Kernel-Based Ensemble Gaussian Mixture Probability Hypothesis Density Filter

Dalton Durant
Dept. of Aerospace Engineering
and Engineering Mechanics
The University of Texas at Austin
Austin, TX, USA
ddurant@utexas.edu

Renato Zanetti
Dept. of Aerospace Engineering
and Engineering Mechanics
The University of Texas at Austin
Austin, TX, USA
renato@utexas.edu

Abstract—In this work, a kernel-based Ensemble Gaussian Mixture Probability Hypothesis Density (EnGM-PHD) filter is presented for multi-target filtering applications. The EnGM-PHD filter combines the Gaussian-mixture-based techniques of the Gaussian Mixture Probability Hypothesis Density (GM-PHD) filter with the particle-based techniques of the Sequential Monte Carlo Probability Hypothesis Density (SMC-PHD) filter. It achieves this by obtaining particles from the posterior intensity function, propagating them through the system dynamics, and then using Kernel Density Estimation (KDE) techniques to approximate the Gaussian mixture of the prior intensity function. This approach guarantees convergence to the true intensity function in the limit of the number of components. Moreover, in the special case of a single target with no births, deaths, clutter, and perfect detection probability, the EnGM-PHD filter reduces to the standard Ensemble Gaussian Mixture Filter (EnGMF). In the presented experiment, the results indicate that the EnGM-PHD filter achieves better multi-target filtering performance than both the GM-PHD and SMC-PHD filters while using the same number of components or particles.

Index Terms—multi-target tracking, multi-target filtering, state estimation, nonlinear filtering

I. Introduction

The Probability Hypothesis Density (PHD) filter [1] provides an efficient approximation of the intractable multi-target Bayesian filter by propagating only its first-order moment— *a.k.a* the intensity function. The intensity function describes the expected number of targets found in a unit volume of space, serving as a compact representation of the multi-target density. While it has reduced complexity compared to the full multi-target filter, there are unfortunately no closed form solutions of the PHD filter for nonlinear and non-Gaussian conditions.

Ref. [2] developed a Gaussian mixture implementation (GM-PHD), which approximates the intensity function as a weighted sum of Gaussian components, and [3] developed a sequential Monte Carlo implementation (SMC-PHD), which approximates the same intensity function as a weighted mixture of Dirac delta distributions using particles. The GM-PHD

filter provides an efficient closed-form solution under mild nonlinear conditions, but breaks down under larger nonlinearities [2], [4]. On the other hand, the SMC-PHD filter can handle larger nonlinearities, but at the cost of being more computationally expensive [3], [4].

Recent advances in single-target tracking (STT) have introduced kernel-based methods that merge the computational efficiency of Gaussian mixture filters with the nonlinear capabilities of particle filters. One such method, the Ensemble Gaussian Mixture Filter (EnGMF) [5], [6], uses Kernel Density Estimation (KDE) to transform a set of particles into a Gaussian mixture. In this approach, each particle is treated as a Gaussian component with nonzero covariance and equal weight, producing a smooth and accurate approximation of the target's probability density function (PDF). Although the EnGMF has demonstrated advantages in STT, no direct extension has been developed for multi-target tracking (MTT).

In the MTT literature, kernel-based approaches have seen limited application. Ref. [4] introduces a distributed cardinalized SMC-PHD filter that fuses particle sets from multiple nodes using KDE, while [7] applies the EnGMF as the single-target filter to the multi-target generalized labeled multi-Bernoulli (GLMB) filter. Despite these contributions, kernel-based approaches have not been developed to approximate the intensity function directly.

This work proposes the Ensemble Gaussian Mixture Probability Hypothesis Density (EnGM-PHD) filter, which uses the KDE approach of the EnGMF to produce a smooth and accurate approximation of the intensity function. The EnGM-PHD filter combines the Gaussian-mixture-based computational efficiency of the GM-PHD filter with the particle-based nonlinear capabilities of the SMC-PHD filter. It achieves this by obtaining particles from the posterior intensity function, propagating them through the system dynamics, and then using KDE techniques to approximate the Gaussian mixture of the prior intensity function.

The EnGM-PHD filter guarantees convergence to the true intensity function in the limit of the number of components. Moreover, in the special case of a single target with no births, deaths, clutter, and perfect detection probability, the EnGM-PHD filter simplifies to the standard EnGMF, making the

EnGM-PHD filter a straightforward MTT extension of the single-target EnGMF.

This work tests the GM-PHD, SMC-PHD, and EnGM-PHD filters in a MTT scenario with noisy measurements and considerable uncertainty in new target births. The scenario demonstrates the need to accurately approximate the intensity function in the presence of large measurement and model errors. While using the same number of components or particles, the EnGM-PHD filter demonstrates better multi-target filtering performance than both the GM-PHD and the SMC-PHD filters, indicating a more accurate approximation of the intensity function.

The remainder of this work is structured as follows: Section II provides background to the PHD, GM-PHD, and SMC-PHD filters. Section III details the methodology for the EnGM-PHD filter. Section IV compares the results of the GM-PHD, SMC-PHD, and EnGM-PHD filters in a simulated MTT scenario. Finally, Section V gives a brief discussion with concluding remarks.

II. BACKGROUND AND MOTIVATION

A. Probability Hypothesis Density Filter

The PHD filter [1] is designed to provide an efficient approximation of the intractable multi-target Bayesian filter. It is based on Random Finite Set (RFS) theory and avoids the combinatorial complexities of data association. It is able to handle target survivals, spawns, births, and deaths, all while estimating the time-varying number of targets—in real time.

The PHD filter accomplishes these features by only propagating the first-order moment of the full multi-target density, *a.k.a.* the intensity function. The intensity function, denoted by $v(\mathbf{x})$, serves as a compact representation of the multi-target density and describes the expected number of targets at state $\mathbf{x} \in \mathbb{R}^{n_{\mathbf{x}}}$, scattered randomly in space, according to a Poisson distribution. Its integral over any surveillance region $S \subseteq \mathbb{R}^{n_{\mathbf{x}}}$ gives the expected number of targets N in that region:

$$N = \int_S v(\mathbf{x}) \, \mathrm{d}\mathbf{x}, \tag{1}$$

where $\mathrm{d}\mathbf{x}$ is an infinitesimally small region of $\mathbf{x}$ in S.

The PHD filter assumes:

(i) targets evolve independently and generate independent measurements,

(ii) false alarms from clutter are Poisson distributed and are independent of measurements from actual targets,

(iii) the predicted and updated multi-target RFS are also Poisson distributed.

