

# Estimating the Shape of Targets with a PHD Filter

Christian Lundquist, Karl Granström, Umut Orguner  
Department of Electrical Engineering  
Linköping University  
583 33 Linköping, Sweden  
Email: {lundquist, karl, umut}@isy.liu.se

**Abstract**—This paper presents a framework for tracking extended targets which give rise to a structured set of measurements per each scan. The concept of a measurement generating point (MGP) which is defined on the boundary of each target is introduced. The tracking framework contains an hybrid state space where MGP:s and the measurements are modeled by random finite sets and target states by random vectors. The target states are assumed to be partitioned into linear and nonlinear components and a Rao-Blackwellized particle filter is used for their estimation. For each state particle, a probability hypothesis density (PHD) filter is utilized for estimating the conditional set of MGP:s given the target states. The PHD kept for each particle serves as a useful means to represent information in the set of measurements about the target states. The early results obtained show promising performance with stable target following capability and reasonable shape estimates.

**Keywords:** Tracking, data association, particle filter, Kalman filter, estimation, PHD filter, extended target, Rao-Blackwellized particle filter.

## I. INTRODUCTION

In target tracking, the task is to detect, track and identify an unknown number of targets using measurements that are affected by noise and clutter. In recent years, so called extended target tracking has received increasing research attention. In extended target tracking, the classic point target assumption is relaxed and the target tracking framework is modified to handle multiple measurements per target and time step. The multiple measurements per target raise interesting possibilities to estimate the target's size and shape in addition to the target's position, velocity and heading. Typical sensors where targets cause multiple measurements are cameras, automotive radars and laser range sensors – especially cameras and laser range sensors give measurements with a high degree of structure.

Using finite set statistics (FISST) Mahler has derived rigorous tools for multiple target tracking, see [1] and [2]. In the Probability Hypothesis Density filter (PHD-filter) the targets and measurements are treated as random finite sets (RFS); an implementation where the PHD-intensity is approximated using a mixture of Gaussians has been presented in [3]. An extension of the PHD-filter to handle extended targets that give Poisson distributed measurements, as in [4], was given in [5]. A Gaussian mixture implementation of this extended target PHD-filter was recently presented in [6].

In the recent work by Mullane *et al.* [7], the Simultaneous Localization and Mapping (SLAM) problem was addressed using a Bayesian approach. The measurements and feature

map is modeled using RFS, giving a framework in which the number and location of map features is jointly estimated with the vehicle trajectory. A Rao-Blackwellized implementation is suggested where the vehicle trajectory is estimated with a particle filter and the map is handled using a Gaussian-mixture (GM-PHD) filter.

In this paper, the ideas presented in [7] are utilized to solve the problem of estimating the shape of an extended target. The sensors mentioned earlier typically obtain point measurements from reflection points on the surface of the target. The realizations of the reflection points in this framework are called *measurement generating points* (MGP:s) and their positions on the target are estimated as a means to estimate the shape, size and position of the target. By considering the MGP as an RFS it is possible to create a measurement model which better adapts to the actual and visible reflection points of the target. The target state vector is estimated using a particle filter, and the RFS of MGP:s are estimated with a GM-PHD-filter. The target's state vector is too large to be efficiently estimated with a particle filter, and the linear and nonlinear parts are therefore partitioned and estimated in a marginalized or Rao-Blackwellized particle filter, see e.g., [8]. The joint estimation of the target density and the density of the MGP:s on the boundary leads to a Rao-Blackwellized implementation of the joint particle and PHD filter.

Modeling of extended targets in this work is very similar to active contours [9] and snakes [10], which model the outline of an object based on 2 dimensional image data, studied extensively in computer vision. Detecting and identifying shapes in cluttered point clouds has been studied in [11], where the MGP:s on the surface of the target are denoted samples. The underlying low-level processing involved assumes reasonably the existence of image data from which features (like Harris corners) or feature maps (like Harris measures etc.) can be extracted. The so-called Condensation algorithm [12], for example, searches for suitable sets of features in the image data (or in the feature map) iteratively for each of its different hypotheses (particles) in the calculation of the likelihoods. The approach presented in this work carries the distinction of working only with a single set of measurement points provided most of the time by thresholding of the raw sensor data (like conventional target tracking) and hence is mainly aimed at applications where the users (of the sensors) either are unable to reach or do not have the computation power to work with the raw sensor data. The mapping of the boundaries of complex

objects is also achieved in [13] using splinegon techniques and a range sensor such as e.g., laser mounted on moving vehicles.

There are also several other approaches to extended target tracking in the literature. Gilholm presents in [4], [14] an approach where it is assumed that the number of received target measurements is Poisson distributed, hence several measurements may originate from the same target. In [15] a similar approach is presented where raw data is considered in a track-before-detect framework and no data association is needed. Monte Carlo methods are applied to solve the extended target tracking problem in [16], [17] and random matrices are used by Koch in [18].

The paper is organized as follows. Section II introduces the RFS of measurements and MGP:s as well as the model of the target. The filter framework is described in Section III and an implementation is given in Section IV. A simple example with simulation results is shown in Section V. Section VI contains conclusions and some thoughts on future work.

## II. THE RFS EXTENDED TARGET MODEL

Consider an extended target whose characteristics, e.g., position, heading, velocity and shape are described by a state vector  $\mathbf{x}_k$ . The target is observed by a sensor, which provides point observations  $\mathbf{z}_k^{(m)}$  of the target. The number of measurements  $z_k$  at any given time is not fixed due to detection uncertainty, spurious measurements and unknown number of reflection points on the surface of the target. Since the target is extended, it potentially gives rise to more than one measurement per time step. Hence, the measurements obtained from one target may be represented by a random finite set (RFS) of observations

$$\mathbf{Z}_k = \left\{ \mathbf{z}_k^{(1)}, \dots, \mathbf{z}_k^{(z_k)} \right\} \quad (1)$$

where  $k$  refers to discrete time. One way to associate the point measurements to the target is to consider a number of reflection points, called measurement generating points (MGP) in this work, on the surface of the target. A MGP is defined on a one dimensional coordinate axis denoted as  $s \in \mathbb{R}$  on the boundary of the target, which can generally be restricted to an interval  $[s_{\min}, s_{\max}]$  called the MGP-space. All MGP:s belonging to a single target may be modeled as a finite set shown as

$$\mathbf{S}_k = \left\{ s_k^{(1)}, \dots, s_k^{(s_k)} \right\}. \quad (2)$$

A simple example is shown in Figure 1, where visible MGP:s and point measurements are illustrated.

In a Bayesian framework, the states and the measurements are treated as realizations of random variables. Since in this case both the measurements and the MGP:s are represented by finite sets, the concept of RFS is required for the Bayesian estimation of the target state. An RFS is a finite set valued random variable, whose elements are random variables and the number of elements are treated as a random non-negative integer variable. In a filtering and estimation framework the probability density is a very useful descriptor of an RFS.

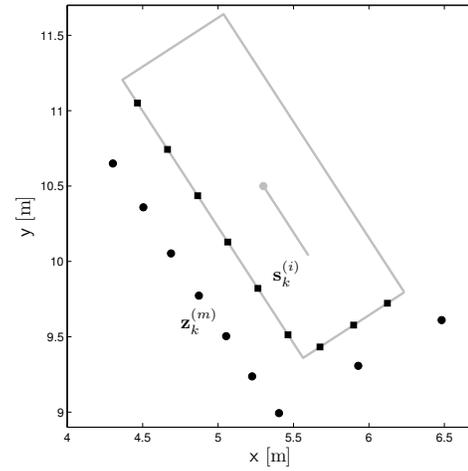


Figure 1. A shape with visible MGP:s  $\mathbf{S}_k$  (■) and measurements  $\mathbf{Z}_k$  (●).

Standard tools and notation to compute densities and integrations of random variables are not appropriate for RFS and therefore finite set statistics FISST, developed by Mahler [1] as a practical mathematical tool to deal with RFS, has to be applied.

