(Preprint) AAS 25-187

OPTIMAL GRID POINT SAMPLING FOR POINT MASS FILTERING

Felipe Giraldo-Grueso[*], Andrey A. Popov[†], Uwe D. Hanebeck[‡], Renato Zanetti[§]

The point mass filter (PMF) is a commonly used strategy for solving the state estimation problem. In this filter, a deterministic grid of point particles is used to numerically solve the Bayesian recursive relations. Since the probability density function of the state is only approximated at the grid points, the placement of these points is crucial for the performance of the filter. A new strategy, named the Silverman mass filter (SMF), has recently been developed, in which a Gaussian sum filter (GSF) update is performed before placing a new grid. In the SMF, the grid is placed at the mean of the GSF-updated points and is oriented and expanded to match their covariance. While the SMF has been shown to improve upon the standard PMF, the grid placement in the SMF can be suboptimal for highly non-Gaussian problems. This work introduces two new techniques for grid placement in the SMF: a clustering-based approach and an optimal deterministic sampling technique. These new techniques are shown to improve grid placement in the SMF when applied to a bimodal distribution example. Furthermore, the techniques are evaluated in a sequential filtering problem using the Ikeda map, where they show improved performance over the standard SMF by providing a more consistent and accurate state estimate.

INTRODUCTION

The state estimation problem for discrete-time continuous-amplitude systems involves solving the Bayesian recursive relations (BRRs).[1] The BRRs first describe the time evolution of a state probability density function (pdf) using the Chapman-Kolmogorov equation. After the pdf has been propagated and a measurement obtained, the pdf is updated using Bayes' rule. For linear dynamics and measurement models, these recursive relations can be solved analytically. However, when either the dynamics or the measurement models are nonlinear, the solution can become intractable.[1] Since nonlinear dynamics and measurement models are of interest for aerospace applications, various alternatives must be explored to numerically approximate the solution to the BRRs.

Point mass filters (PMF) are a common strategy for numerically solving the BRRs.[2] In this type of filter, the state pdf is discretized using a *deterministic* grid, where each grid point is assigned a finite probability. In the PMF, the BRRs are numerically solved by first propagating an initial grid, representing the previous posterior pdf, using the system dynamics. A new grid is then placed at the mean and covariance of the approximated prior distribution, calculated using the propagated grid points. Once a new measurement is obtained, the grid is appropriately weighted using Bayes'

[*]Ph.D. Student, Department of Aerospace Engineering and Engineering Mechanics, The University of Texas at Austin, Austin, TX 78712.

[†]Assistant Professor, Department of Information and Computer Sciences, The University of Hawai'i at Mānoa, Honolulu, HI 96822.

[‡]Professor, Department of Computer Science, Karlsruhe Institute of Technology, Germany.

[§]Associate Professor, Department of Aerospace Engineering and Engineering Mechanics, The University of Texas at Austin, Austin, TX 78712.

rule. Since the PMF approximates the state pdf at these discrete points, placement of the grid can severely affect the estimation performance. Consequently, much of the research on this type of filter has focused on finding the optimal placement of the grid to enhance estimation performance.[3–8]

A new variant of the PMF, called the Silverman mass filter (SMF), has recently been proposed.[9] In this new filter, the propagated points from the initial grid are represented as a Gaussian mixture, enabling a Gaussian sum filter (GSF) update before constructing the new grid. Performing a GSF update leads to a more precise estimation of the mean and covariance of the posterior pdf, which can then be used to place the new grid closer to the true posterior distribution. This approach is similar to the techniques used in the ensemble Gaussian mixture filter (EnGMF),[10–13] but has the advantages of using deterministic points.

If the posterior pdf is approximately Gaussian, placing a new grid at the updated mean and covariance from the GSF-updated points (as done in the SMF) can be an improvement over placing a grid at the approximated prior distribution (as done in the PMF). However, for non-Gaussian distributions, using only the mean and covariance of the approximated posterior distribution can potentially degrade estimation performance. Since the points from the GSF update each carry their own mean, covariance, and probability mass, more efficient approaches can use the full information from these updated points to find higher-quality grids.

This work presents two different strategies for generating the new grid in the SMF from the GSF-updated points. In the first strategy, a density-based clustering algorithm is used to cluster the updated points. Once clustered, the individual mean and covariance of each cluster are calculated. A new grid is then placed on top of each cluster, improving the placement of the deterministic points and resulting in an overall grid that better describes the posterior distribution. Given that the clustering algorithm used is purely density-based, a second strategy is presented that considers the probability masses of the updated points. In this strategy, numerous deterministic points are sampled from each Gaussian component describing the approximated posterior pdf, which are then optimally reduced to obtain a higher-quality grid.

Throughout this work, a 'dual banana' distribution[14] is used to illustrate the limitations of the standard grid in the SMF and the advantages of the two proposed techniques for grid construction. For this example, the prior distribution is assumed to be Gaussian and centered at the origin, with a range measurement also assumed to be obtained from the origin. This setup results in a bimodal posterior distribution that resembles the shape of two bananas. Although the standard grid in the SMF fails to capture the true posterior distribution, the new grids obtained using the two new approaches result in a better representation of the posterior pdf. Additionally, the performance of the standard SMF and its two new variants is evaluated using a sequential filtering problem involving nonlinear and chaotic dynamics, as described by the Ikeda map. Both new techniques demonstrate better estimation performance and consistency compared to the standard SMF.