Given the above assumptions, the PHD filter's predicted intensity from time step $k-1$ to k can be written as

$$v(\mathbf{x}_k) = \int \Phi(\mathbf{x}_k, \mathbf{x}_{k-1}) v(\mathbf{x}_{k-1}) \mathrm{d}\mathbf{x}_{k-1} + v_\gamma(\mathbf{x}_k), \tag{2}$$

and

$$\begin{aligned} \Phi(\mathbf{x}_k, \mathbf{x}_{k-1}) &= p_S(\mathbf{x}_{k-1}) \phi(\mathbf{x}_k | \mathbf{x}_{k-1}) \\ &+ v_\beta(\mathbf{x}_k | \mathbf{x}_{k-1}), \end{aligned} \tag{3}$$

where $p_S(\mathbf{x}_{k-1})$ is the probability of survival, $\phi(\cdot | \mathbf{x}_{k-1})$ is the state transition probability density, $v_\gamma(\cdot)$ is the birth intensity, and $v_\beta(\cdot | \mathbf{x}_{k-1})$ is the spawning intensity.

Let the measurement set at k be denoted by $\mathbf{Z}_k \subset \mathbb{R}^{n_{\mathbf{z}}}$, which contains measurements and clutter. The intensity is then updated as

$$\begin{aligned} v(\mathbf{x}_k | \mathbf{Z}_k) &= \big(1 - p_D(\mathbf{x}_k)\big) v(\mathbf{x}_k) \\ &+ \sum_{\mathbf{z} \in \mathbf{Z}_k} \frac{p_D(\mathbf{x}_k) p(\mathbf{z} | \mathbf{x}_k) v(\mathbf{x}_k)}{\kappa(\mathbf{z}) + \int p_D(\boldsymbol{\xi}) p(\mathbf{z} | \boldsymbol{\xi}) v(\boldsymbol{\xi}) \mathrm{d}\boldsymbol{\xi}}, \end{aligned} \tag{4}$$

where $p_D(\mathbf{x}_k)$ is the probability of detection, $p(\cdot | \mathbf{x}_k)$ is the likelihood of a single-target measurement given $\mathbf{x}_k$, and $\kappa(\cdot)$ is the clutter intensity. The clutter intensity can be further defined as $\kappa(\cdot) := \lambda_C p_C(\cdot)$, where λ_C is the Poisson average rate of uniform clutter per scan, and $p_C(\cdot)$ is the clutter density.

While the PHD filter has reduced complexity compared to the full multi-target filter, there are unfortunately no closed form solutions to the integrals of the intensity function for nonlinear and non-Gaussian conditions. Therefore, [2] developed a Gaussian mixture implementation for approximate linear Gaussian conditions, and [3] developed a particle implementation for more nonlinear and non-Gaussian conditions.

B. Gaussian Mixture Implementation

The GM-PHD filter [2] approximates the intensity function as a weighted sum of Gaussian mixture components. The GM-PHD filter assumes (i)–(iii) and further assumes:

(iv) the dynamics and measurement models are linear,

(v) the survival and detection probabilities are independent of the target state,

(vi) the birth and spawning intensities are Gaussian mixtures.

Given the assumptions (i)–(vi), the GM-PHD approximates the PHD filter's prior intensity from (2) as

$$\begin{aligned} v(\mathbf{x}_k) &\approx \underbrace{\sum_{i=1}^{J_{k-1}} p_S w_{k-1}^{(i)} \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{S,k|k-1}^{(i)}, \hat{\mathbf{P}}_{S,k|k-1}^{(i)})}_{\text{survived components}} \\ &+ \underbrace{\sum_{i=1}^{J_{k-1}} \sum_{j=1}^{J_{\beta,k}} w_{k-1}^{(i)} w_{\beta,k}^{(j)} \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{\beta,k|k-1}^{(i,j)}, \hat{\mathbf{P}}_{\beta,k|k-1}^{(i,j)})}_{\text{spawned components}} \\ &+ \underbrace{\sum_{i=1}^{J_{\gamma,k}} w_{\gamma,k}^{(i)} \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{\gamma,k}^{(i)}, \hat{\mathbf{P}}_{\gamma,k}^{(i)})}_{\text{birthed components}} \\ &= \sum_{i=1}^{J_{k|k-1}} w_{k|k-1}^{(i)} \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k-1}^{(i)}, \hat{\mathbf{P}}_{k|k-1}^{(i)}), \end{aligned} \tag{5}$$

where it becomes a mixture of Gaussian mixtures of the survived, spawned, and birthed components. The hat notation $(\hat{\cdot})$ indicates estimates. For example, $\hat{\mathbf{x}}$ represents the estimated mean and $\hat{\mathbf{P}}$ the estimated covariance. The last line in (5) is to simplify the notation, where $J_{k|k-1} = J_{k-1}(1 + J_{\beta,k}) + J_{\gamma,k}$ is the total number of components.

Now, given the Gaussian mixture approximation made in (5), the GM-PHD filter approximates the posterior intensity as

$$
\begin{aligned}
v(\mathbf{x}_k|\mathbf{Z}_k) &\approx (1 - p_D)v(\mathbf{x}_k) \\
&+ \sum_{\mathbf{z}\in\mathbf{Z}_k}\sum_{i=1}^{J_{k|k-1}} w_{k|k}^{(i)}\mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k}^{(i)}, \hat{\mathbf{P}}_{k|k}^{(i)}).
\end{aligned}
\tag{6}
$$

Generally, for linear models, the means and covariances are updated individually following the update equations of the Kalman Filter (KF) [8].

However, for nonlinear models, assumption (iv) can be relaxed, leading to two extensions: EK-PHD and UK-PHD filters [2]. The EK-PHD filter uses the Extended Kalman Filter (EKF) [9] to individually predict and update components, while the UK-PHD filter uses the Unscented Kalman Filter (UKF) [10], [11].

After the update, the number of components grow combinatorially, which the GM-PHD filter manages through a pruning, merging, and capping scheme. Multi-target state extraction is performed by first estimating the cardinality $\hat{N}_k$ as the sum of the remaining weights, and then selecting the $\hat{N}_k$ components with the highest weights as the estimated target states. Alternatively, they can be chosen to be the components whose weights exceed a prescribed threshold. [2]

The GM-PHD filter provides a smooth and accurate approximation to the true intensity function and guarantees convergence, to some specified accuracy, using a minimum number of mixture components [12]. However, even though the GM-PHD filter is an efficient closed-form solution to the PHD filter for mildly nonlinear and non-Gaussian conditions, its multi-target filtering performance will break down under larger nonlinearities [2], [4].