Consider again the target in Figure 1, and let  $\mathbf{S}$  be the RFS of all MGP:s on the surface of the target. Furthermore let  $\mathbf{S}_k \subset \mathbf{S}$  be the detectable MGP:s. As the target moves new MGP:s might become visible to the sensor, either by observing a new side of the object or by observing new reflection points, e.g., a wheel arch of a car, becoming visible when the angle between the target and the sensor changes. The new MGP:s are modeled by an independent RFS  $\mathbf{B}_k(\mathbf{x}_k) \subset \mathbf{S}$  depending on the target state  $\mathbf{x}_k$ . The set of visible MGP:s evolves in time as

$$\mathbf{S}_k = \mathbf{F}_k(\mathbf{S}_{k-1}) \cup \mathbf{B}_k(\mathbf{x}_k). \quad (3)$$

where  $\mathbf{F}_k(\cdot)$  is a set function modeling the survival (or death) of the previous visible MGP:s. The RFS transition density is then given by

$$p(\mathbf{S}_k | \mathbf{S}_{k-1}, \mathbf{x}_k) = \sum_{\bar{\mathbf{S}} \subseteq \mathbf{S}_k} p_{\mathbf{S}}(\bar{\mathbf{S}} | \mathbf{S}_{k-1}) p_{\mathbf{B}}(\mathbf{S}_k - \bar{\mathbf{S}} | \mathbf{x}_k) \quad (4)$$

where  $p_{\mathbf{S}}(\cdot | \mathbf{S}_{k-1})$  is the transition density of the observable set of MGP:s, i.e., those who are in the field of view of the sensor, and  $p_{\mathbf{B}}(\cdot | \mathbf{x}_k)$  is the density of the RFS  $\mathbf{B}_k(\mathbf{x}_k)$  of the new visible MGP:s. Furthermore, the target dynamics is modeled by a standard Markov process with transition density  $p(\mathbf{x}_k | \mathbf{x}_{k-1})$  and the joint transition density of the target and its MGP:s is

$$p(\mathbf{S}_k, \mathbf{x}_k | \mathbf{S}_{k-1}, \mathbf{x}_{k-1}) = p(\mathbf{S}_k | \mathbf{S}_{k-1}, \mathbf{x}_k) p(\mathbf{x}_k | \mathbf{x}_{k-1}). \quad (5)$$

The RFS of measurements  $\mathbf{Z}_k$  received at time  $k$  by the sensor from a target with state vector  $\mathbf{x}_k$  and RFS  $\mathbf{S}_k$  of MGP is modeled as

$$\mathbf{Z}_k = \bigcup_{s \in \mathbf{S}_k} \mathbf{H}_k(s, \mathbf{x}_k) \cup \mathbf{C}_k \quad (6)$$

where  $\mathbf{H}_k(\mathbf{s}, \mathbf{x}_k)$  is the RFS of measurement produced by the MGP:s modeling also the non-perfect detection process and  $\mathbf{C}$  is the clutter RFS. The RFS of measurements produced by a MGP is modeled by a Bernoulli RFS, i.e., it is  $\mathbf{H}_k(\mathbf{s}, \mathbf{x}_k) = \emptyset$  with probability  $1 - P_D(\mathbf{s}|\mathbf{x}_k)$  and  $\mathbf{H}_k(\mathbf{s}, \mathbf{x}_k) = \{\mathbf{z}\}$  with probability density  $P_D(\mathbf{s}|\mathbf{x}_k)p(\mathbf{z}|\mathbf{s}, \mathbf{x}_k)$ , cf. [2]. The probability of detection is denoted by  $P_D(\mathbf{s}|\mathbf{x}_k)$  defined over the MGP-space, and  $p(\mathbf{z}|\mathbf{s}, \mathbf{x}_k)$  is the likelihood that the MGP  $\mathbf{s}$  produces the measurement  $\mathbf{z}$ . The likelihood function of the measurement RFS  $\mathbf{Z}_k$  is

$$p(\mathbf{Z}_k|\mathbf{x}_k, \mathbf{S}_k) = \sum_{\mathbf{Z} \subseteq \mathbf{Z}_k} p_{\mathbf{Z}}(\bar{\mathbf{Z}}|\mathbf{S}_k, \mathbf{x}_k)p_{\mathbf{C}}(\mathbf{Z}_k - \bar{\mathbf{Z}}|\mathbf{x}_k) \quad (7)$$

where  $p_{\mathbf{Z}}(\bar{\mathbf{Z}}|\mathbf{S}_k, \mathbf{x}_k)$  is the density representing the RFS of detected observations. The density of the clutter RFS  $\mathbf{C}$  is denoted  $p_{\mathbf{C}}(\cdot|\mathbf{x}_k)$  which is Poisson in cardinality and uniform in space.

The relation between the shape of the target and some of the state components is highly nonlinear and this makes an application of the particle filter suitable. Suppose that the target state  $\mathbf{x}_k$  can be partitioned into a nonlinear and a linear part according to

$$\mathbf{x}_k = [\mathbf{x}_k^n \quad \mathbf{x}_k^l]^T, \quad (8)$$

the state space model for  $\mathbf{x}_k$  is given by

$$\mathbf{x}_{k+1}^n = f_k^n(\mathbf{x}_k^n) + F_k^n(\mathbf{x}_k^n)\mathbf{x}_k^l + \mathbf{v}_k^n \quad (9a)$$

$$\mathbf{x}_{k+1}^l = f_k^l(\mathbf{x}_k^n) + F_k^l(\mathbf{x}_k^n)\mathbf{x}_k^l + \mathbf{v}_k^l. \quad (9b)$$

For a single measurement, the measurement model is given by

$$\mathbf{z}_k \sim \begin{cases} \mathcal{N}(\cdot; \hat{\mathbf{z}}_k(\mathbf{x}_k^n, \mathbf{s}_k), R_k) & \text{if } \mathbf{z} \text{ target generated} \\ c_k(\cdot) & \text{if } \mathbf{z} \text{ clutter generated} \end{cases} \quad (9c)$$

where

$$\hat{\mathbf{z}}_k(\mathbf{x}_k^n, \mathbf{s}_k) = h_k^n(\mathbf{x}_k^n, \mathbf{s}_k) + H_k(\mathbf{x}_k^n)\mathbf{x}_k^l \quad (9d)$$

The process noise  $\mathbf{v}_k$  is assumed to be

$$\mathbf{v}_k = \begin{bmatrix} \mathbf{v}_k^n \\ \mathbf{v}_k^l \end{bmatrix} \sim \mathcal{N}(0, Q_k), \quad Q_k = \begin{bmatrix} Q_k^n & 0 \\ 0 & Q_k^l \end{bmatrix} \quad (10)$$

and the process covariance is here assumed to be diagonal. The measurement noise is modeled as  $\mathbf{e}_k \sim \mathcal{N}(0; R_k)$ . The MGP:s  $\mathbf{s}_k^{(i)}$  are assumed to be stationary on the boundary of the target and the motion model for them is selected to be a random walk model.

