Probability Hypothesis Density filter versus Multiple Hypothesis Tracking

Kusha Panta[a], Ba-Ngu Vo[a], Sumeetpal Singh[a] and Arnaud Doucet[b]

[a] Co-operative Research Centre for Sensor and Information Processing (CSSIP), Department of Electrical & Electronic Engineering, The University of Melbourne, Melbourne, Victoria 3010, Australia;

[b] Department of Engineering, Cambridge University, CB2 1PZ Cambridge, UK.

ABSTRACT

The probability hypothesis density (PHD) filter is a practical alternative to the optimal Bayesian multi-target filter based on finite set statistics. It propagates only the first order moment instead of the full multi-target posterior. Recently, a sequential Monte Carlo (SMC) implementation of PHD filter has been used in multi-target filtering with promising results. In this paper, we will compare the performance of the PHD filter with that of the multiple hypothesis tracking (MHT) that has been widely used in multi-target filtering over the past decades. The Wasserstein distance is used as a measure of the multi-target miss distance in these comparisons. Furthermore, since the PHD filter does not produce target tracks, for comparison purposes, we investigated ways of integrating the data-association functionality into the PHD filter. This has lead us to devise methods for integrating the PHD filter and the MHT filter for target tracking which exploits the advantage of both approaches.

Keywords: Multi-target tracking, Random Sets, Probability Hypothesis Density, Multiple Hypothesis Tracking

1. INTRODUCTION

In the multi-target filtering problems, the number of individual targets and the measurements generated by these targets may change over time as the targets or clutter appear and disappear in the scene. One approach to filtering for multiple targets is to consider each target separately from others and track it with a separate filter. However this requires correct association of individual targets with its measurement among the collection of targets and measurements[1,2] and leads to a model-data association problem. Multiple hypothesis tracking (MHT) is a data association technique that has been widely used in multi-target filtering. MHT allows the use of measurements that arrive in future to resolve the uncertainty in the correct association of measurements and targets at present. Another approach involves modelling of the collection of individual targets and measurements with random finite sets (RFS) and propagating the first order moment of the multi-target density of the RFS[3–5] analogous to the tracking of a single target in the constant gain Kalman filtering. This approach however keeps no records of the target identities and avoids the data-association problem.

The Probability Hypothesis Density (PHD) filter is a computationally cheaper alternative to optimal multi-target filtering. It recursively propagates the PHD or the first order moment of the multi-target posterior, assuming the Poisson characteristics of the multi-target density. However, the inherent need for the evaluation of multiple integrals that have no closed form in general makes the implementation of the PHD filter extremely difficult. Recently, a technique that approximates the PHD recursion using a Sequential Monte Carlo (SMC) method has been proposed in Refs. 6 and 7.

Further author information: (Send correspondence to Kusha Panta)
Kusha Panta: E-mail: kusha@ee.mu.oz.au, Telephone: + 613 8344 7436
Ba-Ngu Vo: E-mail: bv@ee.mu.oz.au, Telephone: + 613 8344 6693
Sumeetpal Singh: E-mail: ssss@ee.mu.oz.au, Telephone: + 613 8344 9206
Arnaud Doucet: E-mail: ad2@eng.cam.ac.uk, Telephone: + 44 1223 332 676

The aim of the paper is to compare the performance of the PHD filter with that of the MHT filter. The comparison can be made in terms of the point state estimates of the targets at each time step. This can be achieved by calculating the Wasserstein distance of the point state estimates of the targets from the ground truth for each filter. However the MHT filter does more than just provide the estimates of the target states, it provides target tracks which the PHD filter cannot. Therefore, it may not be fair to compare the PHD and the MHT filter only in terms of the point state estimates of the targets. In this paper, we have investigated possible ways of adding the data-association functionality to the PHD filter to do the comparison in terms of tracks. This lead us to consider a number of alternative schemes for integrating the PHD and the MHT filter so as to exploit the advantages of both methods.

The rest of the paper is organized as follows. Section 2.1 describes the multi-target model considered in this paper. Section 2.2 describes the PHD, the PHD filter and its SMC implementation. Section 2.3 presents an overview of multi-target filtering with MHT. Section 3 presents the comparison of the performance of the PHD filter with that of the MHT filter. Section 4 presents novel schemes for adding target tracking capability to the PHD filter. Finally Sect. 6 summarizes the results obtained in Sects. 3 and 5.

2. MULTI-TARGET FILTERING BACKGROUND

This section introduces the backgrounds to the problem of multi-target filtering and the description of two different approaches to multi-target filtering using the PHD and the MHT filter. Section 2.1 presents the multi-target model of the multi-target scenario used in this paper. Section 2.2 introduce the PHD filter and its particle approximation with the SMC method. Section 2.3 present a brief description of multi-target filtering with the MHT filter.

2.1. Multi-Target Model

In a multi-target scenario, targets appear and disappear randomly. For the duration the target is present, it moves according to a Markov dynamic model

$$x_{k+1} \sim f_{k+1|k}(\cdot|x_k), \tag{1}$$

and generates observation according to

$$y_{k,i} \sim g_k(\cdot|x_{k,i}). \tag{2}$$

At time k, let $M(k)$ be the number of targets present with states $x_{k,1}, \ldots, x_{k,M(k)}$, and $N(k)$ the corresponding number of measurements received. Let

$$X_k = \{x_{k,1}, \ldots, x_{k,M(k)}\} \subset E_s, \tag{3}$$

$$Y_k = \{y_{k,1}, \ldots, y_{k,N(k)}\} \subset E_o, \tag{4}$$

denote the set of targets and measurements received at time k. E_s and E_o represent the state and the observation space where individual targets and observations respectively lie. Some of the $N(k)$ observations may be due to clutter. If $y_{k,i}$ is due to clutter, then $y_{k,i} \sim c_k(\cdot)$. The number of clutter points are assumed to be Poisson distributed with a mean of λ_k.