C. Particle Implementation

The SMC-PHD filter [3] uses a mixture of weighted Dirac delta functions, via particles, rather than a mixture of Gaussians to accommodate more nonlinear and non-Gaussian conditions. The SMC-PHD filter only needs assumptions (i)–(iii) and approximates the PHD filter's prior intensity from (2) as

$$
\begin{aligned}
v(\mathbf{x}_k) &\approx \underbrace{\sum_{i=1}^{J_{k-1}} p_S(\boldsymbol{\mathcal{X}}_{k-1|k-1}^{(i)})w_{k-1}^{(i)}\delta(\mathbf{x}_k - \boldsymbol{\mathcal{X}}_{S,k|k-1}^{(i)})}_{\text{survived particles}} \\
&+ \underbrace{\sum_{i=1}^{J_{k-1}}\sum_{j=1}^{J_{\beta,k}} w_{k-1}^{(i)}w_{\beta,k}^{(j)}\delta(\mathbf{x}_k - \boldsymbol{\mathcal{X}}_{\beta,k|k-1}^{(i,j)})}_{\text{spawned particles}} \\
&+ \underbrace{\sum_{i=1}^{J_{\gamma,k}} w_{\gamma,k}^{(i)}\delta(\mathbf{x}_k - \boldsymbol{\mathcal{X}}_{\gamma,k}^{(i)})}_{\text{birthed particles}} \\
&= \sum_{i=1}^{J_k} w_{k|k-1}^{(i)}\delta(\mathbf{x}_k - \boldsymbol{\mathcal{X}}_k^{(i)}),
\end{aligned}
\tag{7}
$$

where $\delta(\cdot)$ represents the Dirac delta, which can be re-interpreted as the limit of a normal distribution with covariance matrix approaching zero. The notation $\boldsymbol{\mathcal{X}}$ represents particles that are assumed independent and identically distributed (i.i.d.).

Given the Dirac mixture approximation made in (7), the SMC-PHD filter approximates the posterior intensity as

$$
v(\mathbf{x}_k|\mathbf{Z}_k) \approx \sum_{i=1}^{J_k} w_{k|k}^{(i)}\delta(\mathbf{x}_k - \boldsymbol{\mathcal{X}}_k^{(i)}),
\tag{8}
$$

where

$$
\begin{aligned}
w_{k|k}^{(i)} &= \left(1 - p_D(\boldsymbol{\mathcal{X}}_k^{(i)})\right)w_{k|k-1}^{(i)} \\
&+ \sum_{\mathbf{z}\in\mathbf{Z}_k}\frac{p_D(\boldsymbol{\mathcal{X}}_k^{(i)})p(\mathbf{z}|\boldsymbol{\mathcal{X}}_k^{(i)})w_{k|k-1}^{(i)}}{\kappa(\mathbf{z}) + \sum_{j=1}^{J_k} p_D(\boldsymbol{\mathcal{X}}_k^{(j)})p(\mathbf{z}|\boldsymbol{\mathcal{X}}_k^{(j)})w_{k|k-1}^{(j)}}.
\end{aligned}
\tag{9}
$$

After the update, particles are resampled to prevent degeneracy. This process involves sampling from an RFS or point process. Once J_k particles are resampled, they are assigned uniform weights summing to the estimated cardinality $\hat{N}_k$. Similar to the GM-PHD filter, multi-target state extraction is performed by first estimating cardinality $\hat{N}_k$ as the sum of the weights. But, unlike the GM-PHD filter, the SMC-PHD filter obtains the estimated target states by using a clustering algorithm, generally either through k-means or expectation maximization (EM), to group the resampled particles into $\hat{N}_k$ clusters. [3]

Although it does not provide a smooth approximation, the SMC-PHD filter does guarantee convergence to the true posterior intensity function as its number of particles increases [13]. However, even though it can handle larger nonlinearities, the SMC-PHD filter is generally more computationally expensive than the GM-PHD filter because it requires more particles to obtain the same multi-target filtering performance [4]. In addition, its method of state extraction is less efficient as well [4].

It is therefore the motivation of this work to develop a filter that combines the Gaussian-mixture-based computational efficiency of the GM-PHD filter with the particle-based nonlinear capabilities of the SMC-PHD filter.

III. KERNEL-BASED ENSEMBLE GAUSSIAN MIXTURE IMPLEMENTATION

A. EnGM-PHD Filter Recursion

The following proposed EnGM-PHD filter only requires assumptions (i)–(iii) of the PHD filter, making it applicable for nonlinear and non-Gaussian conditions. Its basic idea is to transform the prior intensity of particles into a continuous, smooth and accurate approximation using KDE. This continuous approximation is represented by a Gaussian mixture, which is then updated using the GM-PHD filter equations.

Recall the Dirac mixture approximation of the prior intensity from (7):

$$
v(\mathbf{x}_k) \approx \sum_{i=1}^{J_k} w_{k|k-1}^{(i)}\delta(\mathbf{x}_k - \boldsymbol{\mathcal{X}}_k^{(i)}).
\tag{10}
$$

To convert (10) into a Gaussian mixture, the KDE method from [14] is used to transform each particle into a Gaussian component with nonzero covariance and equal weight:

$$v(\mathbf{x}_k) \approx \sum_{i=1}^{J_k} \frac{\hat{N}_k}{J_k} \mathcal{N}\big(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k-1}^{(i)}, \frac{\beta_{\text{Silv.}}}{\hat{N}_k}\text{Cov}(\{\hat{\mathbf{x}}_{k|k-1}^{(i)}\}_{i=1}^{J_k})\big), \tag{11}$$

$$\hat{N}_k = \sum_{i=1}^{J_k} w_{k|k-1}^{(i)}, \tag{12}$$

$$\beta_{\text{Silv.}} = \left(\frac{4}{n_{\mathbf{x}}+2}\right)^{\frac{2}{n_{\mathbf{x}}+4}} J_k^{-\frac{2}{n_{\mathbf{x}}+4}}, \tag{13}$$

where $\hat{\mathbf{x}}_{k|k-1}^{(i)} = \boldsymbol{\mathcal{X}}_k^{(i)}$ is the predicted mean of the i-th Gaussian component, $\text{Cov}(\{\hat{\mathbf{x}}_{k|k-1}^{(i)}\}_{i=1}^{J_k})$ is the sample covariance matrix of all the particles, and $\beta_{\text{Silv.}}$ is the bandwidth parameter using Silverman's Rule of Thumb [14].

For a single target, if the sampling distribution is Gaussian, Silverman's Rule of Thumb provides the optimal bandwidth parameter by minimizing the mean integrated square error (MISE) as a performance criterion [14]. However, for non-Gaussian distributions, it may produce conservative estimates.

A conservative estimate is often preferable because it helps maintain filter stability and prevents overconfidence, which could otherwise lead to divergence. Furthermore, Silverman's Rule of Thumb is relatively simple to compute, which reduces computational cost over alternative bandwidth selection algorithms.

The above approach guarantees convergence to the true intensity function in the limit of the number of components. It inherits the GM-PHD filter's convergence to the posterior intensity through its Gaussian mixture approximation [12] and it inherits the SMC-PHD filter's convergence to the prior intensity through its Dirac mixture approximation [13].

Furthermore, when transforming the prior intensity into a Gaussian mixture, the EnGM-PHD filter maintains convergence through KDE and Silverman's Rule of Thumb. This is because KDE and Silverman's Rule of Thumb can approximate any function arbitrarily well in the limit, producing a smoothed representation of the particle distribution, and ultimately converging to exact Bayesian inference [6], [14].