### III. FILTERING FRAMEWORK

The aim of the filtering algorithm is to estimate the joint posterior  $p(\mathbf{x}_{1:k}^n, \mathbf{x}_{1:k}^l, \mathbf{S}_k|\mathbf{Z}_{0:k})$  given the set of measurements  $\mathbf{Z}_{0:k}$ . The posterior may be factorized according to

$$p(\mathbf{x}_{1:k}^n, \mathbf{x}_{1:k}^l, \mathbf{S}_k|\mathbf{Z}_{0:k}) = p(\mathbf{x}_k^l|\mathbf{x}_{0:k}^n, \mathbf{S}_k, \mathbf{Z}_{0:k})p(\mathbf{S}_k|\mathbf{x}_{0:k}^n, \mathbf{Z}_{0:k})p(\mathbf{x}_{1:k}^n|\mathbf{Z}_{0:k}) \quad (11)$$

and the three posteriors for the states of a target  $\mathbf{x}_k^l$ ,  $\mathbf{x}_k^n$  and the MGP:s  $\mathbf{S}_k = \{\mathbf{s}_k^{(1)}, \dots, \mathbf{s}_k^{(s_k)}\}$  can be computed separately. However, since the filter recursion needed to estimate the probability density of the MGP RFS  $p(\mathbf{S}_k|\mathbf{x}_{0:k}^n, \mathbf{Z}_{0:k})$  includes

set integrals it is numerically intractable, and it is necessary to find a tractable approximation. The first order moment of an RFS is represented by a PHD, denoted  $D$ . For the RFS  $\mathbf{S}_k$ , the PHD is a nonnegative function, such that for each region  $S$  in the space of MGP:s

$$\int_S D_k(\mathbf{x})d\mathbf{x} = \mathbb{E}(|\mathbf{S}_k \cap S|) = \hat{s}_k \quad (12)$$

where  $\hat{s}_k$  is the expected number of elements in the region  $S$ , and the peaks of the PHD indicates locations of MGP:s with high probability. If the cardinality of elements in the RFS  $\mathbf{S}_k$  is Poisson distributed and the elements are i.i.d., then the probability density of  $\mathbf{S}_k$  can be computed from the PHD according to

$$p(\mathbf{S}_k) = \frac{\prod_{\mathbf{s} \in \mathbf{S}_k} D_k(\mathbf{s})}{\exp(\int D_k(\mathbf{s})d\mathbf{s})}. \quad (13)$$

The PHD-filter propagates the PHD of an RFS in time and provides a reasonable approximation to the multitarget Bayesian filter. The filter recursion for the joint RFS and target state posterior density is described in the following. First the target state and the MGP:s are predicted as described in section III-A and thereafter a measurement update of the MGP:s and the target states is performed as described in Section III-B.

#### A. Prediction

Given the prior

$$p(\mathbf{x}_{1:k-1}^n, \mathbf{x}_{1:k-1}^l, \mathbf{S}_{k-1}|\mathbf{Z}_{0:k-1}) = p(\mathbf{x}_{1:k-1}^l|\mathbf{x}_{0:k-1}^n, \mathbf{S}_{k-1}, \mathbf{Z}_{0:k-1}) \times p(\mathbf{S}_{k-1}|\mathbf{x}_{0:k-1}^n, \mathbf{Z}_{0:k-1})p(\mathbf{x}_{1:k-1}^n|\mathbf{Z}_{0:k-1}) \quad (14)$$

1. Predict first the nonlinear state components  $\mathbf{x}^n$ , i.e.,

$$p(\mathbf{x}_{1:k-1}^n|\mathbf{Z}_{0:k-1}) \rightarrow p(\mathbf{x}_{1:k}^n|\mathbf{Z}_{0:k-1}) \quad (15)$$

using (9a).

2. Predict the linear state components  $\mathbf{x}^l$ , i.e.,

$$p(\mathbf{x}_{1:k-1}^l|\mathbf{x}_{0:k-1}^n, \mathbf{Z}_{0:k-1}) \rightarrow p(\mathbf{x}_{1:k}^l|\mathbf{x}_{0:k}^n, \mathbf{Z}_{0:k-1}) \quad (16)$$

using (9a) and (9b). Compare with the marginalized particle filter [8], where the linear components are predicted and updated with a Kalman filter, see Line 3-6 in Table I.

3. Finally predict the MGP:s according to

$$p(\mathbf{S}_{k-1}|\mathbf{x}_{0:k}^n, \mathbf{Z}_{0:k-1}) \rightarrow p(\mathbf{S}_k|\mathbf{x}_{0:k}^n, \mathbf{Z}_{0:k-1}). \quad (17)$$

The predicted RFS is approximated by a Poisson RFS with PHD  $D(\mathbf{s}_k|\mathbf{x}_{0:k}, \mathbf{Z}_{0:k-1})$  as

$$p(\mathbf{S}_k|\mathbf{x}_{0:k}, \mathbf{Z}_{0:k-1}) \approx \frac{\prod_{\mathbf{s} \in \mathbf{S}_k} D(\mathbf{s}|\mathbf{x}_{0:k}, \mathbf{Z}_{0:k-1})}{\exp(\int D(\mathbf{s}|\mathbf{x}_{0:k}, \mathbf{Z}_{0:k-1})d\mathbf{s})} \quad (18)$$

cf. (13). Using this approximation and the motion model (3), the PHD prediction equation is given by

$$D_{k|k-1}(\mathbf{s}|\mathbf{x}_{0:k}) = P_S(\mathbf{s})D_{k-1}(\mathbf{s}|\mathbf{x}_{0:k}) + D_{b,k}(\mathbf{s}) \quad (19)$$

where  $D_{b,k}(\mathbf{s})$  is the PHD of the birth RFS  $\mathbf{B}_k$  and  $P_S(\cdot)$  models the probability of survival of a MGP. Note that the abbreviation

$$D_{k|k-1}(\mathbf{s}|\mathbf{x}_{0:k}) \triangleq D(\mathbf{s}_k|\mathbf{x}_{0:k}, \mathbf{Z}_{0:k-1}) \quad (20)$$

is adopted for clarity.

### B. Measurement Update

4. Update the MGP:s according to

$$p(\mathbf{S}_k|\mathbf{x}_{0:k}^n, \mathbf{Z}_{0:k-1}) \rightarrow p(\mathbf{S}_k|\mathbf{x}_{0:k}^n, \mathbf{Z}_{0:k}) \quad (21)$$

where the update is described by

$$p(\mathbf{S}_k|\mathbf{x}_{0:k}^n, \mathbf{Z}_{0:k}) = \frac{p(\mathbf{Z}_k|\mathbf{S}_k, \mathbf{x}_k^n)p(\mathbf{S}_k|\mathbf{x}_{0:k}^n, \mathbf{Z}_{0:k-1})}{p(\mathbf{Z}_k|\mathbf{Z}_{0:k-1}, \mathbf{x}_{0:k}^n)}. \quad (22)$$

Like the case in the prediction, the posterior is approximated by a Poisson RFS with PHD  $D(\mathbf{s}_k|\mathbf{x}_{0:k}, \mathbf{Z}_{0:k})$  as

$$p(\mathbf{S}_k|\mathbf{x}_{0:k}, \mathbf{Z}_{0:k}) \approx \frac{\prod_{\mathbf{s} \in \mathbf{S}_k} D(\mathbf{s}|\mathbf{x}_{0:k}, \mathbf{Z}_{0:k})}{\exp(\int D(\mathbf{s}|\mathbf{x}_{0:k}, \mathbf{Z}_{0:k})d\mathbf{s})} \quad (23)$$

cf. (13). Using this approximation and the measurement model (6), the PHD corrector equation is given by

$$D_k(\mathbf{s}|\mathbf{x}_{0:k}) = D_{k|k-1}(\mathbf{s}|\mathbf{x}_{0:k}) \left[ 1 - P_D(\mathbf{s}|\mathbf{x}_k) + \sum_{\mathbf{z} \in \mathbf{Z}_k} \frac{\Lambda(\mathbf{z}_k|\mathbf{s}_k, \mathbf{x})}{\lambda c_k(\mathbf{z}) + \int \Lambda(\mathbf{z}_k|\zeta, \mathbf{x}_k)D_{k|k-1}(\zeta|\mathbf{x}_{0:k})d\zeta} \right] \quad (24)$$

where

$$\Lambda(\mathbf{z}_k|\mathbf{s}_k, \mathbf{x}_k) = P_D(\mathbf{s}_k|\mathbf{x}_k)p(\mathbf{z}_k|\mathbf{s}_k, \mathbf{x}_k). \quad (25)$$

For clarity the abbreviation

$$D_k(\mathbf{s}|\mathbf{x}_{0:k}) \triangleq D(\mathbf{s}_k|\mathbf{x}_{0:k}, \mathbf{Z}_{0:k}) \quad (26)$$

is used.