The remainder of this paper is organized as follows. First, a brief introduction to state estimation for discrete-time, continuous-amplitude systems is provided. Following that, the Silverman mass filter is introduced. The 'dual banana' distribution is then used to demonstrate the limitations in grid construction. Next, the clustering-based approach and the optimal point sampling techniques are presented. Both techniques are evaluated using the 'dual banana' distribution. The proposed techniques are then compared in a sequential filtering example using the Ikeda map. Finally, conclusions are drawn.

STATE ESTIMATION FOR DISCRETE-TIME CONTINUOUS-AMPLITUDE SYSTEMS

The solution to the BRRs are the foundation of the state estimation problem. In this section, these relations are presented and discussed in the case of discrete-time continuous-amplitude systems, as they are fundamental for the derivation of the SMF. First, let $\boldsymbol{x}_k$ be the state at time step k, following discrete dynamics $\boldsymbol{f}_k$ and measurements $\boldsymbol{h}_k$, such that

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}_k\left(\boldsymbol{x}_k, \boldsymbol{q}_k\right), \tag{1}$$

$$\boldsymbol{y}_k = \boldsymbol{h}_k\left(\boldsymbol{x}_k, \boldsymbol{\eta}_k\right), \tag{2}$$

where $\boldsymbol{y}_k$ is the associated measurement, $\boldsymbol{q}_k$ is process noise and $\boldsymbol{\eta}_k$ is measurement noise.

Starting from an initial pdf of $\boldsymbol{x}_k$, the Chapman-Kolmogorov equation is used first to propagate this pdf to the time step $k+1$,[1]

$$p\left(\boldsymbol{x}_{k+1}|\boldsymbol{y}_k\right) = \int\limits_{\mathcal{S}(\boldsymbol{x}_k)} p\left(\boldsymbol{x}_{k+1}|\boldsymbol{x}_k\right) p\left(\boldsymbol{x}_k|\boldsymbol{y}_k\right)\, d\boldsymbol{x}_k, \tag{3}$$

where $\mathcal{S}\left(\boldsymbol{x}_k\right)$ denotes the support of $\boldsymbol{x}_k$. When a measurement is obtained ($k \leftarrow k+1$), the pdf is then updated using Bayes' rule,[1]

$$p\left(\boldsymbol{x}_k|\boldsymbol{y}_k\right) = \frac{p\left(\boldsymbol{y}_k|\boldsymbol{x}_k, \boldsymbol{y}_{k-1}\right) p\left(\boldsymbol{x}_k|\boldsymbol{y}_{k-1}\right)}{p\left(\boldsymbol{y}_k|\boldsymbol{y}_{k-1}\right)}. \tag{4}$$

Therefore, the Chapman-Kolmogorov equation is used to propagate the pdf, and Bayes' rule is used to update the pdf, together forming the BRRs. In the state estimation problem, these equations are solved recursively until a desired time. However, as noted earlier, the solution can become intractable when the dynamics and measurement models are nonlinear.

POINT MASS FILTERING

The PMF is a filtering strategy that is used to solve the state estimation problem deterministically using a structured grid of points. This section provides a summary of the SMF, a new variant of the standard PMF, which uses Gaussian mixtures and kernel density estimation techniques to achieve better estimation performance. This summary is included in order for this work to be self-contained, but the readers are encouraged to refer to the original paper for a more detailed explanation.[9] This approach begins by approximating the initial posterior state pdf with a regular grid of point particles,

$$p\left(\boldsymbol{x}_k|\boldsymbol{y}_k\right) \approx \sum_{i=1}^{N} w_{k|k}^{(i)}\, \delta\left(\boldsymbol{x}_k - \boldsymbol{\mathcal{X}}_{k|k}^{(i)}\right), \tag{5}$$

where $\boldsymbol{\mathcal{X}}_{k|k}^{(i)}$ are the posterior discretization points, $w_{k|k}^{(i)}$ represents the probability mass of each point, and N is the total number of points.

By describing the initial Dirac mixture as a Gaussian mixture with infinitesimal covariance,

$$p\left(\boldsymbol{x}_k|\boldsymbol{y}_k\right) \approx \sum_{i=1}^{N} w_{k|k}^{(i)} \lim_{P\to 0} \mathcal{N}\left(\boldsymbol{x}_k; \boldsymbol{\mathcal{X}}_{k|k}^{(i)}, P\right), \tag{6}$$

and assuming zero-mean additive process noise with covariance Q_k, the GSF algorithm[15,16] can be used to propagate this pdf over time, resulting in a predictive distribution given by

$$p\left(\boldsymbol{x}_{k+1}|\boldsymbol{y}_k\right) \approx \sum_{i=1}^{N} w_{k+1|k}^{(i)} \mathcal{N}\left(\boldsymbol{x}_{k+1}; \boldsymbol{f}_k\left(\boldsymbol{\mathcal{X}}_{k|k}^{(i)}\right), B_k\right), \tag{7}$$

with $w_{k+1|k}^{(i)} = w_{k|k}^{(i)}$, and,

$$B_k = \beta^2 \hat{P}_{k+1|k} + Q_k, \tag{8}$$

$$\hat{P}_{k+1|k} = \sum_{i=1}^{N} w_{k|k}^{(i)} \left[\boldsymbol{f}_k\left(\boldsymbol{\mathcal{X}}_{k|k}^{(i)}\right) - \sum_{i=1}^{N} w_{k|k}^{(i)} \boldsymbol{f}_k\left(\boldsymbol{\mathcal{X}}_{k|k}^{(i)}\right)\right] [\cdots]^{\mathrm{T}}, \tag{9}$$

where $[\cdots]$ signifies the same quantity as the preceding parentheses, and β^2 is the bandwidth parameter, calculated using Silverman's rule of thumb,[17]