Conventionally, $\beta_{\text{Silv.}}$ remains unscaled in the single-target case. However, in the multi-target case, the bandwidth parameter must be adjusted by dividing it by the estimated cardinality $\hat{N}_k$. In (11), this scaling ensures that the covariance is appropriately applied to the estimated number of targets.

Given the Gaussian mixture approximation made in (11), the EnGM-PHD filter approximates the posterior intensity as

$$v(\mathbf{x}_k|\mathbf{Z}_k) \approx \big(1 - p_D\big)v(\mathbf{x}_k)$$
$$+ \sum_{\mathbf{z}\in\mathbf{Z}_k} \sum_{i=1}^{J_k} w_{k|k}^{(i)} \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k}^{(i)}, \hat{\mathbf{P}}_{k|k}^{(i)}), \tag{14}$$

and,

$$w_{k|k}^{(i)} = \frac{p_D w_{k|k-1}^{(i)} \boldsymbol{\xi}_k^{(i)}}{\kappa(\mathbf{z}) + \sum_{j=1}^{J_k} p_D w_{k|k-1}^{(j)} \boldsymbol{\xi}_k^{(j)}}, \tag{15}$$

$$\hat{\mathbf{x}}_{k|k}^{(i)} = \hat{\mathbf{x}}_{k|k-1}^{(i)} + \mathbf{K}_k^{(i)}(\mathbf{z} - h(\hat{\mathbf{x}}_{k|k-1}^{(i)})), \tag{16}$$

$$\hat{\mathbf{P}}_{k|k}^{(i)} = \hat{\mathbf{P}}_{k|k-1}^{(i)} - \mathbf{K}_k^{(i)}\mathbf{H}_k^{(i)}\hat{\mathbf{P}}_{k|k-1}^{(i)}, \tag{17}$$

$$\mathbf{K}_k^{(i)} = \hat{\mathbf{P}}_{k|k-1}^{(i)}(\mathbf{H}_k^{(i)})^T(\mathbf{S}_k^{(i)})^{-1}, \tag{18}$$

$$\mathbf{S}_k^{(i)} = \mathbf{H}_k^{(i)}\hat{\mathbf{P}}_{k|k-1}^{(i)}(\mathbf{H}_k^{(i)})^T + \mathbf{R}, \tag{19}$$

$$\hat{\mathbf{P}}_{k|k-1}^{(i)} = \frac{\beta_{\text{Silv.}}}{\hat{N}_k}\text{Cov}(\{\hat{\mathbf{x}}_{k|k-1}^{(i)}\}_{i=1}^{J_k}), \tag{20}$$

$$\mathbf{H}_k^{(i)} = \frac{\partial h(\mathbf{x})}{\partial\mathbf{x}}\Big|_{\mathbf{x}=\hat{\mathbf{x}}_{k|k-1}^{(i)}}, \tag{21}$$

$$\boldsymbol{\xi}_k^{(i)} = \mathcal{N}\left(\mathbf{z}; h(\hat{\mathbf{x}}_{k|k-1}^{(i)}), \mathbf{S}_k^{(i)}\right), \tag{22}$$

where $(\cdot)^T$ represents the matrix transpose and $\mathbf{R}$ is the measurement noise covariance.

After the update, the EnGM-PHD filter resamples J_k i.i.d. samples from the Gaussian mixture approximation of the posterior intensity following the steps in Algorithm I.

Algorithm I: Sampling from a Gaussian mixture.

Given: $\{w^{(i)}, \hat{\mathbf{x}}^{(i)}, \hat{\mathbf{P}}^{(i)}\}_{i=1}^{J}$ and requested samples J_{new}.
Check: weights are normalized and sum to 1.

For each j, where $j = 1, \ldots, J_{\text{new}}$:
① Draw a uniform random variable $u^{(j)} \sim \mathcal{U}(0,1)$.
② Select the smallest index ℓ: $\sum_{i=1}^{\ell} w^{(i)} \geq u^{(j)}$.
③ Draw a sample $\boldsymbol{\mathcal{X}}^{(j)} \sim \mathcal{N}(\mathbf{x}; \hat{\mathbf{x}}^{(\ell)}, \hat{\mathbf{P}}^{(\ell)})$.
Output: $\{\boldsymbol{\mathcal{X}}^{(j)}\}_{j=1}^{J_{\text{new}}}$

Similar to the SMC-PHD filter, once $J_{\text{new}} = J_k$ particles have been resampled, they are assigned uniform weights such that the sum of the weights is equal to the estimated cardinality $\hat{N}_k$. Multi-target state extraction is then performed following the same procedure as the SMC-PHD filter. This method of state extraction adds a layer of complexity beyond that of the GM-PHD filter, which is deemed the price to pay to extend the applicability of the GM-PHD filter to large nonlinearities.

B. Ensuring Uniform Weights in the Prior Intensity

While the proposed KDE approach seems straightforward, assuming the weights are uniform comes with some challenges. For example, a situation may arise where a prior intensity has the form:

$$v(\mathbf{x}_k) \approx \underbrace{\sum_{i=1}^{J_{k-1}} p_S(\boldsymbol{\mathcal{X}}_{k-1|k-1}^{(i)})w_{k-1}^{(i)}\delta(\mathbf{x}_k - \boldsymbol{\mathcal{X}}_{S,k|k-1}^{(i)})}_{\text{survived particles}}$$
$$+ \underbrace{\sum_{i=1}^{J_{\gamma,k}} w_{\gamma,k}^{(i)}\delta(\mathbf{x}_k - \boldsymbol{\mathcal{X}}_{\gamma,k}^{(i)})}_{\text{birthed particles}}. \tag{23}$$

The KDE step would be to construct a Gaussian mixture for each Dirac mixture:

$$v(\mathbf{x}_k) \approx$$

$$\underbrace{\sum_{i=1}^{J_{k-1}} \frac{\hat{N}_{S,k}}{J_{k-1}} \mathcal{N}\big(\mathbf{x}_k; \hat{\mathbf{x}}_{S,k|k-1}^{(i)}, \frac{\beta_{S,\text{Silv.}}}{\hat{N}_{S,k}} \text{Cov}(\{\hat{\mathbf{x}}_{S,k|k-1}^{(i)}\}_{i=1}^{J_{k-1}}))}_{\text{KDE constructed survived mixture}}$$

$$+ \underbrace{\sum_{i=1}^{J_{\gamma,k}} \frac{\hat{N}_{\gamma,k}}{J_{\gamma,k}} \mathcal{N}\big(\mathbf{x}_k; \hat{\mathbf{x}}_{\gamma,k}^{(i)}, \frac{\beta_{\gamma,\text{Silv.}}}{\hat{N}_{\gamma,k}} \text{Cov}(\{\hat{\mathbf{x}}_{\gamma,k}^{(i)}\}_{i=1}^{J_{\gamma,k}}))}_{\text{KDE constructed birthed mixture}}.$$

$$(24)$$

However, despite being transformed into Gaussian mixtures, the weights in the above equation remain non-uniform. This imbalance can lead to issues where the birth model overwhelms the surviving components, resulting in a poor representation of the prior intensity.