5. Update the nonlinear state components  $\mathbf{x}^n$

$$p(\mathbf{x}_{1:k}^n|\mathbf{Z}_{0:k-1}) \rightarrow p(\mathbf{x}_{1:k}^n|\mathbf{Z}_{0:k}) \quad (27)$$

where the update is modeled as

$$p(\mathbf{x}_{1:k}^n|\mathbf{Z}_{0:k}) = p(\mathbf{Z}_k|\mathbf{Z}_{0:k-1}, \mathbf{x}_{0:k}^n) \times \frac{p(\mathbf{x}_k^n|\mathbf{x}_{k-1}^n)p(\mathbf{x}_{1:k-1}^n|\mathbf{Z}_{0:k-1})}{p(\mathbf{Z}_k|\mathbf{Z}_{0:k-1})}. \quad (28)$$

The term  $p(\mathbf{Z}_k|\mathbf{Z}_{0:k-1}, \mathbf{x}_{0:k}^n)$  is given by a set integration over  $\mathbf{S}_k$ , according to

$$p(\mathbf{Z}_k|\mathbf{Z}_{0:k-1}, \mathbf{x}_{0:k}^n) = \int p(\mathbf{Z}_k, \mathbf{S}_k|\mathbf{Z}_{0:k-1}, \mathbf{x}_{0:k}^n)\delta\mathbf{S}_k. \quad (29)$$

To avoid the set integral, note that  $p(\mathbf{Z}_k|\mathbf{Z}_{0:k-1}, \mathbf{x}_{0:k}^n)$  is the normalization constant in (22), hence it may be rewritten according to

$$p(\mathbf{Z}_k|\mathbf{Z}_{0:k-1}, \mathbf{x}_{0:k}^n) = \frac{p(\mathbf{Z}_k|\mathbf{S}_k, \mathbf{x}_k^n)p(\mathbf{S}_k|\mathbf{x}_{0:k}^n, \mathbf{Z}_{0:k-1})}{p(\mathbf{S}_k|\mathbf{x}_{0:k}^n, \mathbf{Z}_{0:k})}. \quad (30)$$

Note that  $\mathbf{S}_k$  is only contained in the RHS, thus the relation holds for arbitrary choices of  $\mathbf{S}_k$ . We substitute here the last estimated  $\mathbf{S}_k$  by making further approximations as follows. Using the Poisson RFS approximations (18) and (23) and the RFS measurement likelihood (7) it holds that

$$\begin{aligned} p(\mathbf{Z}_k|\mathbf{Z}_{0:k-1}, \mathbf{x}_{0:k}^n) &= \frac{p(\mathbf{Z}_k|\mathbf{S}_k, \mathbf{x}_k^n)p(\mathbf{S}_k|\mathbf{x}_{0:k}^n, \mathbf{Z}_{0:k-1})}{p(\mathbf{S}_k|\mathbf{x}_{0:k}^n, \mathbf{Z}_{0:k})} \\ &\approx \underbrace{p(\mathbf{Z}_k|\mathbf{S}_k, \mathbf{x}_k^n)}_{\triangleq A} \frac{\prod_{\mathbf{s} \in \mathbf{S}_k} D_{k|k-1}(\mathbf{s}|\mathbf{x}_{0:k}) \exp(\int D_{k|k-1}(\mathbf{s}|\mathbf{x}_{0:k})d\mathbf{s})}{\prod_{\mathbf{s} \in \mathbf{S}_k} D_k(\mathbf{s}|\mathbf{x}_{0:k}) \exp(\int D_k(\mathbf{s}|\mathbf{x}_{0:k})d\mathbf{s})} \\ &= A \underbrace{\frac{\prod_{\mathbf{s} \in \mathbf{S}_k} D_{k|k-1}(\mathbf{s}|\mathbf{x}_{0:k})}{\prod_{\mathbf{s} \in \mathbf{S}_k} D_k(\mathbf{s}|\mathbf{x}_{0:k})}}_{\triangleq B} \underbrace{\frac{\exp(\int D_k(\mathbf{s}|\mathbf{x}_{0:k})d\mathbf{s})}{\exp(\int D_{k|k-1}(\mathbf{s}|\mathbf{x}_{0:k})d\mathbf{s})}}_{\triangleq C} \quad (31a) \end{aligned}$$

where

$$\begin{aligned} A &\approx \prod_{\mathbf{z} \in \mathbf{Z}} (\lambda c(\mathbf{z}) + P_D(\mathbf{s}_k^{(i)})p(\mathbf{z}^{(i)}|\mathbf{s}_k^{(i)}, \mathbf{x}_k^n)) \\ &\quad \times \prod_{\mathbf{s} \in \mathbf{S}_k} (1 - P_D(\mathbf{s}_k^{(i)})) \quad (31b) \end{aligned}$$

with the  $i$ th MGP being the one closest to measurement  $i$  according to

$$\mathbf{s}^{(i)} = \arg \min_{\mathbf{s} \in \mathbf{S}_k} \|\mathbf{z}_k^{(i)} - \mathbf{s}\| \quad (31c)$$

$$\tilde{\mathbf{S}}_k = \mathbf{S}_k \setminus \{\mathbf{s}^{(i)}\}_{i=1}^{\hat{s}_k} \quad (31d)$$

Furthermore,

$$B = \prod_{i=1}^{\hat{s}_k} \frac{1}{(1 - P_D(\mathbf{s}^{(i)})) + \sum_{\mathbf{z} \in \mathbf{Z}} \frac{\Lambda(\mathbf{z}|\mathbf{s}_k^{(i)}, \mathbf{x}_k)}{\lambda c(\mathbf{z}) + \int \Lambda(\mathbf{z}|\zeta, \mathbf{x}_k)D_k(\zeta|\mathbf{x}_{0:k})d\zeta}} \quad (31e)$$

$$C = \exp(\hat{s}_k - \hat{s}_{k|k-1}) \quad (31f)$$

and  $\Lambda(\cdot)$  is given in (25). The number of predicted and updated MGP:s are

$$\hat{s}_k = \int D_k(\mathbf{s}_k|\mathbf{x}_{0:k}, \mathbf{Z}_{0:k})d\mathbf{s}_k \quad (32a)$$

$$\hat{s}_{k|k-1} = \int D_{k|k-1}(\mathbf{s}_k|\mathbf{x}_{0:k}, \mathbf{Z}_{0:k-1})d\mathbf{s}_k. \quad (32b)$$

6. Update the linear state components  $\mathbf{x}^l$

$$p(\mathbf{x}_{1:k}^l|\mathbf{x}_{0:k}^n, \mathbf{S}_k, \mathbf{Z}_{0:k-1}) \rightarrow p(\mathbf{x}_{1:k}^l|\mathbf{x}_{0:k}^n, \mathbf{S}_k, \mathbf{Z}_{0:k}) \quad (33)$$

The update of  $\mathbf{x}_{1:k}^l$  is conditioned on the RFS of the MGP:s  $\mathbf{S}_k$ . As an approximation, the posterior of  $\mathbf{x}^l$  is written as

$$p(\mathbf{x}_{1:k}^l|\mathbf{x}_{0:k}^n, \mathbf{S}_k, \mathbf{Z}_{0:k}) \approx p(\mathbf{x}_{1:k}^l|\mathbf{x}_{0:k}^n, \hat{\mathbf{S}}_k, \mathbf{Z}_{0:k}) \quad (34)$$

where the estimate  $\hat{\mathbf{S}}_k$  is calculated from the posterior PHD  $D_k(\mathbf{s}|\mathbf{x}_{0:k})$ . The linear components are updated as described in Line 31 to 36 in Table II.

7. The joint posterior is now given by multiplying the separate posteriors from (21), (27) and (33) as

$$p(\mathbf{x}_{1:k}^n, \mathbf{x}_{1:k}^l, \mathbf{S}_k | \mathbf{Z}_{0:k}) \approx p(\mathbf{x}_{1:k}^l | \mathbf{x}_{0:k}^n, \hat{\mathbf{S}}_k, \mathbf{Z}_{0:k}) p(\mathbf{S}_k | \mathbf{x}_{0:k}^n, \mathbf{Z}_{0:k}) p(\mathbf{x}_{1:k}^n | \mathbf{Z}_{0:k}) \quad (35)$$

An implementation of the proposed filter recursion is presented in the next section.