$$\beta_{\text{sil}}^2 = \left(\frac{4}{N\left(N_s+2\right)}\right)^{\frac{2}{N_s+4}}, \tag{10}$$

where N_s is the dimension of the state space. Since the predictive distribution is typically non-Gaussian due to the use of nonlinear dynamics, Silverman's rule of thumb is known to be conservative.[18] Therefore, in the SMF, β_{sil}^2 can be scaled by a factor $\alpha > 0$, to achieve better results in terms of filter consistency,

$$\beta^2 = \alpha \beta_{\text{sil}}^2. \tag{11}$$

Once a measurement is obtained ($k \leftarrow k+1$), a posterior distribution is approximated from the prior Gaussian mixture,[15,16] such that

$$p\left(\boldsymbol{x}_k|\boldsymbol{y}_k\right) \approx \sum_{i=1}^{N} \widetilde{w}_{k|k}^{(i)} \mathcal{N}\left(\boldsymbol{x}_k; \widetilde{\boldsymbol{\mathcal{X}}}_{k|k}^{(i)}, \widetilde{\mathcal{P}}_{k|k}^{(i)}\right), \tag{12}$$

where the notation $\widetilde{\cdot}$ means that these are auxiliary variables obtained with a GSF update. Each mean and covariance of the Gaussian mixture are given by

$$\widetilde{\boldsymbol{\mathcal{X}}}_{k|k}^{(i)} = \boldsymbol{\mathcal{X}}_{k|k-1}^{(i)} + K_k^{(i)} \boldsymbol{\nu}_k^{(i)}, \tag{13}$$

$$\widetilde{\mathcal{P}}_{k|k}^{(i)} = B_{k-1} - K_k^{(i)} W_k^{(i)} K_k^{(i)\mathrm{T}}, \tag{14}$$

with the intermediate variables defined as

$$\boldsymbol{\mathcal{X}}_{k|k-1}^{(i)} = \boldsymbol{f}_{k-1}\left(\boldsymbol{\mathcal{X}}_{k-1|k-1}^{(i)}\right), \tag{15}$$

$$\boldsymbol{\nu}_k^{(i)} = \boldsymbol{y}_k - \boldsymbol{h}_k\left(\boldsymbol{\mathcal{X}}_{k|k-1}^{(i)}\right), \tag{16}$$

$$K_k^{(i)} = B_{k-1} H_k^{(i)\mathrm{T}} \left(W_k^{(i)}\right)^{-1}, \tag{17}$$

$$W_k^{(i)} = H_k^{(i)} B_{k-1} H_k^{(i)\mathrm{T}} + R_k, \tag{18}$$

$$H_k^{(i)} = \left.\frac{\partial \boldsymbol{h}_k\left(\boldsymbol{x}\right)}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{\mathcal{X}}_{k|k-1}^{(i)}}. \tag{19}$$

The weights of this new Gaussian mixture are computed as proportional to the probability of the obtained measurement,

$$\widetilde{w}_{k|k}^{(i)} \propto w_{k|k-1}^{(i)} \mathcal{N}\left(\boldsymbol{y}_k; \boldsymbol{h}_k\left(\boldsymbol{\mathcal{X}}_{k|k-1}^{(i)}\right), W_k^{(i)}\right). \tag{20}$$

With the approximated posterior distribution, a new grid is constructed in order to continue with the recursion. In the SMF, a new regular grid is assembled after the GSF update, using both the updated mean and covariance estimates,

$$\widetilde{\boldsymbol{x}}_{k|k} = \sum_{i=1}^{N} \widetilde{w}_{k|k}^{(i)} \widetilde{\boldsymbol{\mathcal{X}}}_{k|k}^{(i)}, \tag{21}$$

$$\widetilde{P}_{k|k} = \sum_{i=1}^{N} \widetilde{w}_{k|k}^{(i)} \left(\widetilde{\mathcal{P}}_{k|k}^{(i)} + \widetilde{\boldsymbol{\mathcal{X}}}_{k|k}^{(i)} \left(\widetilde{\boldsymbol{\mathcal{X}}}_{k|k}^{(i)}\right)^{\mathrm{T}} - \widetilde{\boldsymbol{x}}_{k|k} \widetilde{\boldsymbol{x}}_{k|k}^{\mathrm{T}}\right), \tag{22}$$

where the center of the grid is placed at the updated mean and the orientation and extent are set to align with the updated covariance. With this new grid, the posterior distribution is approximated as a Dirac mixture,

$$p\left(\boldsymbol{x}_k|\boldsymbol{y}_k\right) \approx \sum_{j=1}^{M} w_{k|k}^{(j)} \delta\left(\boldsymbol{x}_k - \boldsymbol{\mathcal{X}}_{k|k}^{(j)}\right), \tag{23}$$

where $\boldsymbol{\mathcal{X}}_{k|k}^{(j)}$ are the new grid points, and $w_{k|k}^{(j)}$ are the new weights for each point. It is important to note that M does not necessarily need to be equal to N.

