it has the same function form, and the same numerical representations with particles or Gaussian mixtures have been proposed in literature.

Fig. 4 shows an example where ellipse models are fitted to each object. An additional benefit with having extended objects is that occlusion can be modeled in a direct and natural way. Ultimately, such filters should be able to give very accurate situational awareness including both stationary and moving objects and their extensions in a scenario such as in Fig. 1.

REFERENCES

- [1] Y. Bar-Shalom, P. K. Willett, and X. Tian, *Tracking and data fusion, a handbook of algorithms*. YBS, 2011.
- [2] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping (SLAM): part I," *IEEE Robotics & Automation Magazine*, vol. 13, no. 2, pp. 99–110, Jun. 2006.
- [3] R. Mahler, *Statistical Multisource-Multitarget Information Fusion*. Norwood, MA, USA: Artech House, 2007.
- [4] —, "Multitarget Bayes filtering via first-order multi target moments," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1152–1178, Oct. 2003.
- [5] —, "PHD filters for nonstandard targets, I: Extended targets," in *Proceedings of the International Conference on Information Fusion*, Seattle, WA, USA, Jul. 2009, pp. 915–921.
- [6] A. Swain and D. Clark, "Extended object filtering using spatial independent cluster processes," in *Proceedings of the International Conference on Information Fusion*, Edinburgh, UK, Jul. 2010.
- [7] J. Mullane, B.-N. Vo, M. Adams, and B.-T. Vo, "A Random-Finite-Set Approach to Bayesian SLAM," *IEEE Transactions on Robotics*, vol. 27, no. 2, pp. 268–282, Apr. 2011.
- [8] C. S. Lee, D. Clark, and J. Salvi, "SLAM with dynamic targets via single-cluster PHD filtering," *IEEE Journal of Selected Topics in Signal Processing, Special Issue on Multi-target Tracking*, vol. 7, no. 3, pp. 543–552, 2013.
- [9] J. Neira and J. D. Tardos, "Data association in stochastic mapping using the joint compatibility test," *IEEE Transactions on Robotics*, vol. 17, no. 6, pp. 890–897, Dec. 2001.
- [10] K. Granström, C. Lundquist, and U. Orguner, "Extended Target Tracking using a Gaussian Mixture PHD filter," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 4, pp. 3268–3286, Oct. 2012.
- [11] K. Granström and U. Orguner, "A PHD filter for tracking multiple extended targets using random matrices," *IEEE Transactions on Signal Processing*, vol. 60, no. 11, pp. 5657–5671, Nov. 2012.
- [12] K. Gilholm and D. Salmond, "Spatial distribution model for tracking extended objects," *IEE Proceedings Radar, Sonar and Navigation*, vol. 152, no. 5, pp. 364–371, Oct. 2005.
- [13] J. W. Koch, "Bayesian approach to extended object and cluster tracking using random matrices," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 44, no. 3, pp. 1042–1059, Jul. 2008.
- [14] M. Baum and U. D. Hanebeck, "Shape Tracking of Extended Objects and Group Targets with Star-Convex RHM's," in *Proceedings of the International Conference on Information Fusion*, Chicago, IL, USA, Jul. 2011, pp. 338–345.
- [15] C. Lundquist, K. Granström, and U. Orguner, "Estimating the Shape of Targets with a PHD Filter," in *Proceedings of the International Conference on Information Fusion*, Chicago, IL, USA, Jul. 2011, pp. 49–56.
- [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. Granström, C. Lundquist, and U. Orguner, "Tracking Rectangular and Elliptical Extended Targets Using Laser Measurements," in *Proceedings of the International Conference on Information Fusion*, Chicago, IL, USA, Jul. 2011, pp. 592–599.
- [18] B.-T. Vo, B.-N. Vo, and A. Cantoni, "Analytic implementations of the cardinalized probability hypothesis density filter," *IEEE Transactions on Signal Processing*, vol. 55, no. 7, pp. 3553–3567, Jul. 2007.
- [19] C. Lundquist, K. Granström, and U. Orguner, "An extended target CPHD filter and a gamma Gaussian inverse Wishart implementation," *IEEE Journal of Selected Topics in Signal Processing, Special Issue on Multi-target Tracking*, vol. 7, no. 3, pp. 472–483, Jun. 2013.
- [20] X.-R. Li and V. Jilkov, "Survey of maneuvering target tracking: Part I. Dynamic models," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1333–1364, Oct. 2003.