Extended Target Tracking using a Gaussian-Mixture PHD filter

Karl Granström, Christian Lundquist, Umut Orguner

Division of Automatic Control
E-mail: karl@isy.liu.se, lundquist@isy.liu.se, umut@isy.liu.se

17th October 2011

Report no.: LiTH-ISY-R-3028
Submitted to IEEE Transactions on Aerospace and Electronic Systems

Address:
Department of Electrical Engineering
Linköpings universitet
SE-581 83 Linköping, Sweden

WWW: http://www.control.isy.liu.se
Abstract
This paper presents a Gaussian-mixture implementation of the PHD filter for tracking extended targets. The exact filter requires processing of all possible measurement set partitions, which is generally infeasible to implement. A method is proposed for limiting the number of considered partitions and possible alternatives are discussed. The implementation is used on simulated data and in experiments with real laser data, and the advantage of the filter is illustrated. Suitable remedies are given to handle spatially close targets and target occlusion.

Keywords: Target tracking, extended target, PHD filter, random set, Gaussian-mixture, laser sensor.
Extended Target Tracking using a Gaussian-Mixture PHD filter

Karl Granström, Member, IEEE, Christian Lundquist, and Umut Orguner, Member, IEEE

Abstract—This paper presents a Gaussian-mixture implementation of the PHD filter for tracking extended targets. The exact filter requires processing of all possible measurement set partitions, which is generally infeasible to implement. A method is proposed for limiting the number of considered partitions and possible alternatives are discussed. The implementation is used on simulated data and in experiments with real laser data, and the advantage of the filter is illustrated. Suitable remedies are given to handle spatially close targets and target occlusion.

Index Terms—Target tracking, extended target, PHD filter, random set, Gaussian-mixture, laser sensor.

I. INTRODUCTION

In most multi-target tracking applications it is assumed that each target produces at most one measurement per time step. This is true for cases when the distance between the target and the sensor is large in comparison to the target’s size. In other cases however, the target size may be such that multiple resolution cells of the sensor are occupied by the target. Targets that potentially give rise to more than one measurement per time step are categorized as extended. Examples include the cases when vehicles use radar sensors to track other road-users, when ground radar stations track airplanes which are sufficiently close to the sensor, or in mobile robotics when pedestrians are tracked using laser range sensors.

Gilholm and Salmond [1] have presented an approach for tracking extended targets under the assumption that the number of received target measurements in each time step is Poisson distributed. Their algorithm was illustrated with two examples where point targets which may generate more than one measurement per time step are categorized extended. Examples include the cases when vehicles use radar sensors to track other road-users, when ground radar stations track airplanes which are sufficiently close to the sensor, or in mobile robotics when pedestrians are tracked using laser range sensors.

Using the rigorous finite set statistics (FISST), Mahler has pioneered the recent advances in the field of multiple target tracking with a set theoretic approach where the targets and measurements are treated as random finite sets (RFS). This type of approach allows the problem of estimating multiple targets in clutter and uncertain associations to be cast in a Bayesian filtering framework [4], which in turn results in an optimal multi-target Bayesian filter. As is the case in many nonlinear Bayesian estimation problems, the optimal multi-target Bayesian filter is infeasible to implement except for simple examples and an important contribution of FISST is to provide structured tools in the form of the statistical moments of a RFS. The first order moment of a RFS is called probability hypothesis density (PHD), and it is an intensity function defined over the state space of the targets. The so called PHD filter [4], [5] propagates in time PHDs corresponding to the set of target states as an approximation of the optimal multi-target Bayesian filter. A practical implementation of the PHD filter is provided by approximating the PHDs with Gaussian-mixtures (GM) [6] which results in the Gaussian-mixture PHD (GM-PHD) filter. In the recent work [7], Mahler presented an extension of the PHD filter to also handle extended targets of the type presented in [2].

In this paper, we present a Gaussian-mixture implementation of the PHD-filter for extended targets [7], which we call the extended target GM-PHD-filter (ET-GM-PHD). In this way, we, to some extent, give a practical extension of the series of work in [2], [6], [7]. An earlier version of this work was presented in [8] and the current, significantly improved, version includes also practical examples with real data. For space considerations, we do not repeat the derivation of the PHD-filter equations for extended targets and instead refer the reader to [7].

The document is outlined as follows. The multiple extended target tracking problem is defined in Section II. The details of the Gaussian-mixture implementation are given in Section III. For the measurement update step of the ET-GM-PHD-filter, different partitions of the set of measurements have to be considered. A measurement clustering algorithm used to reduce the combinatorially exploding number of possible measurement partitions is described in Section IV. The proposed approaches are evaluated using both simulations and experiments. The target tracking setups for these evaluations are described in Section V, the simulation results are presented in Section VI and results using data from a laser sensor are presented in Section VII. Finally, Section VIII contains conclusions and thoughts on future work.
II. TARGET TRACKING PROBLEM FORMULATION

In previous work, extended targets have often been modeled as targets having a spatial extension or shape that would lead to multiple measurements, as opposed to at most a single measurement. On the other hand, the extended target tracking problem can be simplified by the assumption that the measurements originating from a target are distributed approximately around a target reference point [1] which can be e.g. the centroid or any other point depending on the extent (or the shape) of the target. Though all targets obviously have a spatial extension and shape, in the latter type of modeling, only the target reference point is important and the target extent does not need to be estimated.

The relevant target characteristics that are to be estimated form the target’s state vector \( x \). Generally, beside the kinematic variables as position, velocity and orientation, the state vector may also contain information about the target’s spatial extension. As mentioned above, when the target’s state does not contain any variables related to the target extent, though the estimation is done as if the target was a point (i.e. the target reference point), the algorithms should still take care of the multiple measurements that originate from a target. Hence, in this study, we use a generalized definition of an extended target, given below, which does not depend on whether the target extent is estimated or not.

**Definition 1** (Extended Target). A target which potentially gives rise to more than one measurement per time step irrespective of whether the target’s extent is explicitly modeled (and/or estimated) or not.

In this work, to simplify the presentation, no information about the size and shape of the target is kept in the state vector \( x \), i.e. the target extent is not explicitly estimated. Nevertheless, it must be emphasized that this causes no loss of generality as shown by the recent work [9] where the resulting ET-GM-PHD filter is used to handle the joint estimation of size, shape and kinematic variables for rectangular and elliptical extended targets. We model both the target states to be estimated, and the measurements collected, as RFSs. The motivation behind this selection is two-fold. First, in many practical systems, although the sensor reports come with a specific measurement order, the results of the target tracking algorithms are invariant under the permutations of this order. Hence, modeling the measurements as the elements of a set in which the order of the elements is irrelevant makes sense. Second, this work unavoidably depends on the previous line of work [7], which is based on such a selection.

The initial GM-PHD work [6] does not provide tools for ensuring track continuity, for which some remedies are described in the literature, see e.g. [10]. It has however been shown that labels for the Gaussian components can be included into the filter in order to obtain individual target tracks, see e.g. [11]. In this work, for the sake of simplicity, labels are not used, however they can be incorporated as in [11] to provide track continuity.

We denote the unknown number of targets as \( N_{x,k} \), and the set of target states to be estimated at time \( k \) is \( X_k = \{ x^{(i)}_k \}_{i=1}^{N_{x,k}} \). The measurement set obtained at time \( k \) is \( Z_k = \{ z^{(i)}_k \}_{i=1}^{N_{z,k}} \) where \( N_{z,k} \) is the number of the measurements. The dynamic evolution of each target state \( x^{(i)}_k \) in the RFS \( X_k \) is modeled using a linear Gaussian dynamical model,

\[
x^{(i)}_{k+1} = F_k x^{(i)}_k + G_k w^{(i)}_k,
\]

for \( i = 1, \ldots, N_{x,k} \), where \( w^{(i)}_k \) is Gaussian white noise with covariance \( Q^{(i)} \). Note that each target state evolves according to the same dynamic model independent of the other targets.