To obtain the new weights for each point, the following process is used. First, $\boldsymbol{x}_k$ is marginalized to obtain an expression for the posterior distribution,

$$p\left(\boldsymbol{x}_k|\boldsymbol{y}_k\right) = \int_{\mathcal{S}(\boldsymbol{x}_{k-1})} p\left(\boldsymbol{x}_k, \boldsymbol{x}_{k-1}|\boldsymbol{y}_k\right)\, d\boldsymbol{x}_{k-1}, \tag{24}$$

$$= \int_{\mathcal{S}(\boldsymbol{x}_{k-1})} p\left(\boldsymbol{y}_k|\boldsymbol{x}_k\right) p\left(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}\right) p\left(\boldsymbol{y}_{k-1}|\boldsymbol{x}_{k-1}\right)\, d\boldsymbol{x}_{k-1}, \tag{25}$$

$$= p\left(\boldsymbol{y}_k|\boldsymbol{x}_k\right) \int_{\mathcal{S}(\boldsymbol{x}_{k-1})} p\left(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}\right) p\left(\boldsymbol{x}_{k-1}|\boldsymbol{y}_{k-1}\right)\, d\boldsymbol{x}_{k-1}. \tag{26}$$

At this point, note that the new grid ($\boldsymbol{\mathcal{X}}_{k|k}$) and the previous posterior grid ($\boldsymbol{\mathcal{X}}_{k-1|k-1}$) with its corresponding weights ($w_{k-1|k-1}$) are available. Since the previous posterior is approximated via a Dirac mixture (or a Gaussian mixture with infinitesimal covariance), the sifting property can be used to evaluate the obtained integral. Given

$$p\left(\boldsymbol{x}_{k-1}|\boldsymbol{y}_{k-1}\right) \approx \sum_{i=1}^{N} w_{k-1|k-1}^{(i)} \delta\left(\boldsymbol{x}_{k-1} - \boldsymbol{\mathcal{X}}_{k-1|k-1}^{(i)}\right), \tag{27}$$

then (26) can be rewritten as

$$p\left(\boldsymbol{x}_k|\boldsymbol{y}_k\right) \approx p\left(\boldsymbol{y}_k|\boldsymbol{x}_k\right) \int\limits_{\mathcal{S}(\boldsymbol{x}_{k-1})} p\left(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}\right) \sum_{i=1}^{N} w_{k-1|k-1}^{(i)} \delta\left(\boldsymbol{x}_{k-1} - \boldsymbol{\mathcal{X}}_{k-1|k-1}^{(i)}\right) d\boldsymbol{x}_{k-1}, \tag{28}$$

$$\approx p\left(\boldsymbol{y}_k|\boldsymbol{x}_k\right) \sum_{i=1}^{N} p\left(\boldsymbol{x}_k|\boldsymbol{\mathcal{X}}_{k-1|k-1}^{(i)}\right) w_{k-1|k-1}^{(i)}, \tag{29}$$

resulting in weights proportional to

$$w_{k|k}^{(j)} \propto p\left(\boldsymbol{y}_k|\boldsymbol{\mathcal{X}}_{k|k}^{(j)}\right) \sum_{i=1}^{N} p\left(\boldsymbol{\mathcal{X}}_{k|k}^{(j)}|\boldsymbol{\mathcal{X}}_{k-1|k-1}^{(i)}\right) w_{k-1|k-1}^{(i)}. \tag{30}$$

Note that the weighting of the new grid makes no assumption about the location or distribution of the points. The only assumption made throughout this derivation is the approximation of the pdfs as Dirac mixtures. In addition, from (30) it can be seen that this filtering algorithm has a linear time complexity in both N and M, $\mathcal{O}\left(N \cdot M\right)$. These equations summarize the standard SMF methodology proposed for constructing a more precise grid within the PMF framework. Figure 1 shows a visual summary of the key steps in this new technique.

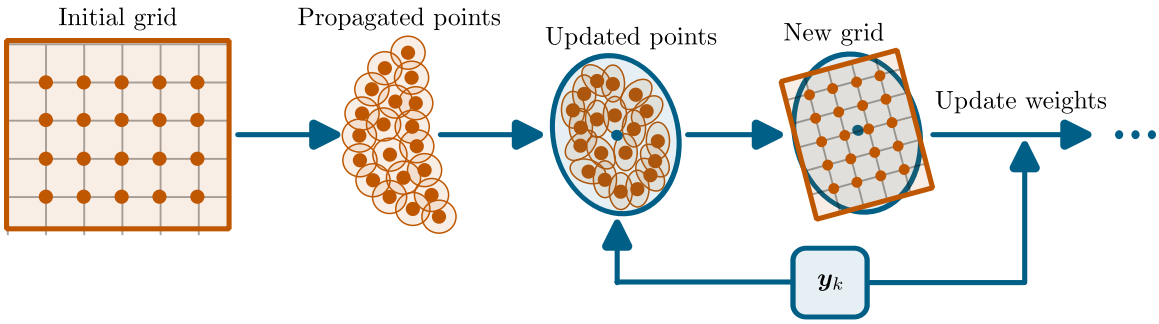

Figure 1. Flowchart summarizing the four key steps in the SMF: First, the initial grid representing the previous posterior is propagated. Then, the propagated points are updated using a GSF update. From the updated points, a new regular grid is placed at the updated mean and covariance. Finally, the grid is weighted accordingly.

Although the SMF has been shown to outperform the standard PMF,[9] this framework has clear limitations. The described procedure works well for approximately Gaussian pdfs, as only the mean and covariance from the GSF update are used to generate the new grid. However, if the posterior pdf is highly non-Gaussian, other techniques have to be used to obtain better grid point positions.