#### IV. RBPF-PHD IMPLEMENTATION

A Rao-Blackwellized implementation is utilized to estimate the target state. The nonlinear target state  $\mathbf{x}^n$  is propagated with a particle filter and the linear state vector is propagated with a Kalman filter. The MGP:s are described by a set  $\mathbf{S}_k$ , which are estimated with a PHD-filter. There exists one PHD  $D_k^{(i)}(\cdot)$  for each particle  $i$  of the target state. The overall summary statistics is represented by

$$\left\{ w_k^{(i)}, \mathbf{x}_{0:k}^{n,(i)}, \hat{\mathbf{x}}_k^{l,(i)}, P_k^{l,(i)}, D_k^{(i)}(\mathbf{s} | \mathbf{x}_{0:k}^{(i)}) \right\}_{i=1}^N \quad (36)$$

where

- $\mathbf{x}_{0:k}^{n,(i)}$  is the  $i$ th particle for the nonlinear part of the target state.
- $\hat{\mathbf{x}}_k^{l,(i)}, P_k^{l,(i)}$  are the  $i$ th mean and covariance for the linear part of the target state.
- $w_k^{(i)}$  is the weights for the  $i$ th particle of the target state.
- $D_k^{(i)}(\mathbf{s} | \mathbf{x}_{0:k}^{(i)})$  is the PHD representing the MGP:s for the  $i$ th particle of the target state.

In this work the PHD  $D_k^{(i)}(\mathbf{s} | \mathbf{x}_{0:k}^{(i)})$  is represented with by a Gaussian mixture and a realization of the GM-PHD filter recursion is described in Section IV-A, see also [3]. The update of the nonlinear state components based on the PHD is described in Section IV-B, and a pseudo code for the proposed filter is given in Section IV-C.

##### A. PHD Prediction and Update

The MGP:s on the surface of the  $i$ th particle are approximated with a PHD  $D_k^{(i)}(\mathbf{s} | \mathbf{x}_k)$  represented by a Gaussian mixture. The prior PHD is given by

$$D_{k-1}^{(i)}(\mathbf{s} | \mathbf{x}_{k-1}^{(i)}) = \sum_{j=1}^{J_{k-1}^{(i)}} \eta_{k-1}^{(i,j)} \mathcal{N}(\mathbf{s}, \mu_{k-1}^{(i,j)}, P_{k-1}^{(i,j)}). \quad (37)$$

This is a mixture of  $J_{k-1}^{(i)}$  Gaussian components, with  $\eta_{k-1}^{(i,j)}$ ,  $\mu_{k-1}^{(i,j)}$  and  $P_{k-1}^{(i,j)}$  being the weights, means and covariances, respectively, for the  $j$ th Gaussian component of the  $i$ th particle.

The prediction of MGP:s in the motion model (3) is given by the union of prior MGP:s and new MGP, which is approximated by the PHD prediction equation (19). The resulting predicted PHD represented by a Gaussian mixture is then given as

$$D_{k|k-1}^{(i)}(\mathbf{s} | \mathbf{x}_k^{(i)}) = \sum_{j=1}^{J_{k|k-1}^{(i)}} \eta_{k|k-1}^{(i,j)} \mathcal{N}(\mathbf{s}, \mu_{k|k-1}^{(i,j)}, P_{k|k-1}^{(i,j)}). \quad (38)$$

where  $J_{k|k-1}^{(i)} = J_{k-1}^{(i)} + J_{b,k}^{(i)}$  and  $J_{b,k}^{(i)}$  is the number of new Gaussian components. The posterior PHD is computed in the GM form as

$$D_k^{(i)}(\mathbf{s} | \mathbf{x}_k^{n,(i)}) = D_{k|k-1}^{n,(i)}(\mathbf{s} | \mathbf{x}_k^{n,(i)}) \left( 1 - p_D(\mathbf{s} | \mathbf{x}_k^{n,(i)}) \right) + \sum_{\mathbf{z} \in \mathbf{Z}_k} \sum_{j=1}^{J_{k|k-1}^{(i)}} D_{G,k}^{(i,j)}(\mathbf{z}, \mathbf{s} | \mathbf{x}_k^{n,(i)}) \quad (39)$$

where the components are given by

$$D_{G,k}^{(i,j)}(\mathbf{z}, \mathbf{s} | \mathbf{x}_k^{n,(i)}) = \eta_{k|k}^{(i,j)} \mathcal{N}(\mathbf{s}; \mu_{k|k}^{(i,j)}, P_{k|k}^{(i,j)}) \quad (40a)$$

$$\eta_{k|k}^{(i,j)} = \frac{P_D(\mathbf{s}) \eta_{k|k-1}^{(i,j)} q^{(i,j)}(\mathbf{z}, \mathbf{x}_k^{n,(i)})}{\lambda c(\mathbf{z}) + \sum_{\ell=1}^{J_{k|k-1}^{(i)}} P_D(\mathbf{s}) \eta_{k|k-1}^{(i,\ell)} q^{(i,\ell)}(\mathbf{z}, \mathbf{x}_k^{n,(i)})} \quad (40b)$$

where  $q^{(i,j)}(\mathbf{z}, \mathbf{x}_k^{n,(i)}) = \mathcal{N}(\mathbf{z}; \hat{\mathbf{z}}(\mathbf{x}_k^{n,(i)}, \eta_{k|k-1}^{(i,j)}, S_k^{(i,j)}))$  and where  $\hat{\mathbf{z}}(\cdot)$  is given in (9d). The terms  $\eta_{k|k}^{(i,j)}$ ,  $P_{k|k}^{(i,j)}$  and  $S_k^{(i,j)}$  can be obtained using standard filtering techniques, such as EKF or UKF. The clutter density is  $c(\mathbf{z}) = \mathcal{U}(\mathbf{z})$ , where  $\lambda_c$  is the average number of clutter measurements and  $\mathcal{U}(\mathbf{z})$  is a uniform distribution on the measurement space.

##### B. Particle Update

The transition density is chosen as the proposal distribution and hence the particles are sampled as

$$\mathbf{x}_k^{n,(i)} \sim p(\mathbf{x}_k^{n,(i)} | \mathbf{x}_{0:k-1}^{n,(i)}) \quad (41a)$$

$$w_k^{(i)} = p(\mathbf{Z}_k | \mathbf{Z}_{0:k-1}, \mathbf{x}_{0:k}^{(i)}) w_{k-1}^{(i)} \quad (41b)$$

where the prediction can be done using (9a) with the substitution of the last estimated values of the linear components. Note that the linear components are treated as noise here and that therefore the process noise is  $Q_k^n + F_k^n(\mathbf{x}_{k-1}^{n,(i)}) P_{k-1}^{l,(i)} (F_k^n(\mathbf{x}_{k-1}^{n,(i)}))^T$  when sampling the particles. The update of the weights can be computed according to (31), where

$$\hat{\mathbf{s}}_{k|k-1}^{(i)} = \sum_{j=1}^{J_{k|k-1}^{(i)}} \eta_{k|k-1}^{(i,j)} \quad \text{and} \quad \hat{\mathbf{s}}_k^{(i)} = \sum_{j=1}^{J_{k|k}^{(i)}} \eta_k^{(i,j)} \quad (42)$$

are the sums of the predicted and updated PHD weights for the  $i$ th particle.

##### C. Algorithm

The algorithm is given in Table I and II. The state estimates are extracted by taking the weighted mean

$$\hat{\mathbf{x}}_k = \frac{1}{\sum w_k^{(i)}} \sum_{i=1}^N w_k^{(i)} \mathbf{x}_k^{(i)}. \quad (43)$$

The MGP:s must not be extracted since they are only considered as a mean to estimate the target state.