To mitigate this, i.i.d. samples are drawn from the combined distribution filter and used to reconstruct a single Gaussian mixture with uniform weights, ensuring a more balanced representation. This procedure is outlined in Algorithm II, resulting in the following prior intensity:

$$v(\mathbf{x}_k) \approx \sum_{i=1}^{J_k'} \frac{\hat{N}_k'}{J_k'} \mathcal{N}\big(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k-1}'^{(i)}, \frac{\beta_{\text{Silv.}}'}{\hat{N}_k'} \text{Cov}(\{\hat{\mathbf{x}}_{k|k-1}'^{(i)}\}_{i=1}^{J_k'})),$$

$$(25)$$

$$\hat{N}_k' = \hat{N}_k + \hat{N}_{\gamma,k}, \tag{26}$$

$$\beta_{\text{Silv.}}' = \left(\frac{4}{n_\mathbf{x}+2}\right)^{\frac{2}{n_\mathbf{x}+4}} J_k'^{-\frac{2}{n_\mathbf{x}+4}}, \tag{27}$$

where $J_k' = J_{k-1} + J_{\gamma,k}$ is the total number of components and $\hat{\mathbf{x}}_{k|k-1}'^{(i)} = \boldsymbol{\mathcal{X}}_k'^{(i)}$ are the new samples. The above Gaussian mixture ensures that the prior intensity has uniform weights and is now ready for the update step.

Algorithm II: Sampling from two Gaussian mixtures.

Given: $\{w_1^{(i)}, \hat{\mathbf{x}}_1^{(i)}, \hat{\mathbf{P}}_1^{(i)}\}_{i=1}^{J_1}$, $\{w_2^{(i)}, \hat{\mathbf{x}}_2^{(i)}, \hat{\mathbf{P}}_2^{(i)}\}_{i=1}^{J_2}$, and requested samples J_{new}.
Precompute: $W_1 = \sum_{i=1}^{J_1} w_1^{(i)}$, $W_2 = \sum_{i=1}^{J_2} w_2^{(i)}$, and $Z = W_1 + W_2$.
For each j, where $j = 1, \ldots, J_{\text{new}}$:
① Draw a uniform random variable $u^{(j)} \sim \mathcal{U}(0,1)$.
② If $u^{(j)} < \frac{W_1}{Z}$, then select first mixture. Otherwise, select the second.
③ Within the selected mixture, use Algorithm I to draw a sample $\boldsymbol{\mathcal{X}}^{(j)}$.
Output: $\{\boldsymbol{\mathcal{X}}^{(j)}\}_{j=1}^{J_{\text{new}}}$

C. Relation to the Ensemble Gaussian Mixture Filter

Theorem III.1. *(Reduction of the EnGM-PHD filter to the EnGMF) In the special case of a single target with no births, deaths, clutter, and perfect detection probability, the EnGM-PHD filter equations simplify exactly to those of the standard EnGMF from [6].*

Proof. If there is a single target, then the estimated cardinality is $\hat{N} = 1$, and the probability of survival is assumed to be perfect. Moreover, in the absence of target births, the prior intensity reduces to a prior PDF:

$$v(\mathbf{x}_k) \approx \sum_{i=1}^{J_k} \frac{1}{J_k} \mathcal{N}\big(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k-1}^{(i)}, \beta_{\text{Silv.}} \text{Cov}(\{\hat{\mathbf{x}}_{k|k-1}^{(i)}\}_{i=1}^{J_k}))$$
$$= p(\mathbf{x}_k). \tag{28}$$

Here, the bandwidth parameter $\beta_{\text{Silv.}}$ is given by

$$\beta_{\text{Silv.}} = \left(\frac{4}{n_\mathbf{x}+2}\right)^{\frac{2}{n_\mathbf{x}+4}} J_k^{-\frac{2}{n_\mathbf{x}+4}}. \tag{29}$$

Furthermore, in the absence of deaths, clutter, and with perfect detection probability, the posterior intensity simplifies to a posterior PDF:

$$v(\mathbf{x}_k|\mathbf{z}_k) \approx \sum_{i=1}^{J_k} w_{k|k}^{(i)} \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k}^{(i)}, \hat{\mathbf{P}}_{k|k}^{(i)})$$
$$= p(\mathbf{x}_k|\mathbf{z}_k). \tag{30}$$

The corresponding update equations are

$$w_{k|k}^{(i)} = \frac{w_{k|k-1}^{(i)} \boldsymbol{\xi}_k^{(i)}}{\sum_{j=1}^{J_k} w_{k|k-1}^{(j)} \boldsymbol{\xi}_k^{(j)}}, \tag{31}$$

$$\hat{\mathbf{x}}_{k|k}^{(i)} = \hat{\mathbf{x}}_{k|k-1}^{(i)} + \mathbf{K}_k^{(i)}(\mathbf{z}_k - h(\hat{\mathbf{x}}_{k|k-1}^{(i)})), \tag{32}$$

$$\hat{\mathbf{P}}_{k|k}^{(i)} = \hat{\mathbf{P}}_{k|k-1}^{(i)} - \mathbf{K}_k^{(i)} \mathbf{H}_k^{(i)} \hat{\mathbf{P}}_{k|k-1}^{(i)}, \tag{33}$$

$$\mathbf{K}_k^{(i)} = \hat{\mathbf{P}}_{k|k-1}^{(i)} (\mathbf{H}_k^{(i)})^T (\mathbf{S}_k^{(i)})^{-1}, \tag{34}$$

$$\mathbf{S}_k^{(i)} = \mathbf{H}_k^{(i)} \hat{\mathbf{P}}_{k|k-1}^{(i)} (\mathbf{H}_k^{(i)})^T + \mathbf{R}, \tag{35}$$

$$\hat{\mathbf{P}}_{k|k-1}^{(i)} = \beta_{\text{Silv.}} \text{Cov}(\{\hat{\mathbf{x}}_{k|k-1}^{(i)}\}_{i=1}^{J_k}), \tag{36}$$

$$\mathbf{H}_k^{(i)} = \frac{\partial h(\mathbf{x})}{\partial \mathbf{x}}\Big|_{\mathbf{x}=\hat{\mathbf{x}}_{k|k-1}^{(i)}}, \tag{37}$$

$$\boldsymbol{\xi}_k^{(i)} = \mathcal{N}\left(\mathbf{z}; h(\hat{\mathbf{x}}_{k|k-1}^{(i)}), \mathbf{S}_k^{(i)}\right). \tag{38}$$

Equations (28)–(38) are precisely the prior and posterior PDFs of the EnGMF. Additionally, the resampling procedure in Algorithm I is identical to that in the EnGMF. Therefore, under these conditions, the EnGM-PHD filter recursion reduces exactly to the EnGMF. □

IV. NUMERICAL EXPERIMENT

This experiment tests the GM-PHD, SMC-PHD, and EnGM-PHD filters in a scenario that is concerned with tracking two targets. Both targets travel at constant velocities, and cross at the exact same point in time and space. They are being tracked by a radar sensor, placed at the origin, which produces noisy range, azimuth, and elevation measurements, recorded at a rate of $\mathrm{d}t$. The targets remain trackable at all times.