Limitations in Grid Construction

As described in the previous section, the SMF framework works well in the case of approximately Gaussian pdfs. This section provides an example where the SMF fails to obtain an accurate representation of the posterior distribution in the case of a bimodal pdf.[14] Consider the following system where the prior distribution is given by

$$p(\boldsymbol{x}_{k+1}|\boldsymbol{y}_k) \sim \mathcal{N}\left(\begin{bmatrix}0\\0\end{bmatrix}, \begin{bmatrix}1 & 0.5\\0.5 & 1\end{bmatrix}\right). \tag{31}$$

If a range measurement is assumed to be obtained from the origin according to

$$y = h(\boldsymbol{x}) = \sqrt{\left(x^{(1)}\right)^2 + \left(x^{(2)}\right)^2}, \tag{32}$$

with a value of $y = 3$ and a measurement error covariance of $R = 0.1^2$, the true posterior distribution will have two modes that resemble the shape of two bananas, as shown in Figure 2. If a GSF update is performed on an initial grid centered on the prior mean and expanding the prior covariance (simulating the propagated points from the previous posterior distribution), the GSF-updated points cluster around the two modes of the true posterior distribution, as illustrated in the left panel of Figure 2. If the new grid is created using only the first two moments of the GSF-updated points (as is done in the SMF), a significant amount of information is lost, as seen in the right panel of Figure 2. In this case, using the standard PMF approach, where the grid is centered at the prior mean and expands the prior covariance, could yield a better grid approximation.

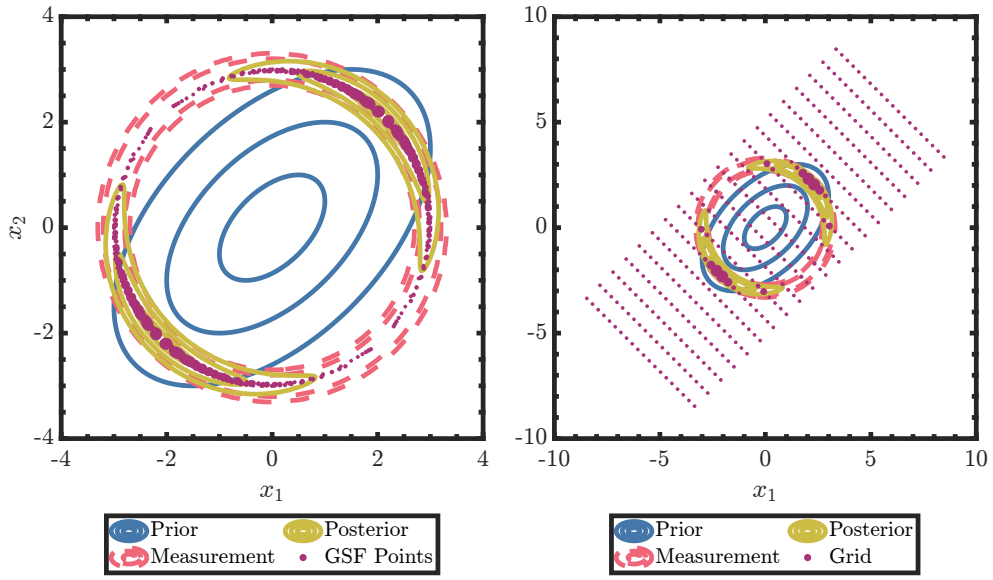

Figure 2. GSF-updated points in purple (left) and new grid in purple (right). The prior distribution is shown in blue, the measurement likelihood in red with dashed lines, and the posterior distribution in solid yellow lines.

Therefore, a more sophisticated method that considers all the information provided by the GSF update is required to create the new grid. Consequently, the problem to be solved in this work can be formulated as follows.

Given the information from the GSF-updated points, what is the optimal grid?

In other words, this work addresses the challenge of optimally representing the GSF-approximated posterior distribution,

$$p(\boldsymbol{x}_k|\boldsymbol{y}_k) \approx \sum_{i=1}^{N} \tilde{w}_{k|k}^{(i)} \mathcal{N}\left(\boldsymbol{x}_k; \tilde{\boldsymbol{\mathcal{X}}}_{k|k}^{(i)}, \tilde{\mathcal{P}}_{k|k}^{(i)}\right) \tag{33}$$

as a Dirac mixture approximation,

$$p(\boldsymbol{x}_k|\boldsymbol{y}_k) \approx \sum_{j=1}^{M} w_{k|k}^{(j)} \delta\left(\boldsymbol{x}_k - \boldsymbol{\mathcal{X}}_{k|k}^{(j)}\right). \tag{34}$$

Note that one possible solution would be to simply use the GSF-updated points for the next iteration. However, using these points to represent the Gaussian mixture in (33) as a Dirac mixture in (34) can result in a poor approximation. As mentioned earlier, each GSF-updated point represents a Gaussian distribution rather than a single point, meaning that their locations are not optimized to be treated as such. Furthermore, points with low probability would be fully considered, which could lead to filter degeneracy or divergence, particularly in cases of poor propagation or low-quality measurements.

A FIRST APPROACH USING CLUSTERING ALGORITHMS

By examining Figure 2, it is clear that the GSF-updated points are clustered around the two different modes of the true posterior distribution. The grouping of these points suggests the potential use of cluster-based algorithms to autonomously determine the number of statistically significant groups and place independent grids on each group. This section explores the possibility of using clustering algorithms to obtain higher quality grids.

Density-based spatial clustering of applications with noise (DBSCAN)[19] is a widely used clustering algorithm. As its name suggests, this is a density-based clustering algorithm that is used to find clusters and noise in a set of points. The advantage of this algorithm is that, since it is purely density-based, there is no need to pre-define the number of clusters desired, unlike other clustering methods such as k-means. In this algorithm, each point is classified as core, border, or noise. Given the parameters `epsilon` and `minPts`, the algorithm works by following the next steps:[19]

1. An unlabeled point is selected as the current point and the first cluster is initialized.
2. The neighboring points within an `epsilon` distance from the current point are found.
3. If the number of neighbors is less than `minPts`, the current point is labeled as noise. Otherwise, the point is labeled as a core point that belongs to the current cluster.
4. The same process is repeated over each neighbor until no new neighbors can be added to the current cluster.
5. Steps 1 to 4 are repeated until all points are labeled.