The number of measurements generated by the \( i \)th target at each time step is a Poisson distributed random variable with rate \( \gamma(x^{(i)}_k) \) measurements per scan, where \( \gamma(\cdot) \) is a known non-negative function defined over the target state space. The probability of the \( i \)th target generating at least one measurement is then given as

\[
1 - e^{-\gamma(x^{(i)}_k)}.
\]

The \( i \)th target is detected with probability \( p_D(x^{(i)}_k) \) where \( p_D(\cdot) \) is a known non-negative function defined over the target state space. This gives the effective probability of detection

\[
1 - e^{-\gamma(x^{(i)}_k)} p_D(x^{(i)}_k).
\]

The measurements originating from the \( i \)th target are assumed to be related to the target state according to a linear Gaussian model given as

\[
z^{(i)}_k = H x^{(i)}_k + e^{(i)}_k,
\]

where \( e^{(i)}_k \) is white Gaussian noise with covariance \( R_k \). Each target is assumed to give rise to measurements independently of the other targets. We here emphasize, that in an RFS framework both the set of measurements \( Z_k \) and the set of target states \( X_k \) are unlabeled, and hence no assumptions are made regarding which target gives rise to which measurement.

The number of clutter measurements generated at each time step is a Poisson distributed random variable with rate \( \beta_{FA,k} \) clutter measurements per surveillance volume per scan. Thus, if the surveillance volume is \( V_s \), the mean number of clutter measurements is \( \beta_{FA,k} V_s \) clutter measurements per scan. The spatial distribution of the clutter measurements is assumed uniform over the surveillance volume.

The aim is now to obtain an estimate of the sets of the target states \( \hat{X}^K = \{ \hat{X}_k \}^K_{k=1} \), the sets of measurements \( \hat{Z}^K = \{ \hat{Z}_k \}^K_{k=1} \), given the sets of measurements \( Z^K = \{ Z_k \}^K_{k=1} \). We achieve this by propagating the predicted and updated PHDs, denoted \( D_{k|k-1}(\cdot) \) and \( D_{k|k}(\cdot) \), respectively, of the set of target states \( X_k \), using the PHD filter presented in [7].

III. GAUSSIAN-MIXTURE IMPLEMENTATION

In this section, following the derivation of the GM-PHD filter for standard single measurement targets in [6], a PHD recursion for the extended target case is described. Since the prediction update equations of the extended target PHD filter are the same as those of the standard PHD filter [7], the Gaussian mixture prediction update equations of the ET-GM-PHD filter are the same as those of the standard GM-PHD filter
Assumption 3. The predicted multi-target RFS is Poisson. □

Assumption 4. Each target follows a linear Gaussian dynamical model, cf. (1), and the sensor has a linear Gaussian measurement model, cf. (4). □

Assumption 5. The survival and detection probabilities are state independent, i.e. \( p_S(x) = p_S \) and \( p_D(x) = p_D \). □

Assumption 6. The intensities of the birth and spawn RFS are Gaussian-mixtures. □

In this paper we adopt all of the above assumptions except that we relax the assumption on detection probability as follows.

Assumption 7. The following approximation about probability of detection function \( p_D(\cdot) \) holds.

\[
p_D(x) \mathcal{N}(\mu; m_{k|k-1}^{(j)}, P_{k|k-1}^{(j)}) \approx p_D(m_{k|k-1}^{(j)}) \mathcal{N}(\mu; m_{k|k-1}^{(j)}, P_{k|k-1}^{(j)})
\]

for all \( x \) and for \( j = 1, \ldots, J_{k|k-1} \). □

Assumption 7 is weaker than Assumption 5 in that it is trivially satisfied when \( p_D(\cdot) = p_D \), i.e. when \( p_D(\cdot) \) is constant. In general, Assumption 7 holds approximately when the function \( p_D(\cdot) \) does not vary much in the uncertainty zone of a target determined by the covariance \( P_{k|k-1}^{(j)} \). This is true either when \( p_D(\cdot) \) is a sufficiently smooth function or when the signal to noise ratio (SNR) is high enough such that \( P_{k|k-1}^{(j)} \) is sufficiently small. We still note here that Assumption 7 is only for the sake of simplification rather than approximation, since \( p_D(x) \) can always be approximated as a mixture of exponentials of quadratic functions (or equivalently as Gaussians) without losing the Gaussian-mixture structure of the corrected PHD, see [6]. This, however, would cause a multiplicative increase in the number of components in the corrected PHD, which would in turn make the algorithm need more aggressive pruning and merging operations. A similar approach to variable probability of detection has been taken in order to model the clutter notch in ground moving target indicator tracking [12].

For the expected number of measurements from the targets, represented by \( \gamma(\cdot) \), similar remarks apply and we use the following assumption.

Assumption 8. The following approximation about \( \gamma(\cdot) \) holds

\[
e^{-\gamma(x)} \mathcal{N}(\mu; m_{k|k-1}^{(j)}, P_{k|k-1}^{(j)}) \approx e^{-\gamma(m_{k|k-1}^{(j)})} \mathcal{N}(\mu; m_{k|k-1}^{(j)}, P_{k|k-1}^{(j)})
\]

for all \( x, j = 1, \ldots, J_{k|k-1} \) and \( n = 1, 2, \ldots \). □
to noise ratio (SNR) is high enough such that \( P_{k|k-1}^{(j)} \) is sufficiently small.

With the assumptions presented above, the posterior intensity at time \( k \) is a Gaussian-mixture given by

\[
D_{k|k}(x) = D_{k|k}^{ND}(x) + \sum_{p \in Z_k} \sum_{W \in p} D_{k|k}^{D}(x, W).
\]

(10)

The Gaussian-mixture \( D_{k|k}^{ND}(\cdot) \), handling the no detection cases, is given by

\[
D_{k|k}^{ND}(x) = \sum_{j=1}^{J_{k|k-1}} \omega_{k|j}^{(j)} N\left( x ; m_{k|j}^{(j)}, P_{k|j}^{(j)} \right),
\]

(11a)

\[
w_{k|j}^{(j)} = \left( 1 - (1 - e^{-\gamma_{(j)}}) \right) \pi_{D}^{(j)} w_{k|j-1}^{(j)},
\]

(11b)

\[
m_{k|j}^{(j)} = m_{k|j-1}^{(j)}; \quad P_{k|j}^{(j)} = P_{k|j-1}^{(j)}.
\]

(11c)

where the product is over all measurements \( z_k \) in the cell \( W \) and \(|W|\) is the number of elements in \( W \). The coefficient \( \omega_{k|j}^{(j)} \) is given by

\[
\omega_{k|j}^{(j)} = \pi_{D}^{(j)} \prod_{z_k \in W} \frac{1}{C_k(z_k)},
\]

(12d)

and is calculated using

\[
z_W \triangleq \bigoplus_{z_k \in W} z_k;
\]

\[
H_W = \begin{bmatrix} H_k^T, H_k^T, \cdots, H_k^T \end{bmatrix}_T,
\]

\[
R_W = \text{blkdiag}(R_k, R_k, \cdots, R_k).
\]

The operation \( \bigoplus \) denotes vertical vectorial concatenation. The partition weights \( \omega_p \) can be interpreted as the probability of the partition \( p \) being true and are calculated as

\[
\omega_p = \frac{\prod_{W \in p} d_W}{\sum_{p'} \prod_{z_k \in W} d_{W',}^{p'}},
\]

(12f)

\[
d_W = \delta_{|W|,1} + \sum_{j=1}^{J_{k|k-1}} \Gamma^{(j)} \pi_{D}^{(j)} \prod_{z_k \in W} \frac{1}{C_k(z_k)},
\]

(12g)

where \( \delta_{i,j} \) is the Kronecker delta. The mean and covariance of the Gaussian components are updated using the standard Kalman measurement update,

\[
m_{k|j}^{(j)} = m_{k|k-1}^{(j)} + K_k^{(j)} (z_W - H_W m_{k|k-1}^{(j)}),
\]

(13a)

\[
P_{k|j}^{(j)} = \left( I - K_k^{(j)} H_W \right) P_{k|k-1}^{(j)}.
\]

(13b)

\[
K_k^{(j)} = P_{k|k-1}^{(j)} \left( H_W F_k^{(j)} H_W^T + R_W \right)^{-1}.
\]

(13c)

In order to keep the number of Gaussian components at a computationally tractable level, pruning and merging is performed as in [6].