Table I  
PREDICTION

---

**Require:**  $\{w_{k-1}^{(i)}, \mathbf{x}_{k-1}^{n,(i)}, \hat{\mathbf{x}}_{k-1}^{l,(i)}, P_{k-1}^{(i)}, D_{k-1}^{(i)}\}_{i=1}^N$ , where the PHD is composed of  $D_{k-1}^{(i)} = \{\eta_{k-1}^{i,j}, \mu_{k-1}^{(i,j)}, \Sigma_{k-1}^{(i,j)}\}_{j=1}^{J_{k-1}^{(i)}}$

- 1: **for**  $i = 1$  to  $N$  **do**
- 2: **Predict**  $\mathbf{x}^n$   
 $\mathbf{x}_{k|k-1}^{n,(i)} \sim p(\mathbf{x}_{1:k}^{n,(i)} | \mathbf{x}_{0:k-1}^{n,(i)}, \hat{\mathbf{x}}_{k-1}^{l,(i)})$  {cf. (9a)}
- 3: **Predict**  $\mathbf{x}^l$   
 $S_{k-1} = F_{k-1}^n P_{k-1}^{(i)} (F_{k-1}^n)^T + Q_{k-1}^n$
- 4:  $K_{k-1} = F_{k-1}^l P_{k-1}^{(i)} (F_{k-1}^l)^T S_{k-1}^{-1}$
- 5:  $\hat{\mathbf{x}}_{k|k-1}^{l,(i)} = F_{k-1}^l \hat{\mathbf{x}}_{k-1}^{l,(i)} + f_{k-1}^l + K_{k-1} (\mathbf{x}_{k|k-1}^{n,(i)} - f_{k-1}^n - F_{k-1}^n \hat{\mathbf{x}}_{k-1}^{l,(i)})$
- 6:  $P_{k|k-1}^{(i)} = F_{k-1}^l P_{k-1}^{(i)} (F_{k-1}^l)^T + Q_{k-1}^l - K_{k-1} S_{k-1} K_{k-1}^{-1}$
- 7: **Predict PHD**  $D$   
 $\ell = 0$
- 8: **for**  $j = 1$  to  $J_{b,k}^{(i)}$  **do**
- 9:  $\ell = \ell + 1$
- 10:  $\eta_{k|k-1}^{(i,\ell)} = \eta_{b,k}^{(i,j)}, \mu_{k|k-1}^{(i,\ell)} = \mu_{b,k}^{(i,j)}, \Sigma_{k|k-1}^{(i,\ell)} = \Sigma_{b,k}^{(i,j)}$
- 11: **end for**
- 12: **for**  $j = 1$  to  $J_k^{(i)}$  **do**
- 13:  $\ell = \ell + 1$
- 14:  $\eta_{k|k-1}^{(i,\ell)} = P_S \eta_{k-1}^{(i,j)}, \mu_{k|k-1}^{(i,\ell)} = \mu_{k-1}^{(i,j)}, \Sigma_{k|k-1}^{(i,\ell)} = \Sigma_{k-1}^{(i,j)} + Q_{k-1}^s$
- 15: **end for**
- 16:  $J_{k|k-1}^{(i)} = \ell, \hat{\mathbf{s}}_{k|k-1}^{(i)} = \sum_{j=1}^{J_{k|k-1}^{(i)}} \eta_{k|k-1}^{(i,j)}$
- 17: **end for**
- 18: **return**  $\{w_{k-1}^{(i)}, \mathbf{x}_{k-1}^{n,(i)}, \hat{\mathbf{x}}_{k-1}^{l,(i)}, P_{k-1}^{(i)}, D_{k-1}^{(i)}\}_{i=1}^N$  and  $\hat{\mathbf{s}}_{k-1}^{(i)}$ ,  
with  $D_{k|k-1}^{(i)} = \{\eta_{k|k-1}^{i,j}, \mu_{k|k-1}^{(i,j)}, \Sigma_{k|k-1}^{(i,j)}\}_{j=1}^{J_{k|k-1}^{(i)}}$

---

## V. SIMULATION EXAMPLE

In this section, we are going to use a simple example to illustrate the filtering solution we propose. In this example, we use the following specific target and measurement models.

- **Target State Model:** The state vector of the target is given by

$$\mathbf{x} = [x \quad y \quad v \quad \psi \quad \mathbf{t}]^T, \quad (44)$$

where  $(x, y)$  is the planar Cartesian position of the target. It may be any point related to the target, however in this example it is assumed to be the center position. Furthermore,  $v$  is the velocity and  $\psi$  is the heading angle. The shape and size of the target is described by the shape component  $\mathbf{t}$ . Considering a simple but common shape e.g., a rectangle, its size may be represented by a length  $l$  and a width  $w$ . In this example a simple coordinated turn motion model is used

$$\mathbf{x}_{k+1} = \mathbf{x}_k + T \mathbf{v}_k \cos(\psi) \quad (45a)$$

$$y_{k+1} = y_k + T \mathbf{v}_k \sin(\psi) \quad (45b)$$

$$\mathbf{v}_{k+1} = \mathbf{v}_k + \mathbf{v}_v \quad (45c)$$

$$\psi_{k+1} = \psi_k + \mathbf{v}_\psi \quad (45d)$$

$$\mathbf{t}_{k+1} = \mathbf{t}_k + \mathbf{v}_t \quad (45e)$$

where  $T$  is the sample time. The heading angle  $\psi$  is clearly modeled as a nonlinear component.

- **Measurements and Shape Model:** Let the point measurement be a Cartesian position, i.e.,

$$\mathbf{z} = [\bar{x} \quad \bar{y}]^T. \quad (46)$$

Table II  
UPDATE

---

**Require:**  $\{w_{k-1}^{(i)}, \mathbf{x}_{k-1}^{n,(i)}, \hat{\mathbf{x}}_{k-1}^{l,(i)}, P_{k-1}^{(i)}, D_{k-1}^{(i)}\}_{i=1}^N$  and  $\hat{\mathbf{s}}_{k-1}^{(i)}$ , with  $D_{k|k-1}^{(i)} = \{\eta_{k|k-1}^{i,j}, \mu_{k|k-1}^{(i,j)}, \Sigma_{k|k-1}^{(i,j)}\}_{j=1}^{J_{k|k-1}^{(i)}}$