Using DBSCAN, this approach proposes first to classify the GSF-updated points into a set of clusters, such that

$$\widetilde{\boldsymbol{\mathcal{X}}}_{k|k} = \left\{ {}^{(1)}\widetilde{\boldsymbol{\mathcal{X}}}_{k|k}, {}^{(2)}\widetilde{\boldsymbol{\mathcal{X}}}_{k|k}, \ldots, {}^{(L)}\widetilde{\boldsymbol{\mathcal{X}}}_{k|k} \right\}, \tag{35}$$

where ${}^{(i)}\widetilde{\boldsymbol{\mathcal{X}}}_{k|k}$ represents each cluster and L is the total number of clusters found. With the clusters identified, the mean and covariance of each one can be calculated as

$${}^{(i)}\widetilde{\boldsymbol{x}}_{k|k} = \sum_{j=1}^{{}^{(i)}N} {}^{(i)}\widetilde{w}_{k|k}^{(j)} \, {}^{(i)}\widetilde{\boldsymbol{\mathcal{X}}}_{k|k}^{(j)}, \tag{36}$$

$${}^{(i)}\widetilde{P}_{k|k} = \sum_{j=1}^{{}^{(i)}N} {}^{(i)}\widetilde{w}_{k|k}^{(j)} \left({}^{(i)}\widetilde{\mathcal{P}}_{k|k}^{(j)} + {}^{(i)}\widetilde{\boldsymbol{\mathcal{X}}}_{k|k}^{(j)} \left({}^{(i)}\widetilde{\boldsymbol{\mathcal{X}}}_{k|k}^{(j)} \right)^{\mathrm{T}} - {}^{(i)}\widetilde{\boldsymbol{x}}_{k|k} \, {}^{(i)}\widetilde{\boldsymbol{x}}_{k|k}^{\mathrm{T}} \right), \tag{37}$$

where ${}^{(i)}N$ are the number of points in the i-th cluster. Note that the weights of the clusters need to be re-normalized, such that

$$\sum_{j=1}^{{}^{(i)}N} {}^{(i)}\widetilde{w}_{k|k}^{(j)} = 1. \tag{38}$$

With the mean and covariance of each cluster, new grids are placed at each cluster center, aligned with its mean and covariance. The number of grid points for every new grid is set to be equal to the number of points in its corresponding cluster,

$$\sum_{i=1}^{L} {}^{(i)}N = N, \tag{39}$$

making $M = N$. The union of all grids is then used as the new grid, which can be weighted as in (30). It is important to mention that, since DBSCAN can classify points as noise, the GSF-updated points classified as noise are not used to place the grids. Instead, the total number of noise points is distributed among the new grids to satisfy the constraint $M = N$.

As mentioned above, the number of grid points for every new grid is set to be equal to the number of points in its corresponding cluster. Satisfying this constraint can be difficult using regular grids, as not every number can be expressed as a sum of squares (in the case of 2D grids). For this work, instead of using regular grids, generalized Fibonacci grids[20] are used to place a new grid on top of each cluster. Generalized Fibonacci grids are a simple method to obtain equally weighted deterministic samples of a multivariate Gaussian pdf, allowing the placement of an arbitrary number of homogeneously distributed samples, even in high-dimensional spaces. Figure 3 shows three Fibonacci grids, each obtained for a different number of grid points in the case of a standard two-dimensional Gaussian. Note that as fewer points are used, fewer points lie outside the 2σ bounds. Therefore, if a smaller number of points is used, the grid points can be stretched without losing homogeneity, to cover up to an arbitrary number of σ, resulting in a more conservative approach.

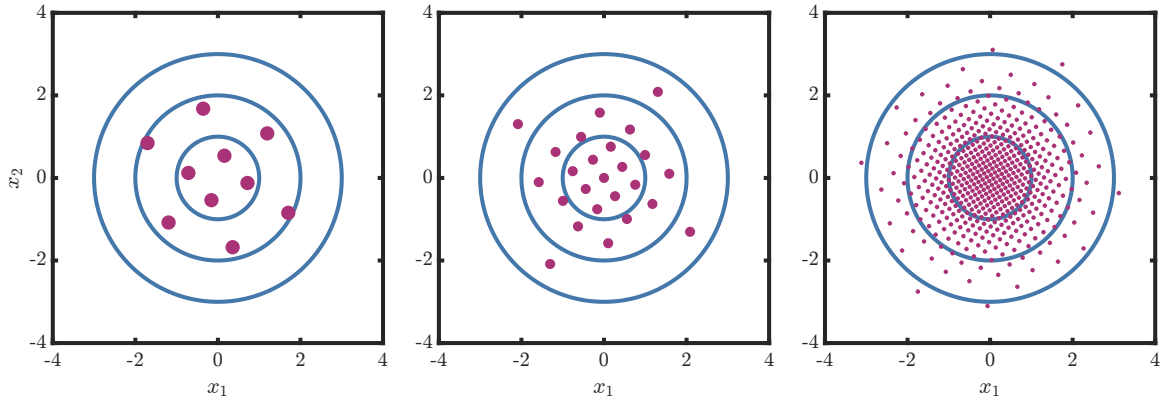

Figure 3. Fibonacci grids for a standard two-dimensional Gaussian with 10 grid points (left), 25 grid points (center), 500 grid points (right). The circles represent the 1σ, 2σ, and 3σ bounds.