### IV. Partitioning the Measurement Set

As observed in the previous section, an integral part of extended target tracking with the PHD filter is the partitioning of the set of measurements [7]. The partitioning is important, since more than one measurement can stem from the same target. Let us exemplify the process of partitioning with a measurement set containing three individual measurements, \( Z_k = \{ z_k^{(1)}, z_k^{(2)}, z_k^{(3)} \} \). This set can be partitioned in the following different ways:

\[
p_1 : W_1^1 = \{ z_k^{(1)}, z_k^{(2)}, z_k^{(3)} \},
\]

\[
p_2 : W_2^1 = \{ z_k^{(1)}, z_k^{(2)} \}, \quad W_2^2 = \{ z_k^{(3)} \},
\]

\[
p_3 : W_3^1 = \{ z_k^{(1)}, z_k^{(3)} \}, \quad W_3^2 = \{ z_k^{(2)} \},
\]

\[
p_4 : W_4^1 = \{ z_k^{(2)}, z_k^{(3)} \}, \quad W_4^2 = \{ z_k^{(1)} \},
\]

\[
p_5 : W_5^1 = \{ z_k^{(1)} \}, \quad W_5^2 = \{ z_k^{(2)} \}, \quad W_5^3 = \{ z_k^{(3)} \}.
\]

(13c)

Here, \( p_i \) is the \( i \)-th partition, and \( W_i^j \) is the \( j \)-th cell of partition \( i \). Let \( |p_i| \) denote the number of cells in the partition, and let \(|W_i^j|\) denote the number of measurements in the cell. It is quickly realized that as the size of the measurement set increases, the number of possible partitions grows very large. In order to have a computationally tractable target tracking method, only a subset of all possible partitions can be considered. In order to achieve good extended target tracking results, this subset of partitions must represent the most likely ones of all possible partitions.

In Section IV-A, we propose a simple heuristic for finding this subset of partitions, which is based on the distances between the measurements. We here note that our proposed method is only one instance of a vast number of other clustering algorithms found in the literature, and that other methods could have been used. Some well-known alternatives are pointed out, and compared to the proposed partitioning method, in Section IV-B. A modification of the partitioning approach to better handle targets which are spatially close is described in Section IV-C.

#### A. Distance Partitioning

Consider a set of measurements \( Z = \{ z^{(i)} \}_{i=1}^{N} \). Our partitioning algorithm relies on the following theorem.

**Theorem 1.** Let \( d(\cdot, \cdot) \) be a distance measure and \( d_{k} \geq 0 \) be an arbitrary distance threshold. Then there is one and only

\[\text{This example was also utilized in [7].}\]
one partition in which any pair of measurements \(z^{(i)}, z^{(j)} \in \mathbf{Z}\) that satisfy
\[
\mathbf{d} \left( z^{(i)}, z^{(j)} \right) \leq d_{\ell}
\]
are in the same cell.

\[\Delta_{ij} \triangleq \mathbf{d}(z^{(i)}, z^{(j)}), \text{ for } 1 \leq i \neq j \leq N_z.\] (16)

Theorem 1 says that there is a unique partition that leaves all pairs \((i, j)\) of measurements satisfying \(\Delta_{ij} \leq d_{\ell}\) in the same cell. We give an example algorithm that can be used to obtain this unique partition in Table I. We use this algorithm to generate \(N_d\) alternative partitions of the measurement set \(\mathbf{Z}\), by selecting \(N_d\) different thresholds
\[
\{d_{\ell}\}_{\ell=1}^{N_d}, \quad d_{\ell} < d_{\ell+1}, \text{ for } \ell = 1, \ldots, N_d - 1.\] (17)

The alternative partitions contain fewer cells as the \(d_{\ell}\)'s are increasing, and the cells typically contain more measurements.

The thresholds \(\{d_{\ell}\}_{\ell=1}^{N_d}\) are selected from the set
\[
\mathcal{D} \triangleq \{0\} \cup \{\Delta_{ij} | 1 \leq i < j \leq N_z\}
\]
(18)

where the elements of \(\mathcal{D}\) are sorted in ascending order. If one uses all of the elements in \(\mathcal{D}\) to form alternative partitions, \(|\mathcal{D}| = N_z(N_z - 1)/2 + 1\) partitions are obtained. Some partitions resulting from this selection might still turn out to be identical, and must hence be discarded so that each partition at the end is unique. Among these unique partitions, the first (corresponding to the threshold \(d_1 = 0\)) would contain \(N_z\) cells with one measurement each. The last partition would have just one cell containing all \(N_z\) measurements. Notice that this partitioning methodology already reduces the number of partitions tremendously.

The smallest and the largest thresholds in the set \(\mathcal{D}\) on the other hand can still contain very similar values due to the fact that the measurements are generally clustered around the targets. In order to further reduce the computational load, partitions in this work are computed only for a subset of thresholds in the set \(\mathcal{D}\). This subset is determined based on the statistical properties of the distances between the measurements belonging to the same target.

Suppose we select the distance measure \(\mathbf{d}(\cdot, \cdot)\) as the Mahalanobis distance, given by
\[
\mathbf{d}_M \left( z_{k}^{(i)}, z_{k}^{(j)} \right) = \sqrt{\left( z_{k}^{(i)} - z_{k}^{(j)} \right)^T R_k^{-1} \left( z_{k}^{(i)} - z_{k}^{(j)} \right)}.
\] (19)

Then, for two target-originated measurements \(z_{k}^{(i)}\) and \(z_{k}^{(j)}\) belonging to the same target, \(\mathbf{d}_M \left( z_{k}^{(i)}, z_{k}^{(j)} \right)\) is \(\chi^2\) distributed with degrees of freedom equal to the measurement vector dimension. Using the inverse cumulative \(\chi^2\) distribution function, which we denote here as \text{invchi2}(.), a unitless distance threshold \(\delta_{P_G}\) can be computed as
\[
\delta_{P_G} = \text{invchi2}(P_G),
\] (20)

for a given probability \(P_G\). We have seen in early empirical simulations that good target tracking results are achieved with partitions computed using the subset of distance thresholds in \(\mathcal{D}\) satisfying the condition \(\delta_{P_L} < d_{\ell} < \delta_{P_U}\) for lower probabilities \(P_L \leq 0.3\) and upper probabilities \(P_U \geq 0.8\).