2.2. Multi-Target Filtering with the PHD Filter

Finite set statistics (FISST)[4,5] enables the multi-target filtering problem to be formulated in the Bayesian framework. Optimal Bayes multi-target filtering involves propagating the multi-target posterior density in time. However the inherent computational intractability means that we have to approximate the multi-target posterior density with its statistical moments and propagate the moments instead. Assuming the point process represented by the RFS is Poisson, its statistics is completely characterized by the its first moment.[3] This section introduces the first order moment of a RFS and multi-target filtering using the first order moment.

2.2.1. Probability Hypothesis Density

The probability hypothesis density (PHD) D_Ξ is the first order moment of the RFS Ξ[5, 6]and is given by

$$D_\Xi(x) \equiv \mathbf{E}[\delta_\Xi(x)] = \int \delta_X(x) P_\Xi(dX) \tag{5}$$

where $\delta_\Xi(x)$ is the random density representation of Ξ and equals the summation of Dirac delta functions centered at x for each $x \in \Xi$, i.e., $\delta_\Xi(x) = \sum_{x \in \Xi} \delta_x$.

The PHD D_Ξ of Ξ is a unique function on the space E where the individual targets exist and its integral over a measurable subset $S \subseteq E$, i.e., $\int_S D_\Xi(x)\lambda(dx)$, yields the expected number of elements of Ξ that are present in S. It can be constructed from belief functions of RFSs using FISST.[3, 4] Moreover, the peaks of the PHD of Ξ gives the estimates of the elements of Ξ.

2.2.2. The PHD filter

The PHD filter consists of the *prediction* operator and the *update* operator. Assuming the RFS is Poisson, it has been shown that the recursion propagating the PHD $D_{k|k}$ of the multi-target posterior $p_{k|k}$ follows[5]

$$D_{k|k} = (\Psi_k \circ \Phi_{k|k-1})(D_{k-1|k-1})$$

where Ψ_k is the prediction operator and $\Phi_{k|k-1}$ is the update operator. These operators are defined as follows:

$$(\Phi_{k|k-1}\alpha)(x) = \int \phi_{k|k-1}(x,\xi)\alpha(\xi)\lambda(d\xi) + \gamma_k, \tag{6}$$

$$(\Psi_x \alpha) = [\upsilon(x) + \sum_{z \in Z_k} \frac{\psi_{k,z}(x)}{\kappa_k(z) + \langle \psi_{k,z}, \alpha \rangle}]\alpha(x), \tag{7}$$

for any (integrable) function α on E_s, where

$$\phi_{k|k-1}(x,\xi) = e_{k|k-1}(\xi) f_{k|k-1}(x|\xi) + b_{k|k-1},$$

$$\upsilon(x) = 1 - p_D(x),$$

$$\psi_{k,z}(x) = p_D(x) g_k(z|x),$$

$$\kappa_k(z) = \lambda_k c_k(z)$$

and γ_k denotes the PHD of the RFS Γ_k of targets which appear spontaneously; $b_{k|k-1}(\cdot|\xi)$ denotes the PHD of the RFS $B_{k|k-1}(\{\xi\})$ spawned by a target with previous state ξ; $e_{k|k-1}(\xi)$ denotes the probability that the target still exist at time k given that it had previous state ξ; $f_{k|k-1}(\cdot|\cdot)$ denotes the transition probability density of individual targets; $g_k(\cdot|\cdot)$ denotes the likelihood of individual targets; c_k denotes the clutter probability density; λ_k denotes the average number of Poisson clutter points per time step; and p_D denotes the probability of detection. The details on the derivation of these densities and likelihood functions from the underlying models of the sensors, individual targets dynamics, target births and deaths are found in Refs. 3 and 4.

Although the PHD filtering is a computationally cheaper alternative to optimal multi-target filtering, it still involves computation of multiple integrals that have no closed form in general making it computationally intractable. However, a generalized sequential Monte Carlo (SMC) implementation of the PHD filtering is proposed in Ref. 6. A brief description of the SMC implementation of the PHD filter is included in Appendix A. The proposed SMC implementation is computationally efficient and takes care of the time-varying number of targets.

2.3. Multi-target Filtering with Multiple Hypothesis Tracking

Given the set of observations, multi-target filtering based on data association requires correct partitioning of observations amongst individual targets and clutter. A simple approach would be to find the most probable association of measurements with individual targets in each time step. Instead of making the decision on the most probable data partitioning at each time step, the MHT hypothesizes several possible partitioning of observations and propagates these hypotheses so that the uncertainty in the correct partitioning can be reduced on the arrival of subsequent data.[1, 2, 8–10] Thus, the MHT allows the use of later measurements in the prior data partitioning. However the number of hypotheses can grow exponentially over time and the required computational costs could render the implementation of MHT infeasible. *Gating* and *pruning* techniques are the commonly used *ad-hoc* methods to limit the number of track hypotheses at each time step by eliminating the unlikely and the least likely track-to-measurement association.