A. Model Parameters

The state vector is six-dimensional:

$$\mathbf{x} = [\mathbf{r}_x, \mathbf{r}_y, \mathbf{r}_z, \mathbf{v}_x, \mathbf{v}_y, \mathbf{v}_z]^T, \tag{39}$$

where $\mathbf{r}$ and $\mathbf{v}$ represent the 3-dimensional Cartesian positions and velocities relative to the origin. The motion model assumes constant velocity, integrated using a Runge-Kutta 8(7) method [15].

The true initial states of the two targets are

$$\mathbf{x}_{0,1} = [50, \ 50, \ 50, \ 0.5, \ 0.5, \ 2]^T, \tag{40}$$

$$\mathbf{x}_{0,2} = [100, \ 100, \ 50, \ -0.5, \ -0.5, \ 2]^T. \tag{41}$$

Whereas, the filters are initialized with a single low-weight component to favor births:

$$w_{0|0}^{(1)} = 1 \times 10^{-16}, \ \hat{\mathbf{x}}_{0|0}^{(1)} = \mathbf{0}_{6 \times 1}, \ \hat{\mathbf{P}}_{0|0}^{(1)} = \mathbf{I}_{6 \times 6}, \tag{42}$$

where $\mathbf{0}$ represents the zero vector and $\mathbf{I}$ represents the identity matrix.

The measurement vector $\mathbf{z} = [\rho, \ \alpha, \ \varepsilon]^T$ contains range ρ, azimuth α, and elevation ε of the observed target relative to the radar. Measurements are corrupted by additive zero-mean Gaussian white noise with standard deviations of one[1] for range and $0.5°$ for both azimuth and elevation. Sampling occurs at $\mathrm{d}t = 1$, starting from 0 and ending at 100.

Let the clutter generated state be $\mathbf{r}_c = [\mathbf{r}_{x,c}, \mathbf{r}_{y,c}, \mathbf{r}_{z,c}]^T$, where it is Poisson distributed in the surveillance region:

$$\mathrm{S} = \{ (\mathbf{r}_{x,c}, \mathbf{r}_{y,c}, \mathbf{r}_{z,c}) \ | 0 \leqslant \mathbf{r}_{x,c} \leqslant 200, \ 0 \leqslant \mathbf{r}_{y,c} \leqslant 200,$$
$$0 \leqslant \mathbf{r}_{z,c} \leqslant 400 \}, \tag{43}$$

with an average rate of $\lambda_C = 10$ and density $p_C = 6.25 \times 10^{-8}$. After being generated, clutter is mapped from the state space into measurements: $\mathbf{z} = h(\mathbf{r}_c)$.

The probability of detection and survival are constant such that $p_D = 0.98$ and $p_S = 0.99$, respectively. This work does not perform measurement gating, please see [16].

A Gaussian birth model is centered at the crossing point with:

$$\hat{\mathbf{x}}_{\gamma,k} = [75, \ 75, \ 150, \ 0, \ 0, \ 0]^T, \tag{44}$$

$$\hat{\mathbf{P}}_{\gamma,k} = \mathrm{diag}([50, \ 50, \ 50, \ 5, \ 5, \ 5]^2), \tag{45}$$

where after each prediction step, $J_{\gamma,k} = 10$ components are sampled with weights $w_{\gamma,k} = 1/100$. This model suggests that prior knowledge is given to the system that the two targets will cross.

In this experiment, the EK-PHD filter is used to represent the GM-PHD filter, but will still be referred to as the GM-PHD filter. Its pruning, merging, and capping parameters are 1×10^{-5}, 4, and $J_{\max} = 250$, respectively. Although the SMC-PHD and EnGM-PHD filters retain a fixed number of components or particles, with $J_k = 250$, the GM-PHD filter is allowed to fluctuate its number of components because of its pruning, merging, and capping scheme.

The SMC-PHD and EnGM-PHD filters both use k-means clustering [3].

---

[1] All units are dimensionless.

B. Multi-Target Performance Metric

The Optimal Subpattern Assignment (OSPA) [17] metric evaluates tracking accuracy by measuring localization and cardinality errors between estimated states $\hat{\mathbf{x}}$ and ground truth $\mathbf{x}$. In this work, the OSPA error metric uses the Euclidean 2-norm, with parameters $p = 2$ and cutoff $c = 100$. Optimal assignments are performed using the Hungarian, *a.k.a.* Munkres, algorithm [18], [19]. More details on OSPA can be found in [2], [4], and [17].

C. Results

Fig. 1 visualizes the extracted position state estimates, while Fig. 2 shows the extracted cardinality estimates. Both figures show the results of 250 Monte Carlo simulations as transparent data points with a single run overlaid without transparency for emphasis.

The GM-PHD filter struggles with the large amount of clutter and overwhelming birth model. This is likely due to its pruning, merging, and capping scheme, which can lead to the loss of information, impacting the filter's ability to accurately approximate the true intensity function. This is apparent in Fig. 1(a) by the red cloud of extracted state estimates that fill the surveillance region. Supporting this reasoning, Fig. 2(a) shows it failing to estimate the correct number of targets.

The SMC-PHD filter also struggles, even with 250 particles. Figs. 1(b) and 2(b) show that the filter fails to produce position estimates and underestimates the cardinality, indicating that the SMC-PHD filter's Dirac mixture approximation is insufficient.

The EnGM-PHD filter has the best performance. It shows to favorably track the two targets in Fig. 1(c) with minimal outliers and consistent cardinality estimates in Fig. 2(c), indicating better approximation of the intensity function.

Next, Fig. 3 presents the OSPA accuracy of the extracted state estimates in comparison to the truth. Corroborating the previous reasoning, the EnGM-PHD filter shows the best overall OSPA, again indicating a better approximation of the intensity function while using the same amount of components or particles.

To evaluate the algorithms relative computational costs, Fig. 4 compares how each filter's average number of components relate to their simulation wall-clock times. It should be noted that while the GM-PHD filter is typically more efficient than the SMC-PHD, this is not a guarantee, suggesting that inefficiencies can arise from other processes. Fig. 4 indicates this phenomena. This is likely due to the GM-PHD filter's merging process, which involves a combinatorial search to identify merging pairs which is computationally expensive and can make the GM-PHD filter slower than the SMC-PHD filter.

Nonetheless, the EnGM-PHD filter runs faster than the GM-PHD filter, even when using k-means clustering, demonstrating improved efficiency. Overall, the proposed EnGM-PHD filter achieves better multi-target filtering performance than both the GM-PHD and SMC-PHD filters while maintaining a similar number of components.