As a simple example, if there are four targets present, each with expected number of measurements 20, and clutter measurements are generated with \(\beta_{FA}V_s = 50\), then the mean number of measurements collected each time step would be 130. For 130 measurements, the number of all possible partitions is given by the Bell number \(B_{130} \propto 10^{161}\) [13]. Using all of the thresholds in the set \(\mathcal{D}\), 130 different partitions would be computed on average. Using the upper and lower probabilities \(P_L = 0.3\) and \(P_U = 0.8\), Monte Carlo simulations show that on average only 27 partitions are computed, representing a reduction of computational complexity several orders of magnitude.

\[\begin{array}{ll}
\text{TABLE I} \\
\text{DISTANCE PARTITIONING}
\end{array}\]

<table>
<thead>
<tr>
<th>Require:</th>
<th>(d_{\ell}, \Delta_{ij}, 1 \leq \ell \neq j \leq N_z).</th>
</tr>
</thead>
<tbody>
<tr>
<td>1: \text{CellNumber}(i) = 0, 1 \leq \ell \leq N_z \text{ (Set cells of all measurements to null)}</td>
<td></td>
</tr>
<tr>
<td>2: \text{CellId} = 1 \text{ (Set the current cell id to 1)}</td>
<td></td>
</tr>
<tr>
<td>for (i = 1 : N_z) do</td>
<td></td>
</tr>
<tr>
<td>3: if \text{CellNumbers}(i) = 0 \text{ then}</td>
<td></td>
</tr>
<tr>
<td>4: \text{CellNumbers}(i) = \text{CellId}</td>
<td></td>
</tr>
<tr>
<td>5: \text{CellNumbers}(i) = \text{CellId}</td>
<td></td>
</tr>
<tr>
<td>6: \text{CellNumbers} = \text{FindNeighbors}(i, \text{CellNumbers}, \text{CellId})</td>
<td></td>
</tr>
<tr>
<td>7: \text{CellId} = \text{CellId}+1</td>
<td></td>
</tr>
<tr>
<td>end if</td>
<td></td>
</tr>
<tr>
<td>end for</td>
<td></td>
</tr>
</tbody>
</table>

The recursive function \text{FindNeighbors}(\cdot, \cdot, \cdot) is given as
\[\begin{array}{ll}
\text{1: function CellNumbers = FindNeighbors(i, CellNumbers, CellId)} |
| 2: \text{for } j = 1 : N_z \text{ do} |
| 3: if \text{j} \neq i \& \text{CellNumbers}(j) = 0 \text{ then} |
| 4: \text{CellNumbers}(j) = \text{CellId} |
| 5: \text{CellNumbers} = \text{FindNeighbors}(j, \text{CellNumbers}, \text{CellId}) |
| 6: \text{end if} |
| 7: \text{end for} |

Another major difference between the suggested distance partitioning and \(K\)-means clustering is highlighted in Fig. 1, which shows a measurement set that consists of \(N_{z,k} = 13\)
measurements, 10 of which are clustered in the northeast of the surveillance region and the other three are scattered individually. The intuitive way to cluster this set of measurements is into 4 clusters, which is achieved by Distance Partitioning using a distance threshold of about 25 m, as shown in the left plot of Fig. 1. When there is a large number of measurements concentrated in one part of the surveillance area, as is the case in this example, K-means clustering tends to split those measurements into smaller cells, and merge remaining but far away measurements into large cells which is illustrated in the right plot of Fig. 1.

The main reason behind this shortcoming of K-means is the initialization of the algorithm, where the initial cluster centers are chosen by uniform sampling. In order to overcome this problem, modifications to the standard K-means algorithm have been suggested, where initial clusters are chosen as separated as possible, see [16], [17]. This improved version of K-means is called K-means++.

In simulations, Distance Partitioning was compared to K-means (using MATLAB’s kmeans) and K-means++ (using an implementation available online [18]). The results show that K-means++ in fact outperforms K-means, however it still fails to compute the correct partitions much too often, except in scenarios with very low $\beta_{FA,k}$. This can be attributed to the existence of counter-intuitive local optima for the implicit cost function involved with K-means (or K-means++). Distance Partitioning on the other hand can handle both high and low $\beta_{FA,k}$ and always gives an intuitive and unique partitioning for a given $d_k$.

Therefore, we argue that a hierarchical method, such as the suggested distance partitioning, should be preferred over methods such as K-means. However, it is important to note here again, that regarding partitioning of the measurement set, the contribution of the current work lies mainly not in the specific method that is suggested, but more in showing that all possible partitions can efficiently be approximated using a subset of partitions.

### C. Sub-Partitioning

Initial results using ET-GM-PHD showed problems with underestimation of target set cardinality in situations where two or more extended targets are spatially close [8]. The reason for this is that when targets are spatially close, so are their resulting measurements. Thus, using Distance Partitioning, measurements from more than one measurement source will be included in the same cell $W$ in all partitions $p$, and subsequently the ET-GM-PHD filter will interpret measurements from multiple targets as having originated from just one target. In an ideal situation, where one could consider all possible partitions of the measurement set, there would be alternative partitions which would contain the subsets of a wrongly merged cell. Such alternative partitions would dominate the output of the ET-GM-PHD filter towards the correct estimated number of targets. Since we eliminate such partitions completely using Distance Partitioning, the ET-GM-PHD filter lacks the means to correct its estimated number of targets.

One remedy for this problem is to form additional sub-partitions after performing Distance Partitioning, and to add them to the list of partitions that ET-GM-PHD filter considers at the current time step. Obviously, this should be done only when there is a risk of having merged the measurements belonging to more than one target, which can be decided based on the expected number of measurements originating from a target. We propose the following procedure for the addition of the sub-partitions.

Suppose that we have computed a set of partitions using Distance Partitioning, cf. the algorithm in Table I. Then, for each partition generated by the Distance Partitioning, say for $p_i$, we calculate the maximum likelihood (ML) estimates $\hat{N}^j_w$ of the number of targets for each cell $W^j_w$. If this estimate is larger than one, we split the cell $W^j_w$ into $\hat{N}^j_w$ sub-cells, denoted as

$$\{W^+_s\}_{s=1}^\hat{N}^j_w. \quad (21)$$

We then add a new partition, consisting of the new sub-cells along with the other cells in $p_i$, to the list of partitions obtained by the Distance Partitioning.

We illustrate the sub-partitioning algorithm in Table II, where the splitting operation on a cell is shown by a function

$$\text{split} \left( \hat{N}^j_w, W^j_w \right). \quad (22)$$

We give the details for obtaining the ML estimate $\hat{N}^j_w$ and choosing the function $\text{split} \left( \cdot, \cdot \right)$ in the subsections below.

1) Computing $\hat{N}^j_w$: For this operation, we assume that the function $\gamma(\cdot)$ determining the expected number measurements generated by a target is constant i.e., $\gamma(\cdot) = \gamma$. Each target generates measurements independently of the other targets, and the number of generated measurements by each target is distributed with the Poisson distribution, $\text{Pois}(\cdot, \gamma)$. The likelihood function for the number of targets corresponding to a cell $W^j_w$ is given as

$$p \left( |W^j_w|, N^j_w = n \right) = \text{Pois}(|W^j_w|, \gamma n). \quad (23)$$

Here, we assumed that the volume covered by a cell is sufficiently small such that the number of false alarms in the