In this section, we briefly describe an MHT algorithm that has been presented in detail in Refs. 9 and 10. A track-oriented MHT is chosen over a hypothesis-oriented one as this approach is simpler to implement and results in smaller number of hypotheses. In a track-oriented approach, the number of possible tracks are pruned by eliminating tracks with low probabilities before the hypotheses are formed. The likelihood of each track hypothesis is determined by its likelihood score, often maintained as log-likelihood ratio (LLR).

2.3.1. Overview of a track-oriented MHT

In a track-oriented approach, each track hypothesis represents a collection of assigned observations over time that is likely to have originated from the same target. It makes no assumptions on the origin of the measurements. To begin with tracks are initiated for all measurements. Clusters of tracks are formed so that tracks in one cluster share observation amongst each other and do not share observations with tracks from another clusters. Fig. 1 shows a block diagram of a practical implementation of the track-oriented MHT. Given a number of track

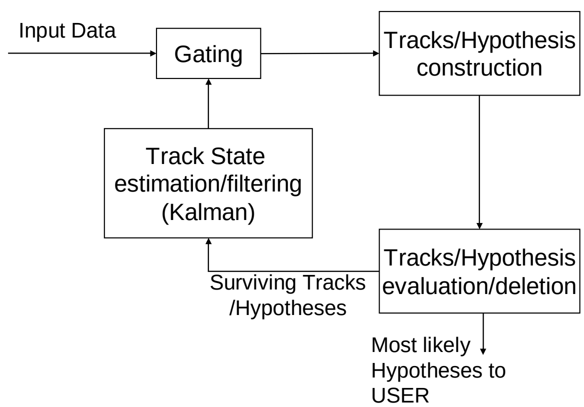

Figure 1. An overview of track-oriented MHT algorithm.

hypotheses at time step $k-1$, MHT allows measurement prediction for each track hypothesis and the predictions are gated with the noisy measurements of time step k. For m_k measurements that fall within the gate of a track prediction, m_k association track hypotheses are constructed. When an existing track is not gated with any measurement, a target detection-miss is noted and the track is propagated ahead. A new track is started for all measurements that are not gated with any of the existing tracks. Upon the creation of each association track hypothesis, its score or LLR is updated.

Pruning of tracks are performed at the track level based on their LLR and the number of consecutive target detection-misses. Track hypotheses with N (usually 3 or more) consecutive miss-detections or with LLRs smaller than a chosen *threshold* are deleted. Track hypotheses that have at least N (usually 3 or more) target detections are considered to be *confirmed*. Only confirmed track hypotheses are considered for data outputs. Confirmed track hypotheses are also subject to N-scan pruning on a global level. The choice of the actual parameter values used in pruning depend on the tracking scenario. The tracks that survive pruning are updated with its gated observation and propagated to the next time step.

For each target, there may exist multiple track hypotheses representing multiple assignments of observations over the subsequent time steps. All track hypotheses that are started by the same target form a tree structure with the same root and track hypotheses as the branches of the tree. The MHT chooses the most likely track hypothesis from each tree representing a true target track to produce a collection of hypotheses which are known as global hypotheses. The sequence of observations represented by each global hypothesis is filtered to produce the track estimates of targets present in the scene.

The MHT uses extended Kalman filtering (EKF) for prediction and update on each track hypothesis. However SMC methods have also been proposed instead of EKF.[11, 12] See Refs. 1 and 9 for the detailed description and the issues related to the implementation of the track-oriented MHT.

3. THE PERFORMANCE OF THE PHD AND MHT FILTER FOR STATE ESTIMATES

For illustration purpose only, we consider a simple one-dimensional scenario, in which the targets move along the line segment $[-100; 100]$ and can appear or disappear in the horizon at any time. The target states consist of positions and velocities with only the position measurements available. We assume that target birth follows a Poisson model with the intensity $0.2\mathcal{N}(\cdot|0,1)$ where $\mathcal{N}(\cdot|0,1)$ represent zero mean and unit variance normal distribution and targets have linear Gaussian dynamics. Similarly, the clutter process is also Poisson with uniform intensity over the region $[-100; 100]$ and has an average rate of 2.0. The probability of target being detected is 0.99 (≈ 1) and the survival of each target is state independent with a probability $e = 0.8$. Observations are made according the multi-target model described in Sect. 2.1 above.

In this particular implementation of MHT, the ellipsoid gate size equals 6.63. A hypothesis track is deleted if it has either 3 or more consecutive target-detection misses. Tracks are not deleted on the basis of their likelihoods as this implementation of MHT can handle the number of track hypotheses without. Track hypotheses are confirmed once they have at least 3 target detections. For data output, the best global hypothesis is considered.