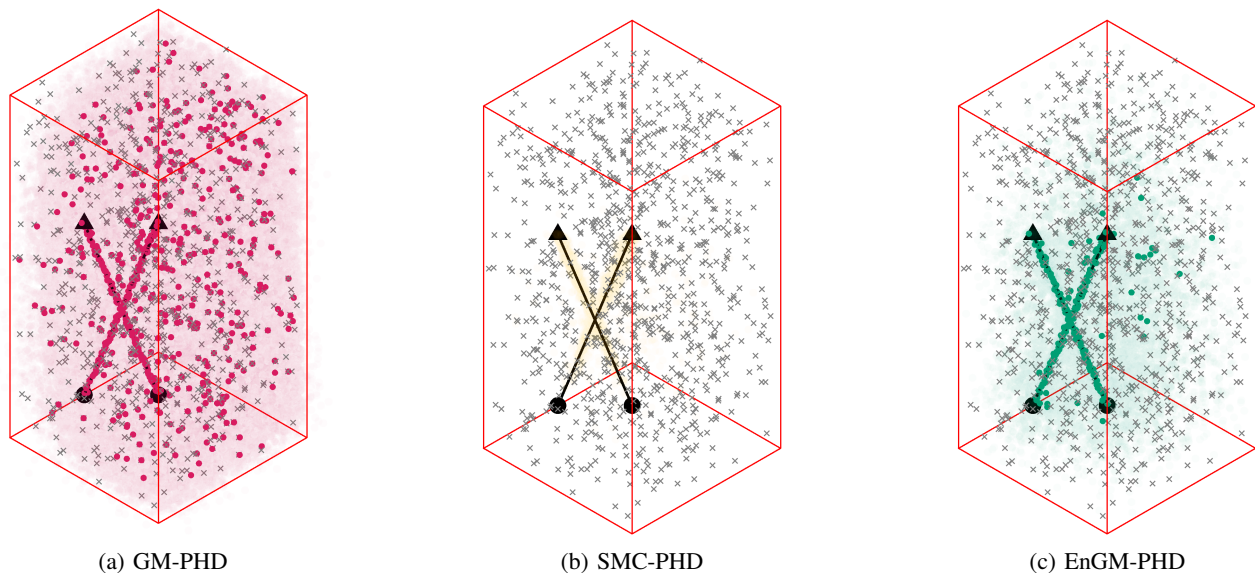| (a) GM-PHD | (b) SMC-PHD | (c) EnGM-PHD |

Fig. 1: This figure shows the 3-dimensional true trajectories (black) of the two targets and the extracted position state estimates of the compared filters. The clutter is represented by the gray crosses which are Poisson distributed in the red rectangular surveillance region. These are the results of 250 Monte Carlo simulations, overlaid with a single run emphasized to illustrate.

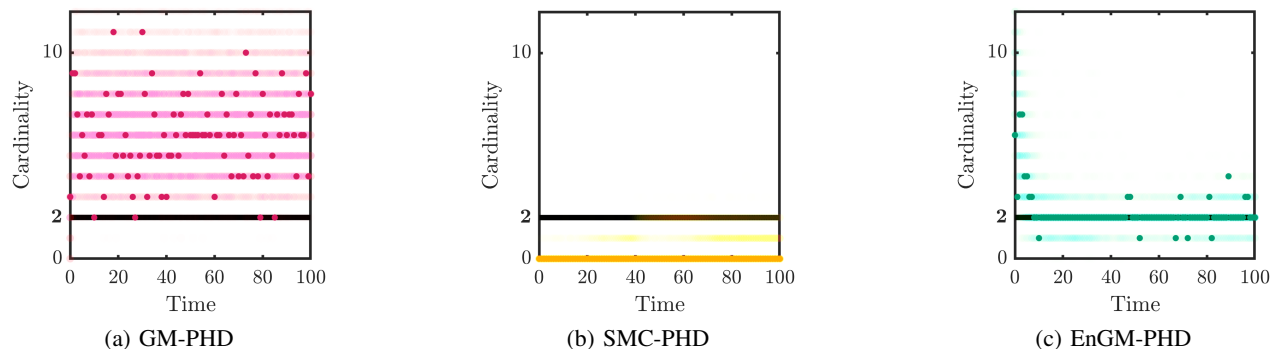| (a) GM-PHD | (b) SMC-PHD | (c) EnGM-PHD |

Fig. 2: This figure shows the true cardinality (black) and the extracted cardinality estimates of the compared filters vs. time. These are the results of 250 Monte Carlo simulations, overlaid with a single run emphasized to illustrate.

V. CONCLUSIONS

This work proposes the Ensemble Gaussian Mixture Probability Hypothesis Density (EnGM-PHD) filter to produce a smooth and accurate approximation of the intensity function. The EnGM-PHD filter combines the Gaussian-mixture-based computational efficiency of the Gaussian Mixture Probability Hypothesis Density (GM-PHD) filter with the particle-based nonlinear capabilities of the Sequential Monte Carlo Probability Hypothesis Density (SMC-PHD) filter. The algorithm obtains particles from the posterior intensity function, propagates them through the system dynamics, and then uses Kernel Density Estimation (KDE) techniques to approximate the Gaussian mixture of the prior intensity function.

This approach guarantees convergence to the true intensity function in the limit of the number of components. Moreover, in the special case of a single target with no births, deaths, clutter, and perfect detection probability, this work proves that the EnGM-PHD filter equations reduce exactly to those of

the Ensemble Gaussian Mixture Filter (EnGMF), making the EnGM-PHD filter a straightforward multi-target extension of the single-target EnGMF.

Furthermore, the presented numerical experiment demonstrates better multi-target filtering performance than both the GM-PHD and the SMC-PHD filters, while using the same number of components or particles. The EnGM-PHD filter shows promising simulation wall-clock times, but its state extraction method adds a layer of inefficiency and is a limitation of this work.