<table>
<thead>
<tr>
<th>TABLE II</th>
<th>SUB-PARTITION</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Require:</strong> Partitioned set of measurements $Z^p = {p_1, \ldots, p_{N_p}}$, where $N_p$ is the number of partitions.</td>
<td></td>
</tr>
<tr>
<td><strong>1:</strong> <strong>Initialise:</strong> Counter for new partitions $\ell = N_p$.</td>
<td></td>
</tr>
<tr>
<td><strong>2:</strong> <strong>for</strong> $i = 1, \ldots, N_p$ <strong>do</strong></td>
<td></td>
</tr>
<tr>
<td><strong>3:</strong> <strong>for</strong> $j = 1, \ldots,</td>
<td>p_i</td>
</tr>
<tr>
<td><strong>4:</strong> $\hat{N}^j_w = \arg\max_n p \left(</td>
<td>W^j_w</td>
</tr>
<tr>
<td><strong>5:</strong> <strong>if</strong> $\hat{N}^j_w &gt; 1$ <strong>then</strong></td>
<td></td>
</tr>
<tr>
<td><strong>6:</strong> $\ell = \ell + 1$ {Increase the partition counter}</td>
<td></td>
</tr>
<tr>
<td><strong>7:</strong> $p_\ell = p_\ell \cup W^j_w$ {Current partition except the current cell}</td>
<td></td>
</tr>
<tr>
<td><strong>8:</strong> ${W^+<em>s}</em>{s=1}^{\hat{N}^j_w}$ {Split the current cell}</td>
<td></td>
</tr>
<tr>
<td><strong>9:</strong> $p_\ell = p_\ell \cup {W^+<em>s}</em>{s=1}^{\hat{N}^j_w}$ {Augment the current partition}</td>
<td></td>
</tr>
<tr>
<td><strong>10:</strong> <strong>end if</strong></td>
<td></td>
</tr>
<tr>
<td><strong>11:</strong> <strong>end for</strong></td>
<td></td>
</tr>
<tr>
<td><strong>12:</strong> <strong>end for</strong></td>
<td></td>
</tr>
</tbody>
</table>
cell is negligible, i.e. there are no false alarms in \( W_j \). The ML estimate \( \hat{N}_x^j \) can now be calculated as

\[
\hat{N}_x^j = \arg \max_n p \left( \left| W_j \right| \left| N_x^j = n \right. \right). \tag{24}
\]

Note that other alternatives can be found for calculating the estimates of \( N_x^j \), e.g. utilizing specific knowledge about the target tracking setup, however, both simulations and experiments have shown that the above suggested method works well.

2) The split \((\cdot, \cdot)\) function: An important part of the Sub-Partition function in Table II is the function split \((\cdot, \cdot)\), which is used to divide the measurements in a cell into smaller cells. In both simulations and experiments, we have used \( K\)-means clustering to split the measurements in the cell, results shows that this works well. Note however that other methods to split the measurements are possible.

**Remark 1** (Limitations of Sub-Partitioning). Notice that the Sub-Partition algorithm given in this section can be interpreted to be only a first-order remedy to the problem, and hence have limited correction capabilities. This is because we do not consider the combinations of the cells when we are adding sub-partitions. In the case, for example, where there are two pairs of close targets whose cells are merged wrongly by Distance Partitioning, the sub-partitioning algorithm presented above would add an additional partition for each of the target pairs (i.e. for each of the wrongly merged cells), but not an additional partition that contains the split versions of both cells. Consideration of all combinations of the wrongly merged cells seems again to be prohibitive, due to the combinatorial growth in the number of additional partitions. An idea for the cases where there can be more than one wrongly merged cells is to add a single additional partition, which contains split versions of all such cells.

**V. Target Tracking Setup**

The presented tracking approach is exemplified with a laser sensor tracking humans at short distance. In this section the tracking setup is defined for both a pure simulation environment and an experimental realisation with laser data. The targets are modeled as points with state variables

\[
x_k = \begin{bmatrix} x_k \\ y_k \\ v^x_k \\ v^y_k \end{bmatrix}^T,
\]

where \( x_k \) and \( y_k \) are the planar position coordinates of the target, and \( v^x_k \) and \( v^y_k \) are the corresponding velocities. The sensor measurements are given in batches of Cartesian \( x \) and \( y \) coordinates as follows:

\[
z_k^{(j)} = \begin{bmatrix} x_k^j \\ y_k^j \end{bmatrix}^T.
\]

A constant velocity model [19], with sampling time \( T \) is used. In all simulations the probability of detection and probability of survival are set as \( p_d = 0.99 \) and \( p_s = 0.99 \), respectively. The algorithm parameters for the simulation and experiment are given in Table III. The surveillance area is \([-1000, 1000] \times [-1000, 1000]\) for the simulations, and for the real data experiments the surveillance area is a semi circle located at the origin with range 13 m. In the simulations, clutter was generated with a Poisson rate of 10 clutter measurements per scan, and (unless otherwise stated) each target generated measurements with a Poisson rate of 10 measurements per scan. The birth intensity in the simulations is

\[
D_b(x) = 0.1\mathcal{N}(x; m_b, P_b) + 0.1\mathcal{N}(x; -m_b, P_b),
\]

\[
m_b = [250, 250, 0, 0]^T, \quad P_b = \text{diag}(\[100, 100, 25, 25]\)).
\]

For the experiments, the birth intensity Gaussian components are illustrated with their corresponding one standard deviation ellipsoids in Fig. 2. Each birth intensity component has a weight \( w_b^{(j)} = \frac{1}{J_b} \), where the number of components is \( J_b = 7 \). The spawn intensity is

\[
D_\beta(x|y) = w_\beta \mathcal{N}(x; \xi, Q_\beta),
\]

where \( \xi \) is the target from which the new target is spawned and the values for \( w_\beta \) and \( Q_\beta \) are given in Table III.

**VI. Simulation Results**

This section presents the results from the simulations using the presented extended target tracking method. In Section VI-A a comparison of the partitioning methods is presented, the results show the increased performance when using
Sub-Partition. A comparison between ET-GM-PHD and GM-
PMD is presented in Section VI-B, where it is shown that
ET-GM-PHD as expected outperforms GM-PHD for extended
targets. Section VI-C presents a comparison of ET-GM-PHD
and GM-PHD for targets that give rise to at most one measure-
ment per time step. Finally, detailed investigations are carried
out about the effects of the possibly unknown parameter \( \gamma \) in
Section VI-D.

A. Partitioning methods

As was noted in Section IV-C, as well as in previous
work [8], using only Distance Partitioning to obtain a subset
of all possible partitions is insufficient when the extended
targets are spatially close. For this reason, Sub-Partition
was introduced to obtain more partitions. In this section, we present
the results from simulations that compare the performance of
ET-GM-PHD tracking with partitions computed using only
Distance Partitioning and with partitions computed using
Distance Partitioning and Sub-Partition. Two scenarios are
considered, both with two targets. The true \( x,y \) positions
and the distance between the targets are shown in Fig. 3a
and Fig. 3b. At the closest points the targets were 60m and
50m apart, respectively. In the simulations, the targets were
modeled as points that generated measurements with standard
deviation of 20m in both \( x \) and \( y \) direction. Thus, a measure
of target extent can be taken as the two standard deviation
measurement covariance ellipses, which here are circles of
radius 40m. In both scenarios these circles partly overlap when
the targets are closest to each other.

Each scenario was simulated 100 times with a constant
expected number of measurements per target \( \gamma (\cdot) = \gamma \) of
5, 10 and 20, respectively. Fig. 4 shows the resulting sum
of weights of the ET-GM-PHD algorithm. As can be seen,
using Sub-Partition the average sum of weights is closer to
the true number of targets. This is especially clear for targets
that generate more measurements per time step, i.e. when \( \gamma \)
is higher.

B. Comparison with GM-PHD

This section presents results from a simulation comparison
of ET-GM-PHD and GM-PHD. Note here that the GM-
PHD filter is applied naively to the simulated measurements, i.e. it is
applied under the (false) assumption that each target produces
at most one measurement per time step. The true targets are
shown in black in Fig. 5a, around time 50–52 two target tracks
cross at a distance of just over 50m, at time 67 a new target is
spawned 20m from a previous one. Thus the scenario presents
challenges that are typical to multiple target applications.

To compare the ET-GM-PHD filter with the GM-PHD filter,
100 Monte Carlo simulations were performed, each with new
measurement noise and clutter measurements. The results are
shown in Fig. 5b and Fig. 5c, which show the corresponding
multi-target measure optimal subpattern assignment metric
(OSPA) [20], and the cardinality, respectively. In the OSPA
metric the parameters are set to \( p = 2 \), corresponding to
using the 2-norm which is a standard choice, and \( c = 60 \),
corresponding to a maximum error equal to three measurement
noise standard deviations. Here, the cardinality is computed
as \( \sum_{j=1}^{K} w_{jk}^{(j)} \). This sum can be rounded to obtain an integer
estimate of target number [6].