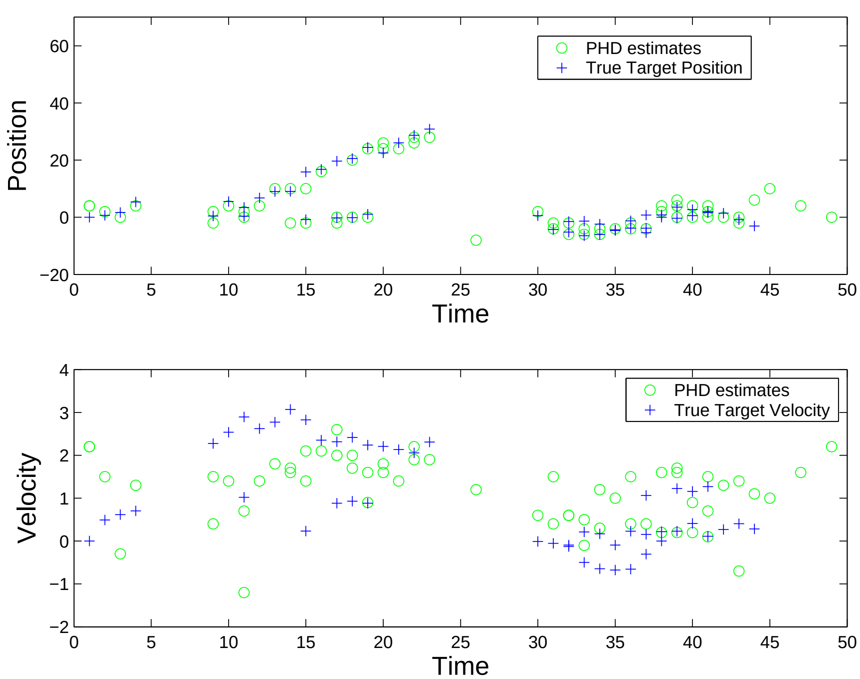

Figure 2. Point estimates of the target states with the PHD filter.

Figures 2 and 3 show the state estimates of the targets given the PHD and the MHT filter for the same set of observations over a number of time steps. The vertical axis of Fig. 3 has been reset from $[-100, 100]$ to $[-20, 70]$ to give the clear comparison of the estimates. Figures 2 and 3 show that most of target positions are picked up by both filters. The PHD filter outputs fewer false alarms than the MHT filter. The velocity estimates in both cases are bad since only the position measurement is obtained. It should also be noted that for MHT to pick up target tracks, the target must be present in the scene for at least a number of time steps equal to the number of target-detection hits needed for a track hypothesis to be confirmed.

The performance of a multi-target tracking algorithm can be measured by a "distance" between two finite sets representing the ground truth (actual target states) and the corresponding point estimates produced by the algorithm. Hoffman and Mahler[13] have shown that the Wasserstein distance is a good measure of the multi-target miss distance. Given the multi-target ground truth $G(k) = \{g_1, \cdots, g_{M(k)}\}$ and the corresponding point estimates $\tilde{X}(k) = \{\tilde{x}_1, \cdots, \tilde{x}_{\tilde{M}(k)}\}$, the Wasserstein distance d_p^W is defined as

$$d_p^W(\tilde{X}, G) = \inf_C \sqrt[p]{\sum_{i=1}^{M(k)} \sum_{j=1}^{\tilde{M}(k)} C_{i,j} d(\tilde{x}_i, g_i)^p} \tag{8}$$

$$d_\infty^W(\tilde{X}, G) = \inf_C max_{1 \le i \le M(k), 1 \le j \le \tilde{M}(k)} \tilde{C}_{i,j} d(\tilde{x}_i, g_i) \tag{9}$$

where the infimum is taken over all $M(k) \times \tilde{M}(k)$ transportation matrices $C = \{C_{i,j}\}$; and $\tilde{C}_{i,j} = 1$ if $C_{i,j} \neq 0$ and $\tilde{c}_{i,j} = 0$ otherwise. An $M(k) \times \tilde{M}(k)$ matrix C is a transportation matrix if for all $i = 1, \ldots, M(k)$ and $j = 1, \ldots, \tilde{M}(k)$, $C_{i,j}$ satisfies the followings:

$$\sum_{i=1}^{\tilde{M}(k)} C_{i,j} = \frac{1}{\tilde{M}(k)},$$

$$\sum_{i=1}^{M(k)} C_{i,j} = \frac{1}{M(k)}$$

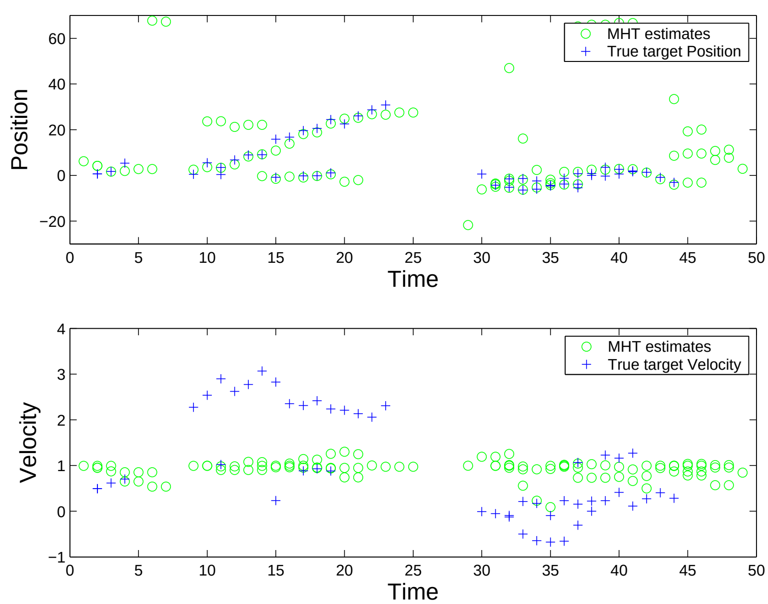

Figure 3. Point estimates of the target state with the track-oriented MHT filter.