- 1: **for**  $i = 1$  to  $N$  **do**
- 2: **Update PHD**  $D$
- 3: **for**  $j = 1$  to  $J_{k|k-1}^{(i)}$  **do**
- 4:  $\hat{\mathbf{z}}_{k|k-1}^{(i,j)} = h_k(\mathbf{x}_{k|k-1}^{n,(i)}, \mu_{k|k-1}^{(i,j)}) + H_k(\mathbf{x}_{k|k-1}^{n,(i)}) \hat{\mathbf{x}}_{k|k-1}^{l,(i)}$  {cf. (9d)}
- 5:  $\nabla H_k = \frac{\partial}{\partial \mathbf{s}} h_k(\mathbf{x}, \mathbf{s}) \Big|_{\mathbf{x}=\mathbf{x}_{k|k-1}^{n,(i)}, \mathbf{s}=\mu_{k|k-1}^{(i,j)}}$  {cf. (48)}
- 6:  $S_k^{n,(i)} = H_k(\mathbf{x}_{k|k-1}^{n,(i)}) P_{k|k-1}^{(i)} H_k^T(\mathbf{x}_{k|k-1}^{n,(i)})$
- 7:  $S_k^{(i,j)} = \nabla H_k \Sigma_{k|k-1}^{(i,j)} (\nabla H_k)^T + S_k^{n,(i)} + R$
- 8:  $K_k^{(i,j)} = \Sigma_{k|k-1}^{(i,j)} (\nabla H_k)^T (S_k^{(i,j)})^{-1}$
- 9: **end for**
- 10: **for** missed detections  $j = 1$  to  $J_{k|k-1}^{(i)}$  **do**
- 11:  $\eta_{k|k-1}^{(i,j)} = (1 - P_D) \eta_{k|k-1}^{(i,j)}, \mu_k^{(i,j)} = \mu_{k|k-1}^{(i,j)}, \Sigma_k^{(i,j)} = \Sigma_{k|k-1}^{(i,j)}$
- 12: **end for**
- 13:  $\ell = 0$
- 14: **for** each detection  $\mathbf{z}_k \in \mathbf{Z}_k$  **do**
- 15:  $\ell = \ell + 1$
- 16: **for**  $j = 1$  to  $J_{k|k-1}^{(i)}$  **do**
- 17:  $\tau^{(j)} = P_D \eta_{k|k-1}^{(i,j)} \mathcal{N}(\mathbf{z}_k; \hat{\mathbf{z}}_{k|k-1}^{(i,j)}, S_k^{(i,j)})$
- 18:  $\mu_k^{(i,\ell)} = \mu_{k|k-1}^{(i,j)} + K_k^{(i,j)} (\mathbf{z}_k - \hat{\mathbf{z}}_{k|k-1}^{(i,j)})$
- 19:  $\Sigma_k^{(i,\ell)} = \Sigma_{k|k-1}^{(i,j)} + K_k^{(i,j)} S_k^{(i,j)} (K_k^{(i,j)})^T$
- 20: **end for**
- 21: **for**  $j = 1$  to  $J_{k|k-1}^{(i)}$  **do**
- 22:  $\eta_k^{(i,\ell)} = \tau^{(j)} / (\lambda c(\mathbf{z}_k) + \sum_{m=1}^{J_{k|k-1}^{(i)}} \tau^{(m)})$
- 23: **end for**
- 24:  $J_k^{(i)} = (\ell + 1) J_{k|k-1}^{(i)}, \hat{\mathbf{s}}_k^{(i)} = \sum_{j=1}^{J_k^{(i)}} \eta_k^{(i,j)}$
- 25: merge and prune Gaussians
- 26: **Update**  $\mathbf{x}^n$   
 $\tilde{w}_k^{(i)} = p(\mathbf{Z}_k | \mathbf{Z}_{0:k-1}, \mathbf{x}_{0:k}^n) w_{k-1}^{(i)}$  {cf. (31)}
- 27: **end for**
- 28:  $w_k^{(i)} = \tilde{w}_k^{(i)} / \sum_{j=1}^N \tilde{w}_k^{(j)}$  {Normalize}
- 29: resample if necessary
- 30: **Update**  $\mathbf{x}^l$
- 31: **for**  $i = 1$  to  $N$  **do**
- 32: extract MGP:s  $\hat{\mathbf{S}}_k^{(i)} = \{\hat{\mathbf{s}}_k^{(i,j)}\}_{j=1}^{J_k^{(i)}}$  from  $D_k^{(i)}$
- 33: compute association matrix  $A$  using e.g., NN
- 34:  $S_k = H_k^l P_{k|k-1}^{(i)} (H_k^l)^T + R_k$
- 35:  $K_k = P_{k|k-1}^{(i)} (H_k^l)^T (S_k)^{-1}$
- 36:  $\hat{\mathbf{x}}_k^{l,(i)} = \hat{\mathbf{x}}_{k|k-1}^{l,(i)} + K_k (A \mathbf{Z} - h_k(\mathbf{x}_{k|k-1}^{n,(i)}, \hat{\mathbf{S}}_k^{(i)}) - H_k(\mathbf{x}_{k|k-1}^{n,(i)}) \hat{\mathbf{x}}_{k|k-1}^{l,(i)})$
- 37:  $P_{k|k}^{(i)} = P_{k|k-1}^{(i)} - K_k S_k (K_k)^T$
- 38: **end for**
- 39: **return**  $\{w_k^{(i)}, \mathbf{x}_k^{n,(i)}, \hat{\mathbf{x}}_k^{l,(i)}, P_k^{(i)}, D_k^{(i)}\}_{i=1}^N$  and  $\hat{\mathbf{s}}_k^{(i)}$ , with  $D_k^{(i)} = \{\eta_k^{i,j}, \mu_k^{(i,j)}, \Sigma_k^{(i,j)}\}_{j=1}^{J_k^{(i)}}$

---

on the border of the target. Common point measurement sensors, such as radar and laser typically measure range and bearing. The polar representation of the sensor data has here been converted to a Cartesian representation. This also means that the measurement noise covariance must be converted and that it not necessarily is diagonal. A MGP is defined on a coordinate  $\mathbf{s}$  along the border of

the target which has a spline representation of order  $d$ . In this case, the measurement model (9d) may be written according to

$$\hat{\mathbf{z}} = h(\mathbf{x}^n, \mathbf{S}) + [\mathbf{x} \quad \mathbf{y}]^T \quad (47a)$$

with

$$h(\mathbf{x}^n, \mathbf{S}) = \text{Rot}(\psi) \begin{bmatrix} \Pi B^\sigma G^\sigma & 0 \\ 0 & \Pi B^\sigma G^\sigma \end{bmatrix} \Gamma \quad (47b)$$

$$\Pi = [1 \quad \mathbf{s} \quad \mathbf{s}^2 \quad \dots \quad \mathbf{s}^{d-1}] \quad (47c)$$

where  $\text{Rot}$  is a rotation matrix,  $B^\sigma$  and  $G^\sigma$  is a basis matrix and a placement matrix, respectively. The vector  $\Gamma$  includes the shape parameters  $\mathbf{t}$ . Spline curves are well described in e.g., [9].

A nice property using spline curves is that they are continuously differentiable, which is important if the EKF is utilized to update the Gaussian components of the PHD, compare with Line 4 in table II. The derivative of the measurement model (47a) with respect to MGP:s is

$$\nabla H_k = \frac{\partial}{\partial \mathbf{s}} h_k(\mathbf{x}^n, \mathbf{s}) \quad (48a)$$

$$= \text{Rot}(\psi) \begin{bmatrix} \nabla \Pi B^\sigma G^\sigma & 0 \\ 0 & \nabla \Pi B^\sigma G^\sigma \end{bmatrix} \Gamma \quad (48b)$$

$$\nabla \Pi = [0 \quad 1 \quad \mathbf{s} \quad \dots \quad (d-1)\mathbf{s}^{d-2}] \quad (48c)$$

In the models above, it is obvious that the center position  $\mathbf{x}$ ,  $\mathbf{y}$  and velocity  $\mathbf{v}$  of the target are linear in the motion and in the measurement model. The heading  $\psi$  and target shape state  $\mathbf{t}$  are nonlinear, hence the state vector may be partitioned as

$$\mathbf{x}^l = [\mathbf{x} \quad \mathbf{y} \quad \mathbf{v}]^T \quad (49a)$$

$$\mathbf{x}^n = [\psi \quad \mathbf{t}]^T. \quad (49b)$$

Comparing (9d) and (47a) the linear measurement matrix is

$$H = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad (50)$$

Note that the MGP:s  $\mathbf{s}$  are given by a one dimensional position along the border, and that they are highly nonlinear in the measurement model (47a) which justifies the use of particle filter.

A rectangle target is considered in this simulation. The shape state is the length  $l$  and width  $w$ . The vector  $\Gamma$  is

$$\Gamma = [l \quad 0 \quad -l \quad -l \quad -l \quad 0 \quad l \quad l \quad l \quad \dots \\ -w \quad -w \quad -w \quad 0 \quad w \quad w \quad w \quad 0 \quad -w]^T \quad (51)$$

A  $d = 3$  order spline is considered. In the simulation the extended target moves from right to left throughout the surveillance area. The orientation of the target is  $0$ rad and the length and width are  $l = 5$ m and  $w = 2.5$ m respectively. The target trajectory and the surveillance region of the sensor located at the origin are illustrated in Figure 2. As seen from the figure, the target starts far from the sensor and hence has few measurements initially. Neither width nor the length of the target is observable. As the target travels towards the sensor, the length and the width observability increases first

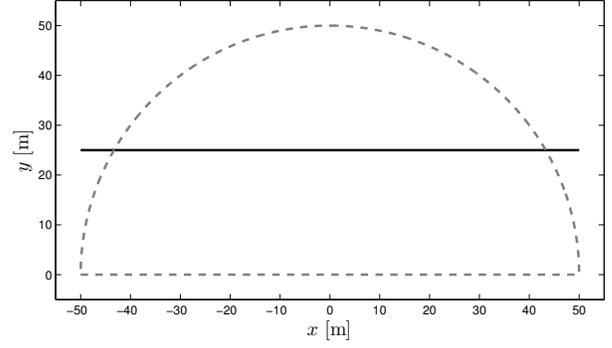


Figure 2. The trajectory used in the simulation. The target trajectory is showed in black. The sensor is located in the origin, the surveillance area boundary is showed with a dashed gray line.