It is evident from the two figures that the presented ET-GM-
PMD significantly outperforms the standard GM-PHD, which
does not take into account the possibility of the multiple mea-
surements from single targets. The main difference between
the two filters is the estimation of cardinality, i.e. the number
of targets. The ET-GM-PHD filter correctly estimates the car-
dinality with the exception of when the new target is spawned
– after time 67 there is a small dip in the mean estimated
cardinality, even though Sub-Partition is used. The reason for
this is that the targets are 20m apart which is smaller than the
effective target extent determined by the measurement standard
deviation (20m). As in the previous section, assuming that
the effective target extent is two standard deviations, which
correspond to circular targets of 40m radius, at 20m distance
the measurements overlap significantly and the probability that
the new target’s measurements were also generated by the old
target, as computed in (12e), is large. As the targets separate,
this probability decreases and the ET-GM-PHD filter recovers
the correct cardinality. It should still be noted that, in reality,
where the targets would probably be rigid bodies, this type of
very close situation is highly unlikely and the results of the
ET-GM-PHD filter with Sub-Partition would be close to those
presented in Section VI-A.

Similar simulations have been performed which compare
Distance Partitioning with \( K \)-means clustering. Over 1000
Monte Carlo simulations, the tracking results in terms of
mean OSPA and mean cardinality are significantly better for
distance partitioning, mainly due to the problem with \( K \)-means
clustering shown in Fig. 1.

C. Standard single measurement targets

This section investigates how ET-GM-PHD handles standard
targets that produce at most one measurement per time step, in
comparison to standard GM-PHD which is designed under the
standard target measurement generation assumption. Note that
the measurement set cardinality distribution (i.e. the probabil-
ity mass function for the number of measurements generated
by a target) for a standard target contains only a single nonzero
element at cardinality 1, which is impossible to model with a
\(^2\)Note that a standard target always generates a single measurement.
Whether no measurements or a single measurement is obtained from the
standard target is determined by the detection process.
Fig. 3. Two simulation scenarios with spatially close targets. To the left, in (a), is a scenario where the two targets move closer to each other and then stand still at a distance of 60m apart. Note that the true \( y \) position was 300m for both targets for the entire simulation. To the right, in (b), is a scenario where the two targets cross paths, at the closest point they are 50m apart. The top and middle rows show the true \( x \) and \( y \) positions over time as a gray solid line and a black dash-dotted line. The light gray shaded areas show the target extent, taken as two measurement noise standard deviations (40m). The bottom row shows the distance between the two targets over time.

Fig. 4. Simulation results for the two scenarios in Fig. 3, comparing different partitioning methods for different values of the expected number of measurements \( \gamma \). The top row, (a), (b) and (c), is for the true tracks in Fig. 3a. The bottom row, (d), (e) and (f), is for the true tracks in Fig. 3b. Black shows Distance Partitioning with Sub-Partition, gray is only Distance Partitioning. It is clear from the plots that using Sub-Partition gives significantly better results, especially when \( \gamma \) is higher.

Fig. 5. Results from multiple target tracking using synthetic data. (a) The true \( x \) and \( y \) positions are shown in black, the light gray shaded areas show the target extent, taken as two measurement noise standard deviations (40m). (b) Mean OSPA (solid lines) \( \pm 1 \) standard deviation (dashed lines). (c) Mean cardinality compared to the true cardinality.

The Poisson distribution underlying the ET-GM-PHD filter. Hence, the case where each target generates measurements whose number is Poisson distributed with rate \( \gamma = 1 \) is very different from the standard target measurement generation.
Four targets were simulated in 100 Monte Carlo simulations, and all the targets were separated, i.e., there were no track crossings or new target spawn. Initially, in the ET-GM-PHD filter, $\gamma^{(j)}$ are all set as $\gamma^{(j)} = 1$ in the comparison. The average sum of weights and the average number of extracted targets (obtained by rounding the weights to the nearest integer) for the algorithms are shown in Fig. 6a and Fig. 6b respectively. As is shown, the sum of weights and number of extracted targets are too large for the ET-GM-PHD filter. The reason for this is that when the expected number of measurements per target (i.e., $\gamma^{(j)}$) is small, the effective probability of detection

$$p_{D, eff}^{(j)} = \left(1 - e^{-\gamma^{(j)}}\right) p_{D}^{(j)} \tag{29}$$

becomes significantly smaller than one. For example, the case $\gamma^{(j)} = 1$ and $p_{D}^{(j)} = 0.99$ gives $p_{D, eff}^{(j)} = 0.6258$. This low effective probability of detection is what causes the weights in the ET-GM-PHD filter to become too large.

Actually, this problem has been seen to be inherited by the ET-GM-PHD filter from the standard PHD filter. We here give a simple explanation to the problem with low (effective) probability of detection in the PHD filter. Assuming no false alarms, and a single target with existence probability $p_{e}$, ideally a single detection should cause the expected number of targets to be unity. However, applying the standard PHD formulae to this simple example, one can calculate the updated expected number of targets to be $1 + p_{e}(1 - p_{D})$ whose positive bias increases as $p_{D}$ decreases. We have seen that when the (effective) probability of detection is low, the increase in $\sum_{j=1}^{J} w_{k|j}^{(j)}$ is a manifestation of this type of sensitivity of the PHD type filters. A similar sensitivity issue is mentioned in [22] for the case of no detection.

This problem can be overcome by increasing $\gamma^{(j)}$ slightly in the ET-GM-PHD filter, e.g. $\gamma^{(j)} = 2$ gives $p_{D, eff}^{(j)} = 0.8560$ which gives sum of weights and number of extracted targets that better match the results from GM-PHD, see Fig. 6c and Fig. 6d. Using $\gamma^{(j)} = 3$ gives results that are more or less identical to GM-PHD, thus a conclusion that can be drawn is that when tracking standard targets with an ET-GM-PHD filter, the parameter $\gamma^{(j)}$ should not be set too small. The following subsection investigates the issue of selecting the parameter $\gamma$ in more detail.

D. Unknown expected number of measurements $\gamma$

In the simulations above, the parameters $\gamma = \gamma^{(j)}$ was assumed to be known a priori. Further, in Section IV-C where Sub-Partitioning is presented, the knowledge of the Poisson rate $\gamma$ was used to determine whether a cell should be split or not to create an additional partition. In this section, some scenarios where $\gamma$ is not known a priori are investigated. For the sake of clarity, $\gamma$ is used to denote the true Poisson rate with which measurements were generated, and $\hat{\gamma}$ is used to denote the corresponding parameter in the ET-GM-PHD filter. In many real world scenarios, the number of measurements generated by a target is dependent on the distance between the target and the sensor. Typically, the longer the distance, the lower the number of measurements generated by the targets. This is true for sensors such as the laser range sensor, the radar sensor and even cameras. Thus, it is of interest to evaluate the ET-GM-PHD-filter in a scenario where the number of generated measurements varies with the target to sensor distance. This is simulated in Section VI-D1, where the ET-GM-PHD filter is compared for the cases when the parameter $\hat{\gamma}$ is constant, and when the parameter is modeled as distance varying. Section VI-D2 presents results from simulations where the parameter $\hat{\gamma}$ is set incorrectly, and Section VI-D3 presents results with targets of different sizes. Finally, Section VI-D4 presents a discussion about the results from the simulations, and supplies some guidelines into the selection of $\hat{\gamma}$.