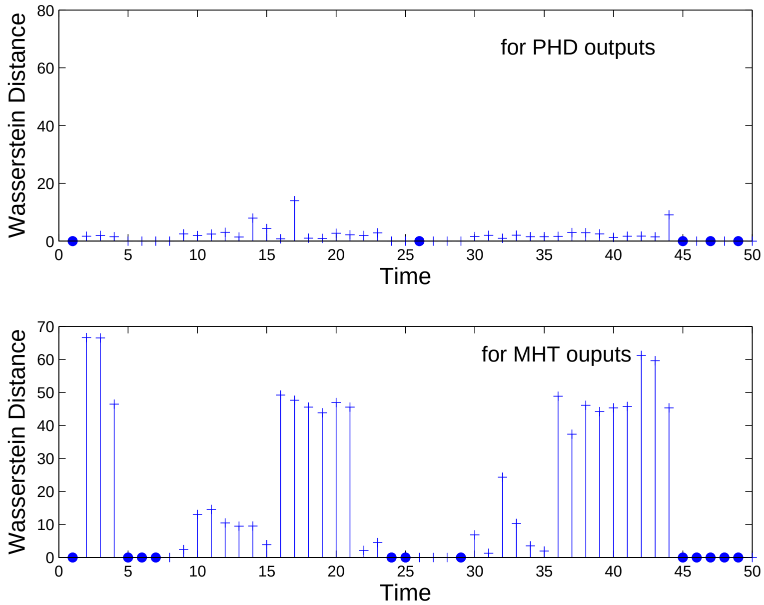

Figure 4. The Wasserstein distance between the point estimate outputs of the PHD and MHT filter and the ground truth

and

$$C_{i,j} \geq 0.$$

The Wasserstein distance is preferred over the commonly used Hausdorff distance for the Hausdorff distance is insensitive to the different number of objects in two sets and hence can not penalize for picking additional clutter in the point estimate outputs .[13]

It should be noted that the Wasserstein distance is not defined when one or both sets are empty. When both sets are empty, the miss distance can be set to zero. However when only one of the set is empty, the notion of miss distance is open to interpretation. However, if we extend the concept of the Wasserstein distance to include the empty set, an unbounded large multi-target miss distance between an empty and a non-empty set would be required to satisfy all the metric axioms. Figure 4 shows that the miss distance of the PHD outputs is lower than that of the MHT outputs. We have assigned zero distance when both the ground truth and the set of point estimates are empty. The filled circles on the plots represent the undefined miss distance between the non-empty set of point estimates of the filter and the empty ground truth.

The implementation of the MHT filter is more complicated than that of the PHD filter. The implementation of the MHT require a careful design of the data structures and algorithm that are complex. The MHT filter need to store previous observations and records of multiple track hypotheses. As a result, the MHT filter would require a large memory. The performance of the MHT filter heavily depends on the particular implementation of gating and pruning techniques[9,10] that are *ad-hoc* in general. A tradeoff exists between the performance of a MHT filter and the associated computational cost and memory. Additional improvement on the performance would require more memory and increased computational cost. On the other hand, the performance of the PHD filter is mainly dictated by the details of its particle approximation; i.e. the number of particles per targets, the choice of the proposal density function and the re-sampling scheme.

The computational comparison of the MHT and PHD filter implemented in this work requires additional consideration. The MHT filter used in this simulation is designed to exploit the linear and Gaussian target dynamics by employing the Kalman filtering. A general tracking scenario includes target dynamics that are nonlinear and noise that are non-Gaussian for which MHT would require particle filters for prediction and filtering on individual tracks. On the other hand, the PHD filter employs particle approximation of the target dynamics and hence can be used for non-linear data models without further modification. It is obvious that for general tracking purposes, multiple number of particle filters (working in parallel) would make the MHT filter computationally expensive than the PHD filter. Actual details on the computational cost is the subject of ongoing research.

4. MULTI-TARGET TRACKING WITH THE PHD FILTER

The PHD filter introduced in Sect. 2.2 above only gives the estimates of the states of targets that are present in the scene at any time step k, i.e. $\tilde{X}_k$. It keeps no records of the target identities and hence does not produce tracks followed by individual targets over time. However some data-association functionalities of MHT can be incorporated with the PHD filter to produce the desired tracks. This section will discuss several methods for doing so. We use the word 'tracking' to denote the trajectories followed by individual targets and the 'point state estimates' to denote the estimates of the target states at individual time steps.

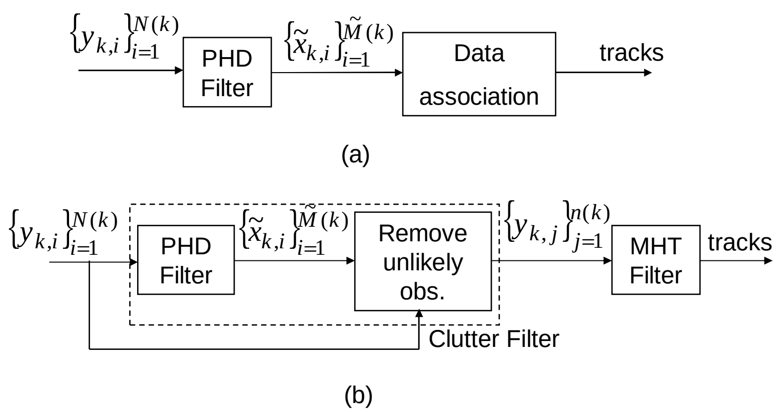

Figure 5. Target Tracking with the PHD filter.