It is important to note that this strategy can result in suboptimal grids for two reasons. First, the results obtained with DBSCAN depend highly on the choice of the parameters `epsilon` and `minPts`. Second, since DBSCAN is a density-based algorithm, it does not consider the probability mass of each GSF-updated point, which represents a Gaussian distribution rather than a single point. However, this clustering algorithm has the potential to produce higher-quality grids compared to the original strategy in the SMF. In addition, since DBSCAN has a worst-case time complexity of $\mathcal{O}(N^2)$, using this framework will not affect the overall complexity of the filtering algorithm.

To study the performance of this new grid construction scheme, consider the example presented in the previous section. In this case, to evaluate the quality of the new grid, the effective grid size is used. This metric is defined as

$$M_{\text{eff}} = \frac{1}{M \sum_{i=1}^{M} w_{k|k}^{(i)}}. \tag{40}$$

If M_{eff} is close to 1, then the new grid accurately represents the approximate posterior distribution. In the case where $M_{\text{eff}} \ll 1$, the new grid is not an appropriate representation of the underlying pdf. Figure 4 shows the different grids obtained for the dual banana problem using the standard SMF approach and the clustering strategy with different values for `epsilon` and `minPts`. More specifically, the top-left figure shows the GSF-updated points with the traditional grid at the bottom-left. The top-center figure shows the clusters obtained (in different colors) for the GSF-updated points using `epsilon` $= 0.50$ and `minPts` $= 5$, with the corresponding new grid at the bottom-center. The top-right figure shows the clusters obtained for the GSF-updated points using `epsilon` $= 0.17$ and `minPts` $= 3$, with the corresponding new grid at the bottom-right.

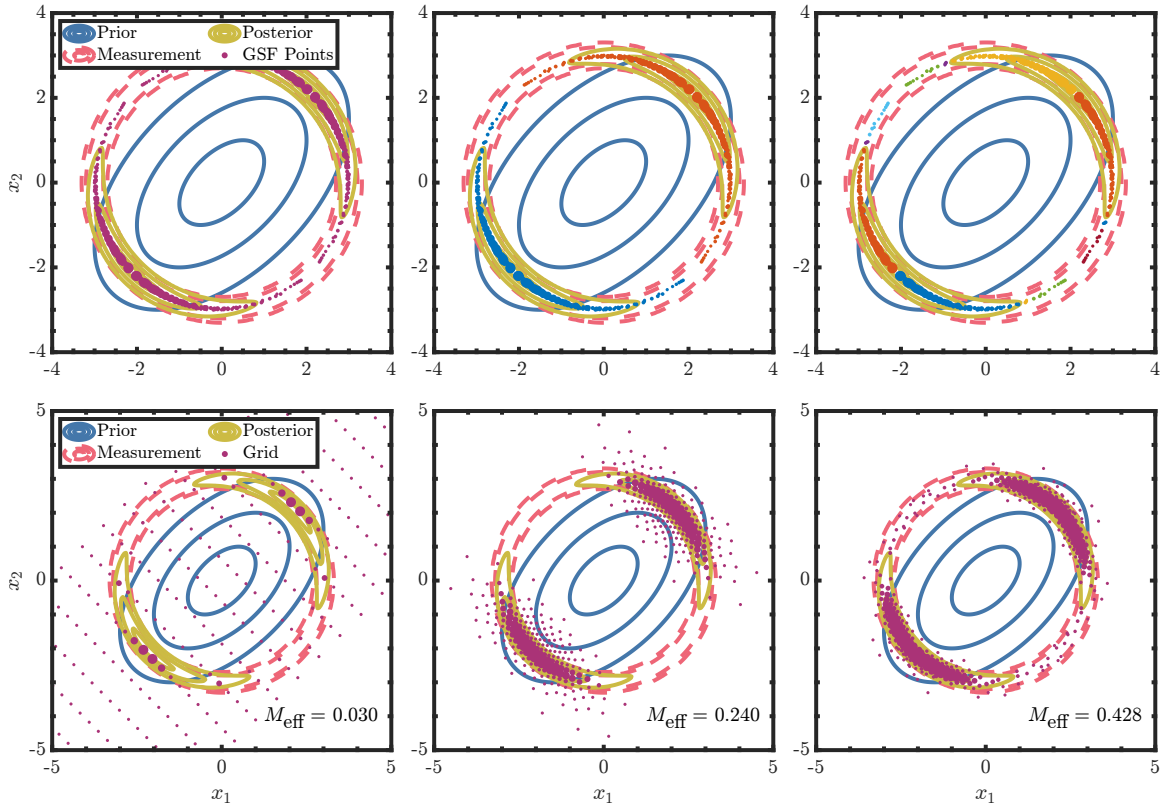

Figure 4. Standard grid placement (left), grid placement using DBSCAN with `epsilon` $= 0.50$ and `minPts` $= 5$ (center), and grid placement using DBSCAN with `epsilon` $= 0.17$ and `minPts` $= 3$ (right). The top panels show the GSF-updated points and clusters, while the bottom panels show the grid placement.

This figure shows the advantages of using the clustering algorithm over the standard SMF method. By using the clustering algorithm with `epsilon` $= 0.50$ and `minPts` $= 5$, the GSF-updated points are divided into two clusters that accurately represent the two modes of the posterior distribution.