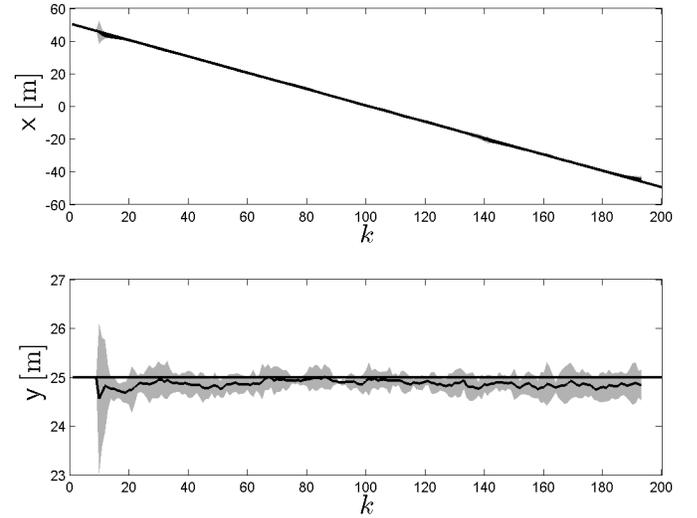


Figure 3. The position estimates ( $\mathbf{x}$  and  $\mathbf{y}$ ) of the algorithm. The true quantities are shown with grey lines. The mean  $\mathbf{x}$  and  $\mathbf{y}$  estimates are in black lines and their 4 standard deviation uncertainty calculated from the particles are illustrated with grey clouds around the estimates.

but when the target starts to get close to the sensor, the width observability is lost again in which case only the length of the target is visible. When the target passes by the sensor the width becomes once more observable. With the distance between the target and the sensor increasing towards the end of the scenario, observability of both quantities decrease.

The position estimation results are shown in Figure 3. Clearly the algorithm can follow the target along the  $\mathbf{x}$ -axis. The shape estimates are illustrated in Figure 4. As predicted the shape estimates degrade as the target gets further from the sensor. Though the variance of the estimates are large at times, the algorithm seems to be capable of keeping them at a smaller level than the initial values. As the target approaches the sensor, the general trend in the variances is to decrease though there exist also occasional increases. Further investigations must be done with various initializations and stability and the robustness must be evaluated more thoroughly in the future

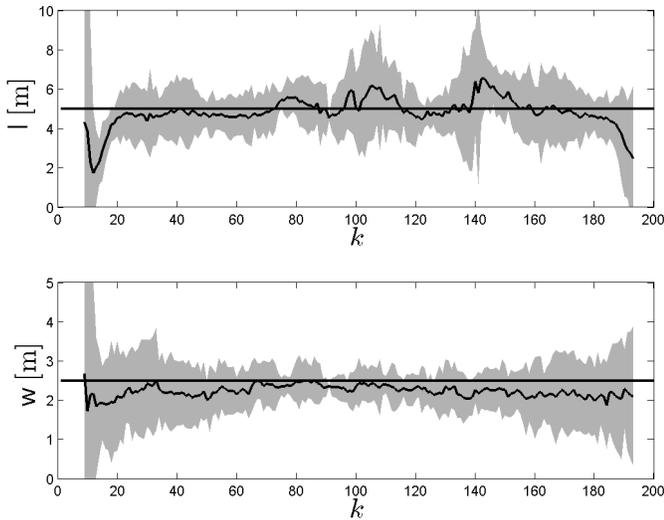


Figure 4. The length and the width estimates of the algorithm. The true quantities are shown with grey lines. The mean length and width estimates are in black lines and their 4 standard deviation uncertainty calculated from the particles are illustrated with grey clouds around the estimates.

work.

## VI. CONCLUSIONS

In this work a new approach to track extended targets has been presented. Point measurements are produced by an unknown number of measurement generating points (MGP:s) on the surface of the target. These MGP:s are modeled as a random finite set RFS and used as a means to estimate the shape, the size and the position of a target. The state of the target is propagated with Rao-Blackwellized particle filter (nonlinear part) and Kalman filters (linear part); the measurements are processed with a GM-PHD-filter.

First simulation results show that this approach is promising. A simple rectangular target is followed with a rather stable performance and it remains for future work to estimate the shape of more complex targets. The next step will be to validate the proposed method on more challenging real data collected on e.g., a road, where different type of shapes, such as pedestrians and vehicles are visible. The stability and robustness must also be evaluated thoroughly with different initializations and settings.

## ACKNOWLEDGEMENT

This work has been supported by the Swedish Research Council under the Linnaeus Center CADICS.

## REFERENCES

- [1] R. Mahler, "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.
- [2] —, *Statistical Multisource-Multitarget Information Fusion*. Norwood, MA, USA: Artech House, 2007.
- [3] 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.
- [4] K. Gilholm, S. Godsill, S. Maskell, and D. Salmund, "Poisson models for extended target and group tracking," in *Proceedings of Signal and Data Processing of Small Targets*, vol. 5913. San Diego, CA, USA: SPIE, Aug. 2005, pp. 230–241.
- [5] R. Mahler, "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] K. Granström, C. Lundquist, and U. Orguner, "A Gaussian mixture PHD filter for extended target tracking," in *Proceedings of the International Conference on Information Fusion*, Edinburgh, UK, Jul. 2010.
- [7] J. Mullane, B.-N. Vo, M. D. Adams, and B.-T. Vo, "A random-finite-set approach to bayesian slam," *IEEE Transactions on Robotics*, vol. PP, no. 99, pp. 1–15, Feb. 2011.
- [8] T. B. Schön, F. Gustafsson, and P.-J. Nordlund, "Marginalized particle filters for mixed linear/nonlinear state-space models," *IEEE Transactions on Signal Processing*, vol. 53, no. 7, pp. 2279–2289, Jul. 2005.
- [9] A. Blake and M. Isard, *Active contours*. London, UK: Springer, 1998.
- [10] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *International Journal of Computer Vision*, vol. 1, no. 4, pp. 321–331, Jan. 1988.
- [11] A. Srivastava and I. H. Jermyn, "Looking for shapes in two-dimensional cluttered point clouds," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, pp. 1616–1629, 2009.
- [12] M. Isard and A. Blake, "CONDENSATION—conditional density propagation for visual tracking," *International Journal of Computer Vision*, vol. 29, no. 1, pp. 5–28, Aug. 1998.
- [13] S. Lazarus, A. Tsourdos, B. White, P. Silson, and R. Zandbikowski, "Airborne vehicle mapping of curvilinear objects using 2-D splinegon," *IEEE Transactions on Instrumentation and Measurement*, vol. 59, no. 7, pp. 1941–1954, Jul. 2010.
- [14] K. Gilholm and D. Salmund, "Spatial distribution model for tracking extended objects," *IEE Proceedings of Radar, Sonar and Navigation*, vol. 152, no. 5, pp. 364–371, Oct. 2005.
- [15] Y. Boers, H. Driessen, J. Torstensson, M. Trieb, R. Karlsson, and F. Gustafsson, "A track before detect algorithm for tracking extended targets," *IEE Proceedings Radar, Sonar and Navigation*, vol. 153, no. 4, pp. 345–351, Aug. 2006.
- [16] J. Vermaak, N. Ikoma, and S. J. Godsill, "Sequential Monte Carlo framework for extended object tracking," *IEE Proceedings of Radar, Sonar and Navigation*, vol. 152, no. 5, pp. 353–363, Oct. 2005.
- [17] D. Angelova and L. Mihaylova, "Extended object tracking using Monte Carlo methods," *IEEE Transactions on Signal Processing*, vol. 56, no. 2, pp. 825–832, Feb. 2008.
- [18] 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.