4.1. Scheme One: PHD-with-association filter

This approach takes the output of the PHD filter and performs data association on them (see Fig. 5(a)). At each time step k, the PHD filter provides an estimate of the number and the states of the targets that are present in the scene, i.e. $\tilde{X}_k$. Assuming these estimates are sufficiently accurate, we may regard the output of the PHD filter as defining a new observation model given by

$$\tilde{y}_{k,i} = x_{k,i} + n_{k,i}, \quad i = 1, \ldots, \tilde{M}(k) \tag{10}$$

where $\tilde{M}(k)$ is the estimate of the number of targets present and error in the estimate $n_{k,i} = \tilde{x}_{k,i} - x_{k,i}$ is regarded as the noise. Thus, regardless of the fact that the observation process is non-linear or non-Gaussian, the data-association functionality is given a linear observation process. Implicit in this scheme is the assumption that $\tilde{M}(k) \approx M(k)$. There are a number of ways to estimate the statistics of the noise $n_{k,i}$. In this paper, we assume $n_{k,i}$ is a zero-mean Gaussian process with variance $Q_{k,i}$ that can be estimated from the particle approximation of the PHD recursion. In simulations, we find that this scheme works well although the true distribution of $n_{k,i}$ is not Gaussian.

The size of the modified observation set $\tilde{Y}_k$ given by Eq. (10) is smaller than that of the original observation set Y_k as each observation in $\tilde{Y}_k$ is associated with a target whilst Y_k has additional clutter. As a result, a much smaller number of track hypotheses are created at each time step and the computational cost of the data-association will be smaller as well. The data-association functionality can perform the track-to-estimate association in a way similar to the MHT filter performs track-to-measurement association. Pruning of track hypotheses can be simpler than that in the MHT. Since track consists of associated estimates, there is no need for further filtering on the tracks as is the case with the MHT filter. We remark that the computation and memory requirement of this PHD-with-association filter as compared to that of the MHT filter implemented with Y_k is ongoing work.

Furthermore, the point estimates of the target states given by the MHT filter can be used toconstruct a proposal density function that approximates the target PHD function. The closely the proposal density matches the target PHD function, the better is the performance of the particle filter. The proposal density $q_k(\cdot|x_{k-1}^{(i)}, Z_k)$ (see Step 1 of the algorithm of the PHD particle filter in Appendix A.) can be constructed as the mixture of weighted Gaussian distributions that are constructed from the respective means and variances of the point estimates of the output target tracks. For mean $\mu_{i,k}$ and variance $\sigma_{i,k}^2$ on track i at the time step k, $q_k(\cdot|x_{k-1}^{(i)}, Z_k)$ is constructed as

$$q_k(\cdot|x_{k-1}^{(i)}, Z_k) = \sum_i a_{i,k} f_{i,k}(\cdot), \tag{11}$$

where $f_{i,k}(\cdot) \sim \mathcal{N}(\mu_{i,k}, \sigma_{i,k}^2)$, weighting coefficient $a_{i,k} = \frac{LR_{i,k}}{\sum_i LR_{i,k}}$ and $LR_{i,k}$ is the likelihood score of the track i at the time step k. The proposal density function obtained according to Eq. (11) should work well provided the target dynamics and the measurement process are linear with Gaussian noise.

4.2. Scheme Two: The MHT-with-PHD clutter Filter

In the above method, the PHD filter was effectively used as a clutter pre-filter that feeds a modified observation set to the data-association functionality. We now describe how to use the PHD filter as a clutter filter but without modifying the observation process. The MHT is used for the purpose of data-association.

The proposed scheme is represented by the block diagram in Fig. 5(b). The PHD filter output $\tilde{X}_k$ are used to define validation gates in the observation space. The observations that fall outside the gates are discarded as clutter and the new reduced observation set is feed to the MHT to produce tracks. In summary, given an observation set $Y_k = \{y_{k,i}\}_{i=1}^{N(k)}$ at time step k, the new observation set is given by

$$\tilde{Y}_k = \{y_k^i : g_k(y_k^i|\tilde{x}_{k,j}) > g_d \quad \text{for some } j\},$$

where $\tilde{X}_k = \{\tilde{x}_{k,j}\}_{j=1}^{\tilde{M}(k)}$ is the PHD output at time step k and g_d is the observation gate threshold. Simulation results in Sect. 3 would justify the use of the PHD filter as a clutter filter. In effect, the PHD filter is used to eliminate most of the observations that are unlikely to have originated from the targets. Hence, this clutter filter approach can be viewed as a way of performing gating on a global level. The proposal density function used in the PHD recursion can be constructed according to the scheme presented above in 4.2.

5. SIMULATION RESULTS FOR TARGET TRACKING WITH THE PHD FILTER

This section shows the tracks produced by the MHT filter and the PHD filter. With the PHD filter, the tracks that have been obtained according to the scheme presented in Fig. 5(a) above. The implementation of the PHD filter follows the algorithm given below in Appendix A. For data-association, the generic MHT is used. Tracks are not subjected to pruning based on their likelihoods. For illustration purpose, we use the same multi-target scenario given in Sect. 3 above.