REFERENCES

[1] R. P. Mahler, "Multitarget Bayes filtering via first-order multitarget moments," *IEEE Transactions on Aerospace and Electronic systems*, vol. 39, no. 4, pp. 1152–1178, 2003.
[2] B.-N. Vo and W.-K. Ma, "The Gaussian mixture probability hypothesis density filter," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4091–4104, 2006.
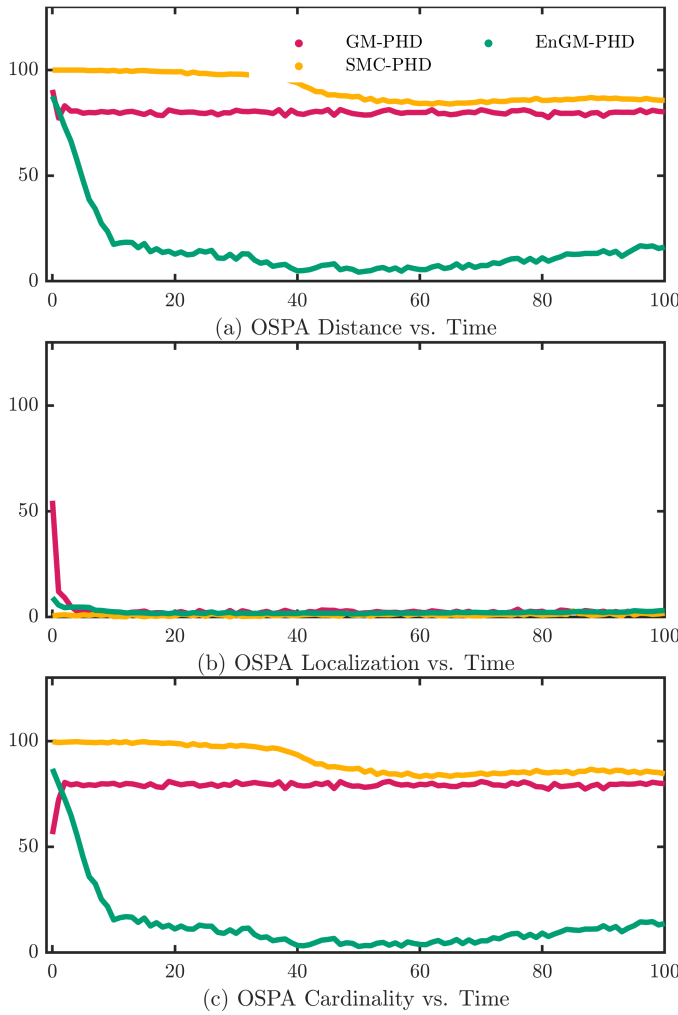
(a) OSPA Distance vs. Time

(b) OSPA Localization vs. Time

(c) OSPA Cardinality vs. Time

Fig. 3: This figure shows the multi-target accuracy (OSPA) of each filter vs. time. Results are averaged over 250 Monte Carlo simulations.

(a) Average number of components
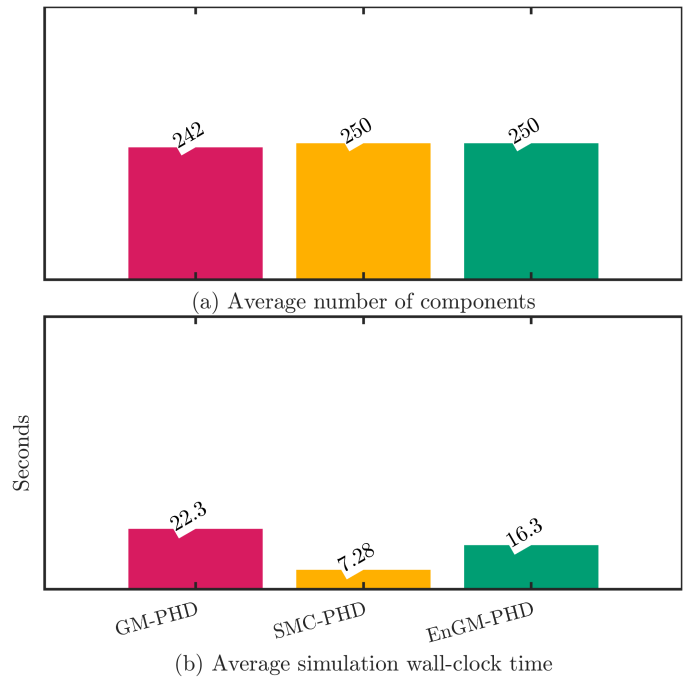

(b) Average simulation wall-clock time

Fig. 4: This figure compares the efficiency of the filters and how their number of components relate to their wall-clock times. Results are averaged over 250 Monte Carlo simulations. Using an Intel Core i7 9700K CPU at a base speed of 3.00 GHz and with 16 GB of RAM.

School of Science and Engineering at Oregon Health and Science University, 2004.

[12] D. Clark and B.-N. Vo, "Convergence analysis of the Gaussian mixture PHD filter," *IEEE Transactions on Signal Processing*, vol. 55, no. 4, pp. 1204–1212, 2007.

[13] A. M. Johansen, S. S. Singh, A. Doucet, and B.-N. Vo, "Convergence of the SMC implementation of the PHD filter," *Methodology and Computing in Applied Probability*, vol. 8, no. 2, pp. 265–291, 2006.

[14] B. W. Silverman, *Density Estimation for Statistics and Data Analysis*. Routledge, 2018.

[15] J. R. Dormand and P. J. Prince, "A family of embedded Runge-Kutta formulae," *Journal of Computational and Applied Mathematics*, vol. 6, no. 1, pp. 19–26, 1980.

[16] B.-N. Vo, M. Mallick, Y. Bar-Shalom, S. Coraluppi, R. Osborne, R. Mahler, and B.-t. Vo, "Multitarget tracking," *Wiley Encyclopedia of Electrical and Electronics Engineering*, 2015.

[17] D. Schuhmacher, B.-T. Vo, and B.-N. Vo, "A consistent metric for performance evaluation of multi-object filters," *IEEE Transactions on Signal Processing*, vol. 56, no. 8, pp. 3447–3457, 2008.

[18] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.

[19] J. Munkres, "Algorithms for the assignment and transportation problems," *Society for Industrial and Applied Mathematics*, vol. 15, pp. 196–210, 1962.

[3] B.-N. Vo, S. Singh, and A. Doucet, "Sequential Monte Carlo methods for multitarget filtering with random finite sets," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 4, pp. 1224–1245, 2005.

[4] W. Wu, H. Sun, M. Zheng, and W. Huang, *Target Tracking with Random Finite Sets*. Springer, 2023.

[5] J. L. Anderson and S. L. Anderson, "A Monte Carlo implementation of the nonlinear filtering problem to produce ensemble assimilations and forecasts," *Monthly Weather Review*, vol. 127, no. 12, pp. 2741–2758, 1999.

[6] S. Yun, R. Zanetti, and B. A. Jones, "Kernel-based ensemble Gaussian mixture filtering for orbit determination with sparse data," *Advances in Space Research*, vol. 69, no. 12, pp. 4179–4197, 2022.

[7] B. L. Reifler, S. Yun, B. A. Jones, and R. Zanetti, "Multi-target ensemble Gaussian mixture tracking with sparse observations," in *AMOS Conf. Proc*, 2021.

[8] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[9] A. Gelb, *Applied Optimal Estimation*. MIT Press, 1974.

[10] S. J. Julier and J. K. Uhlmann, "New extension of the Kalman filter to nonlinear systems," in *Signal Processing, Sensor Fusion, and Target Recognition VI*, vol. 3068. Spie, 1997, pp. 182–193.

[11] R. van der Merwe, "Sigma-point Kalman filters for probabilistic inference in dynamic state-space models," Ph.D. dissertation, The OGI