1) Distance varying $\gamma$: A scenario was constructed where a target moved such that the target to sensor distance first decreased, and then increased. The sensor was simulated such that the true parameter $\gamma$ depended on the target distance $\rho$ as follows.

$$\gamma(\rho) = \begin{cases} 1, & \text{if } \rho > 400m \\ [-0.08\rho + 33.5], & \text{if } 100m \leq \rho \leq 400m \\ 25, & \text{if } \rho < 100m \end{cases} \tag{30}$$

where $\lfloor \cdot \rfloor$ is the floor function. Thus, at distances larger than 400m, with $p_{D}^{(j)} = 0.99$, the effective probability of detection is only 0.6258 (as in the previous subsection). Note that the scenario is different from a target that always generates one measurement, which is detected with probability $p_{D}^{(j)} = 0.99$.

Monte Carlo simulations were made with two ET-GM-PHD-filters: one with constant value $\hat{\gamma} = 10$ and another where $\hat{\gamma}$ was selected to be dependent on the state of the targets via the function (30). The results are shown in Figure 7. For constant $\hat{\gamma}$, the number of targets is underestimated when the true $\gamma$ is low. This is due to the fact that the filter expects a target to generate more measurements, and thus the likelihood that some small number of measurements are all clutter is higher. However, at distances $\rho$ such that $\gamma(\rho) > 5$, $\hat{\gamma} = 10$ works quite well. When the model (30) for distance dependent $\gamma$ is assumed known, the results are much more reasonable and acceptable. The only, and possibly negligible, drawback seems to be the number of targets being slightly overestimated. There are two main reasons for this. The first reason is the low effective probability of detection when $\hat{\gamma}$ is low. This is due to the fact that the filter expects a target to generate more measurements, and thus the likelihood that some small number of measurements are all clutter is higher. However, at distances $\rho$ such that $\gamma(\rho) > 5$, $\hat{\gamma} = 10$ works quite well. When the model (30) for distance dependent $\gamma$ is assumed known, the results are much more reasonable and acceptable. The only, and possibly negligible, drawback seems to be the number of targets being slightly overestimated.

3More correctly, $p_{D, eff}^{(j)}$ in (29) is the probability of the event that at least one measurement from the $(j)$th target is obtained by the sensor.

4Some extreme versions of this phenomenon for lower $p_{D}$ values are illustrated and investigated in detail in the recent work [21].
were run. The set of \( \hat{\gamma} \) values used is given as
\[
\hat{\gamma} = 10, 12, \ldots, 28, 30.
\]
The results, in terms of the sum of weights averaged over the Monte Carlo runs, are shown in Figure 8. The figure shows that for sufficiently separated targets, the ET-GM-PHD-filter correctly estimates the number of targets for all values of \( \hat{\gamma} \). However, for spatially close targets, the ET-GM-PHD-filter overestimates the number of targets when \( \hat{\gamma} \) is set too low, and underestimates the number of targets when \( \hat{\gamma} \) is set too high. This result is actually expected, and is a direct consequence of the simple sub-partitioning algorithm we use. When \( \hat{\gamma} \) is too low, Sub-Partitioning creates an additional partition with too many cells, causing the overestimation of number of targets. Conversely, when \( \hat{\gamma} \) is too high, Sub-Partitioning does not create partitions with sufficient number of cells, causing the underestimation of number of targets. It is very important to note here that Sub-Partitioning operation actually runs even when the targets are well separated and does not cause any overestimation. Our observations show that this is a result of the fact that additional partitions created (when \( \hat{\gamma} \) is selected too low) cannot win over single target partitions when the targets measurements are distributed in a blob shape. It is only when the two targets approach each other resulting in an eight-shaped cluster of measurements that the additional partition can gain dominance over the single target partition. This property, though not proved mathematically, is considered to be a manifestation of the Poisson property and the Gaussianity assumptions underlying the measurements.

If the cardinality estimates of the algorithms are rounded to the nearest integer, an interesting property observed with Figure 8 is that no cardinality error appears for the cases that satisfy
\[
\hat{\gamma} - \sqrt{\hat{\gamma}} \leq \gamma \leq \hat{\gamma} + \sqrt{\hat{\gamma}}.
\]
Thus, when the true parameter \( \gamma \) lies in the interval determined by the mean (\( \hat{\gamma} \)) \pm one standard deviation (\( \sqrt{\hat{\gamma}} \)), cardinality is expected to be estimated correctly even for close targets.

3) Targets of different size: In many scenarios, it is possible to encounter multiple targets of different sizes, thus producing a different number of measurements. This means that two targets would not have the same Poisson rate \( \gamma \). In this section, the results are presented for a scenario with two targets with measurement generating Poisson rates of 10 and 20, respectively. In Monte Carlo simulations, three ET-GM-PHD-filter were run with the parameter \( \hat{\gamma} \) set to 10, 15 and 20, respectively. This corresponds to using either the true value of the smaller target, the mean of both, or the true value...
of the larger target. The results, in terms of average sum of weights over time are shown in Figure 9. When the targets are spatially separated, all three filters perform equally well. However, when the targets are spatially close, the target with is increasingly important. For targets that are spatially close, it is important for to be a good estimate of , since the Sub-Partition algorithm relies on . When such a good estimate is unavailable, a more advanced sub-partitioning algorithm seems to be necessary for robustness. With the proposed sub-partitioning procedure in this work, our findings support the intuitive conclusion that the true parameter should be in one standard deviation uncertainty region around the mean of the Poisson distribution for a reasonable performance for close targets.

The simulation with different target sizes shows that the close target case in this example is harder to tackle than the others. A possible solution is to adaptively estimate the parameters for each Gaussian mixture component based on the previous measurements. Another solution, which is possibly more straightforward, is to use a state dependent parameter, where the state contains information about the target extent, which can readily be estimated, see e.g. [9], [23]–[26]. Using the estimated shape and size, and a model of the sensor that is used, can then be estimated with reasonable accuracy. This has indeed recently been performed using an ET-GM-PHD-filter [9].

VII. EXPERIMENT RESULTS

This section presents results from experiments with data from two data sets acquired with a laser range sensor. The experiments are included more as a proof of concept and as a potential application, rather than as an exhaustive evaluation of the presented target tracking filter. The measurements were collected using a SICK LMS laser range sensor. The sensor measures range every over a 180° surveillance area. Ranges shorter than 13 m were converted to measurements using a polar to Cartesian transformation.

The two data sets contain 411 and 400 laser range sweeps in total, respectively. During the data collection humans moved through the surveillance area, entering the surveillance area at different times. The laser sensor was at the waist level of the humans.

Since there is no ground truth available it is difficult to obtain a definite measure of target tracking quality, however by examining the raw data we were able to observe the true cardinality, which can thus be compared to the estimated cardinality.

Section VII-A presents results from an experiment with spatially close targets, and Section VII-B presents results from an experiment with both spatially close targets and occlusion.

A. Experiment with close targets

In this experiment, a data set containing 411 laser range scans was used. The data set contains two human targets that repeatedly move towards and away from each other, moving right next to each other at several times. The two targets passed each other at close distance moving in the opposite direction, representing instances in time when the targets were close for short periods of time. The targets also moved close to each other moving in the same direction, representing instances in time when the targets were close for longer periods of time.

The locations of the extracted Gaussian components are shown in Fig. 10a, the number of extracted targets is shown in Fig. 10b and the sum of weights are shown in Fig. 10c. Around time 320 there is a decrease in the number of extracted targets for three time steps, in all other situations the filter handles the two targets without problem. Thus, using Sub-Partition with K-means as split function, the ET-GM-PHD filter can be said to handle most of the spatially close target cases.

B. Experiment with occlusion

In this experiment, a dataset containing 400 laser range scans was used. The data set contains four human targets that move through the surveillance area, however there are at most three targets present at any one time. The first target enters the surveillance area at time and proceeds to the center of
the surveillance area where he remains still for the remainder of the experiment. The second target enters the surveillance area at time $k = 38$ and repeatedly moves in front of and behind the first target. The third target enters and exits at time $k = 283$ and $k = 310$, respectively. The last target enters and exits at time $k = 345$ and $k = 362$, respectively.