Figure 6 shows the tracks produced by applying the MHT filter on the noisy observation set generated according the scheme described in Sect. 3. In addition to picking up the majority of true tracks, the MHT filter also picks a number of false tracks. In this example, the number of false tracks is 11. The MHT fails to pick up four tracks that do not exist in the horizon for long enough (in this case a target has to exist at least over three time steps).

Figure 7 presents the tracks that are given by the PHD-with-association filter. The PHD-with-association filter picks up the most of the target tracks for majority of times. Missed tracks include four short tracks (not picked up due to their existence being smaller than required). Two tracks given by this method forms part of the same true track. However the number of false tracks is smaller than that given by the MHT filter. In this example the number of false tracks is one. The performance of this scheme mainly depends on the quality of the PHD outputs. The end user will benefit from the availability of the point state estimates as well as the individual target tracks.

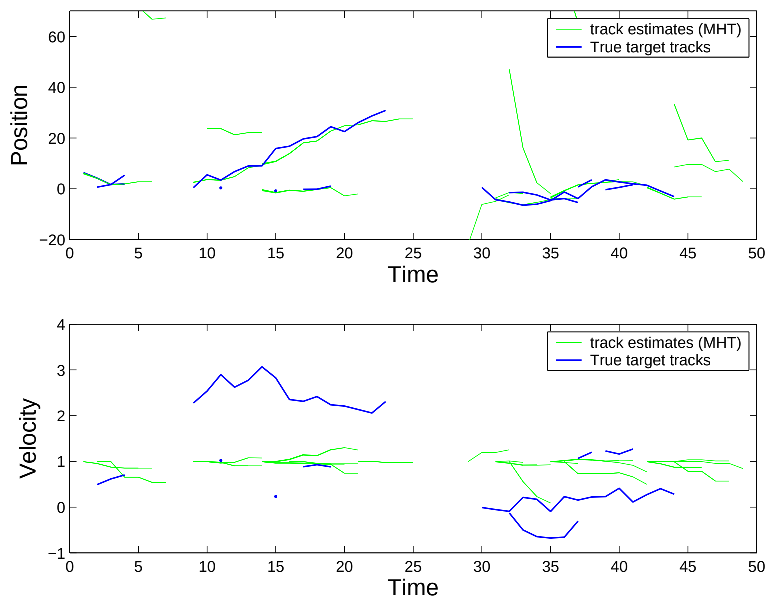

Figure 6. Target tracks obtained using the track-oriented MHT filter on observation sets.

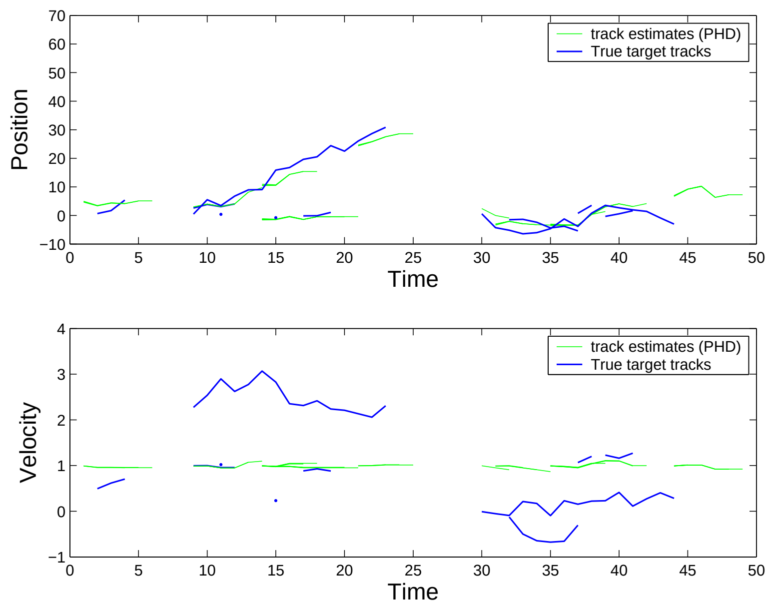

Figure 7. Target tracks obtained using the track-oriented MHT filter on observation sets modified by the PHD filter (Scheme One).

6. CONCLUSION

In this paper, we have presented the simulation results of multi-target filtering with the PHD and MHT filter for a linear, Gaussian target model. The Wasserstein distance of the points state estimates from the ground truth have been calculated for both filters. Comparison between the Wasserstein distances of two filter shows that the PHD filter outperforms the MHT filter. Our contribution also includes the proposal of a number of schemes for providing the data association functionality in the PHD filter. Target tracks are presented for the proposed filtering scheme proposed in Sect. 4.1. Various issues associated with the complexities and computational requirements of the proposed schemes and the optimal solution to initiate tracks with the PHD filter is the subject of the ongoing research.

APPENDIX A. A SMC IMPLEMENTATION OF THE PHD FILTER

Given a set of particles $\alpha_k = \{w_k^{(i)}, x_k^{(i)}\}_{i=1}^{L_k}$ representing the PHD $D_{k|k}$ for any $k \leq 0$, the PHD at time step $k > 0$ can be obtained using the PHD recursion as follows:

Particle PHD filter

At time $k \geq 1$,

Step 1: *Prediction step*

- For $i = 1, ..., L_{k-1}$, sample $\widetilde{x}_k^{(i)} \sim q_k\left(\cdot \middle| x_{k-1}^{(i)}, Z_k\right)$ and compute the predicted weights

$$\widetilde{w}_{k|k-1}^{(i)} = \frac{\phi_k(\widetilde{x}_k^{(i)}, x_{k-1}^{(i)})}{q_k\left(\widetilde{x}_k^{(i)} \middle| x_{k-1}^{(i)}, Z_k\right)} w_{k-1}^{(i)}.$$

- For $i = L_{k-1}+1, ..., L_{k-1}+J_k$, sample

$$\widetilde{x}_k^{(i)} \sim p_k\left(\cdot \middle| Z_k\right)$$

  and compute the weights of new born particles

$$\widetilde{w}_{k|k-1}^{(i)} = \frac{1}{J_k} \frac{\gamma_k\left(\widetilde{x}_k^{(i)}\right)}{p_k\left(\widetilde{x}_k^{(i)} \middle| Z_k\right)}.$$

Step 2: *Update step*

- For each $z \in Z_k$, compute

$$C_k(z) = \sum_{j=1}^{L_{k-1}+J_k} \psi_{k,z}(\widetilde{x}_k^{(j)}) \widetilde{w}_{k|k-1}^{(j)}.$$

- For $i = 1, ..., L_{k-1}+J_k$, update weights

$$\widetilde{w}_k^{(i)} = \left[\upsilon(\widetilde{x}_k^{(i)}) + \sum_{z \in Z_k} \frac{\psi_{k,z}(\widetilde{x}_k^{(i)})}{\kappa_k(z) + C_k(z)}\right] \widetilde{w}_{k|k-1}^{(i)}.$$

Step 3: *Resampling step*

- Compute the total mass $\widehat{N}_{k|k} = \sum_{j=1}^{L_{k-1}+J_k} \widetilde{w}_k^{(j)}$
- Resample $\left\{\widetilde{w}_k^{(i)} \widehat{N}_{k|k}, \widetilde{x}_k^{(i)}\right\}_{i=1}^{L_{k-1}+J_k}$ to get $\left\{w_k^{(i)} \widehat{N}_{k|k}, x_k^{(i)}\right\}_{i=1}^{L_k}$.

J_k represents the number of particles that are sampled from proposal density function related to the target birth process γ_k.

This SMC approximation of PHD filtering adaptively allocates the number of particles L_k to keep the number of particles per target constant. At each time step, a new L_k is obtained from $L_{k-1} + J_k$ so that L_k equals the number of particles per target times the expected number of targets. This in effect stops the SMC approximation from not having enough particles when there exist a large number of targets as well as minimizing the computational resources when only a small number of targets exists.

ACKNOWLEDGEMENTS

The first author would like to extend sincere gratitude to Dr. Mahendra Mallick for his guidance and suggestions for the implementation of the track-oriented MHT algorithm.

This work was supported by a large Australian Research Council (ARC) grant and Cooperative Research Centre for Sensor Signal and Information Processing (CSSIP), Department of Electrical & Electronics Engineering, The University of Melbourne, Australia.

REFERENCES

1. S. Blackman, *Multiple Target Tracking with Radar Applications*, Artech House, Norwood, 1986.
2. Y. Bar-Shalom and T. E. Fortmann, *Tracking and Data Association*, Academic Press, Boston, 1988.
3. I. Goodman, R. Mahler and H. Nguyen, *Mathematics of Data Fusion*, Kluwer Academic Publishers, Dordrecht/Boston/London, 1997.
4. R. Mahler, *An Introduction to Multisource-Multitarget Statistics and its Applications*, Lockheed Martin Technical Monograph, March 15, 2000.
5. R. Mahler, "Approximate Multisensor-Multitarget Joint Detection, Tracking and Indentification using a First Order Multitarget Moment Statistics," *IEEE Trans. on AES*, Vol. 39, No. 4, pp. 1152-1178, Oct. 2003.
6. B. Vo, S. Singh and A. Doucet, "Sequential Monte Carlo Implementation of the PHD Filter for Multi-Target Tracking," in *Proc. of Fusion'2003*, pp. 792-799, Cairns, Australia.
7. B. Vo, S. Singh and A. Doucet, "Random Finite Sets and Sequential Monte Carlo Methods in Multi-Target Tracking," in *Proc. of RADAR'2003*, pp. 486-491, Adelaide, Australia.
8. D. B. Reid, "An Algorithm for Tracking Multiple Targets," *IEEE Trans. on Automatic Control*, Vol. AC-24, pp. 843-854, Dec. 1979.
9. S. Blackman and R. Popoli, *D*esign and Analysis of Modern Tracking Systems, Artech House, Norwood, 1999.
10. T. Kurien, "Issues in the design of practical multitarget tracking algorithsm," in *M*ultitarget-Multisensor Tracking: Adavanced Applications, Y. Bar-Shalom, ed., pp. 43-83, Artech House, Norwood, 1990.
11. N. Gordon, "A Hybrid Bootstrap Filter for Target Tracking in Clutter," *I*EEE Trans. on Aerosp. Electron. Syst., Vol. 33, No. 3, pp. 353-358, Jan. 1997.
12. C. Hue, Jean-Pierre L Cadre and P. Perez, "Sequential Monte Carlo Methods for Multiple Target Tracking and Data Fusion," *IEEE Trans. on Signal Proc.*, Vol. 50, No. 2, pp. 309-325, Feb. 2002.
13. J. Hoffman and R. Mahler, "Multi-Target Miss Distance and Its Applications," in *P*roc. of Information Fusion'2002, pp. 149-155, Annapolis, Maryland, USA.