This case requires a state dependent (i.e. variable) probability of detection $p_D(\cdot)$ selection for the targets. Otherwise, i.e. with a constant probability of detection assumption, when a target is occluded, this would be interpreted as the exit of the target from the area of surveillance while it is only the disappearance of the target behind another. The variable $p_D$ is modeled as a function of the mean, covariance and the weights of the Gaussian components. The intuition behind this idea is that the knowledge of the targets that are present, i.e. the knowledge of the estimated Gaussian components of the PHD, can be used to determine what parts of the surveillance area are likely to be in view of the sensor, and which parts are not. Leaving the details of the variable $p_D$ calculation to Appendix B, we present the results below.

The locations of the extracted Gaussian components are shown in Fig. 11a, the number of the extracted targets is shown in Fig. 11b and the sum of weights are shown in Fig. 11c. In total, there are six situations where one target is occluded by another. The extracted number of targets is wrong in two of these situations, where the occluded target is spatially very close to (right next to) the target which is causing the occlusion. The ET-GM-PHD filter correctly estimates the cardinality in four out of six occlusions.

Thus, using the suggested variable $p_D$, the filter can correctly predict the target while it is occluded, provided that it is not very close to another target while the occlusion is happening. If $\sum_{k=1}^{J_{i,k}} w^{(j)}_{k|k}$ is rounded to the nearest integer there is no cardinality error for the first four occlusions. However, as the target exits the occluded area there is a “jump” in $\sum_{j=1}^{J_{i,k}} w^{(j)}_{k|k}$ around times $k = 75$, $k = 125$, $k = 175$ and $k = 210$, see Fig. 11c. We have seen that this “jumping” behavior is caused by the sensitivity of the cardinality estimates of the PHD filter to detections when $p_D^{(j)}$ is set to a low value, which is the case when the target is half
occluded while it gets out of the occluded region. This is the same phenomenon observed with low effective probability of detection in Section VI-C.

VIII. CONCLUSIONS AND FUTURE WORK

In this paper a Gaussian mixture implementation of the probability hypothesis density filter for tracking extended targets was presented. It was shown that all possible partitions of the measurement set does not need to be considered, instead it is sufficient to consider a subset of partitions, as long as this subset contain the most probable partitions. A simple method for finding this subset of all measurement set partitions was described. This partitioning method is complemented with a sub-partitioning strategy to handle the cases that involve close targets better. Simulations and experiments have shown that the proposed filter is capable of tracking extended targets in cluttered measurements. The number of targets is estimated correctly even for most of the cases when tracks are close. The detailed investigations carried out gave some guidelines about the selection of the Poisson rate for the cases when it is unknown. Using inhomogeneous detection probabilities in the surveillance region, it was shown that targets can be tracked as they move through occluded parts of the surveillance area.

In recent work, a cardinalized PHD filter [27] for extended targets has been presented [21]. This filter has less sensitive estimates of the number of targets. Initial step have also been taken towards including estimation of target extent in the ET-GM-PHD-filter [9]. More work is needed in both of these research directions.

A further interesting research can be to see the potential use of the partitioning algorithms presented in this work with more conventional multiple target tracking algorithms. A comparison of such algorithms with the ET-GM-PHD filter can illustrate the advantage coming from the use of the random set framework.

APPENDIX A

PROOF OF THEOREM 1

The proof is composed of two parts.

- We first prove that there is a partition satisfying the conditions of the theorem. The proof is constructive. Consider the algorithm in Table IV. In the algorithm, one first forms a partition formed of singleton sets of the individual measurements and then combine the cells of this cluster until conditions of the theorem are satisfied.

- We need to prove that the partition satisfying the conditions of the theorem is unique. The proof is by contradiction. Suppose that there are two partitions $p_i$ and $p_j$ satisfying the conditions of the theorem. Then, there must be a measurement $z_m \in Z$ such that the cells $W_{m_i}$ and $W_{m_j}$ are different, i.e., $W_{m_i} \neq W_{m_j}$. This requires an additional measurement $z_m \in Z$ that is in one of $W_{m_i}$, $W_{m_j}$ but not in the other. Without loss of generality, suppose $z_n \in W_{m_i}$ and $z_n \notin W_{m_j}$. Then, since $z_n \in W_{m_i}$, we know that $d(z^{m_i}, z^{m_j}) \leq d_k$. However, this contradicts the fact that $z_n \notin W_{m_j}$ which means that $p_j$ does not satisfy the conditions of the theorem. Then our initial assumption that there are two partitions satisfying the conditions of the theorem must be wrong. The proof is complete. □

APPENDIX B

VARIABLE PROBABILITY OF DETECTION FOR THE LASER SENSOR

With the variable probability of detection function we reduce $p_D$ behind (i.e. at larger range from the sensor) each component of the PHD according to the weight and bearing standard deviation of the Gaussian components.

For a given point $x$ in the surveillance area, the probability of detection $p_D(x)$ is computed as

$$p_D(x) = \max (p_{D,\min}, p_D^0)$$

$$p_D^0 = p_{D,0} - \sum_{i: r > r^{(i)}} w^{(i)} \sqrt{\frac{\sigma_x^2}{\sigma_{\varphi^{(i)}}^2}} \exp \left( - \frac{\left( x - x^{(i)} \right)^2}{2 \sigma_{\varphi^{(i)}}} \right)$$

where

- $p_{D,\min}$ is the minimum probability of detection value a target can have;
- $p_{D,0}$ is the nominal probability of detection of the targets when they are not occluded;
- $r$ and $\varphi$ are the range and bearing of point $x$ with respect to the sensor respectively;
- $r^{(i)}$ and $\varphi^{(i)}$ are the range and bearing of the $i$th Gaussian component with respect to the sensor respectively;
- $w^{(i)}$ is the weight of the $i$th component;
- $\sigma_{\varphi^{(i)}}$ is defined as

$$\bar{\sigma}_{\varphi^{(i)}} \triangleq \begin{cases} \sigma_{\max}, & \text{if } \sigma_{\varphi^{(i)}} > \sigma_{\max} \\ \sigma_{\min}, & \text{if } \sigma_{\varphi^{(i)}} < \sigma_{\min} \\ \sigma_{\varphi^{(i)}}, & \text{otherwise} \end{cases}$$

where $\sigma_{\varphi^{(i)}}$ is the bearing standard deviation of the $i$th component given as

$$\sigma_{\varphi^{(i)}} \triangleq \sqrt{\frac{u^{(i)} \cdot p^{(i)}_p}{u_{\varphi^{(i)}}}}$$

Here, $F_p^{(i)}$ is the position covariance of the $i$th component and $u_{\varphi^{(i)}}$ is the unit vector orthogonal to the range direction from the $i$th component to the sensor.
The constant term $\sigma_s$ is used to scale the bearing standard deviation.

Intuitively, the operation of (34) is to reduce the nominal probability of detection at a point depending on the weights, means and standard deviations of the components of the last estimated PHD which have smaller range values than the range of the point also taking into account the angular proximity of such components to the bearing of the point in the exponential function in (34).

ACKNOWLEDGMENT

The authors would like to thank André Carvalho Bittencourt of Linköping University for the help with the acquisition of the data used in the experiments and Professor Fredrik Gustafsson of Linköping University for his valuable comments on the manuscript.

REFERENCES


### Abstract

This paper presents a Gaussian-mixture implementation of the PHD filter for tracking extended targets. The exact filter requires processing of all possible measurement set partitions, which is generally infeasible to implement. A method is proposed for limiting the number of considered partitions and possible alternatives are discussed. The implementation is used on simulated data and in experiments with real laser data, and the advantage of the filter is illustrated. Suitable remedies are given to handle spatially close targets and target occlusion.