Placing the new grid on top of the two identified clusters significantly increases the effective grid size compared to the standard grid. If `epsilon` and `minPts` are decreased, the GSF-updated points are divided into more clusters. By having more clusters, additional grids are superimposed, better capturing the underlying shape of the posterior pdf, and thus further increasing the effective grid size. Although the use of more clusters may increase the effective grid size, it is important to mention that this can affect the consistency of the overall filter, as the new grid can become too narrow, potentially missing the tails of the true posterior pdf.

A SOLUTION BASED ON OPTIMAL POINT SAMPLING

As mentioned in the previous section, although using clustering-based algorithms might yield higher-quality grids when compared to the standard SMF grid, relying solely on density-based methods can lead to overall suboptimal grids. Since the GSF-updated points carry an individual mean, an individual covariance, and an associated probability mass, density-based methods will not use the full information provided by the Gaussian mixture. This section presents an alternative solution that uses all available information to find an optimal grid using a reduction of Dirac mixture densities by minimizing a distance measure.

Recall that the goal of this work is to represent the GSF-approximated posterior distribution,

$$p\left(\boldsymbol{x}_k|\boldsymbol{y}_k\right) \approx \sum_{i=1}^{N} \tilde{w}_{k|k}^{(i)} \mathcal{N}\left(\boldsymbol{x}_k; \tilde{\boldsymbol{\mathcal{X}}}_{k|k}^{(i)}, \tilde{\mathcal{P}}_{k|k}^{(i)}\right) \tag{41}$$

as a Dirac mixture approximation,

$$p\left(\boldsymbol{x}_k|\boldsymbol{y}_k\right) \approx \sum_{j=1}^{M} w_{k|k}^{(j)} \delta\left(\boldsymbol{x}_k - \boldsymbol{\mathcal{X}}_{k|k}^{(j)}\right). \tag{42}$$

Therefore, previously developed tools can be combined to find optimal grids by maximizing the similarity between the two distributions. Specifically, the strategy used in this work consists of two distinct steps. First, deterministic samples are obtained from each of the Gaussian components that represent the GSF-approximated posterior distribution. Second, the union of these deterministic samples is optimally reduced to obtain a high-quality Dirac mixture approximation.

To explain the process of obtaining the high-quality Dirac mixture approximation, two important concepts have to be introduced first, the localized cumulative distribution (LCD) and the Modified Cramér-von Mises Distance (MCVMD).[21–23]

Localized Cumulative Distribution

Let $p\left(\boldsymbol{x}\right) : \mathbb{R}^{N_s} \rightarrow \mathbb{R}_+$ be the pdf of a random vector $\boldsymbol{x} \in \mathbb{R}^{N_s}$. The corresponding LCD is defined as

$$P\left(\boldsymbol{m}, b\right) = \int_{\mathbb{R}^{N_s}} p\left(\boldsymbol{x}\right) \mathcal{K}\left(\boldsymbol{x} - \boldsymbol{m}, \boldsymbol{b}\right)\, d\boldsymbol{x}, \tag{43}$$

where $\mathcal{K}\left(\boldsymbol{x} - \boldsymbol{m}, \boldsymbol{b}\right)$ is a kernel centered on $\boldsymbol{m}$ with bandwidth $\boldsymbol{b}$.[23] This work considers Gaussian kernels of the form

$$\mathcal{K}\left(\boldsymbol{x} - \boldsymbol{m}, b\right) = \prod_{i=1}^{N_s} \exp\left(\frac{-\left(x^{(i)} - m^{(i)}\right)^2}{2b^2}\right). \tag{44}$$

With this definition and kernel, the LCD for a Dirac mixture is given by

$$P\left(\boldsymbol{m}, b\right) = \sum_{i=1}^{N} w_x^{(i)} \prod_{i=1}^{N_s} \exp\left(-\frac{1}{2}\frac{\left(x^{(i)} - m^{(i)}\right)^2}{b^2}\right). \tag{45}$$

Modified Cramér-von Mises Distance

The MCVMD allows for the comparison of similarity between two different distributions without restrictions on whether they are continuous or discrete. This distance is defined as the integral of the square of the difference between the LCDs of two pdfs,[23]

$$D_{\text{MCVM}} = \int_{\mathbb{R}_+} w(b) \int_{\mathbb{R}^n} \left(P_1\left(\boldsymbol{m}, b\right) - P_2\left(\boldsymbol{m}, b\right)\right)^2 \, d\boldsymbol{m}\, db, \tag{46}$$

where $w(b)\colon \mathbb{R}_+ \mapsto [0,1]$ is an appropriate weighting function. If the weighting function for the MCVMD is defined as

$$w(b) = \begin{cases} b^{1-n}, & b \in [0, b_{\max}] \\ 0, & \text{otherwise,} \end{cases} \tag{47}$$

then the distance between the two distributions and its corresponding gradient can be easily approximated for large values of $b_{\max}$. In the case of two Dirac mixtures and a large value of $b_{\max}$, the MCVMD between these two distributions has a closed-form solution and a closed-form solution gradient as well.[23]

Let the two Dirac mixtures be expressed as

$$p_1\left(\boldsymbol{x}\right) = \sum_{i=1}^{N} w_{\boldsymbol{\mathcal{X}}_1}^{(i)} \delta\left(\boldsymbol{x} - \boldsymbol{\mathcal{X}}_1^{(i)}\right), \tag{48}$$

$$p_2\left(\boldsymbol{x}\right) = \sum_{i=1}^{M} w_{\boldsymbol{\mathcal{X}}_2}^{(i)} \delta\left(\boldsymbol{x} - \boldsymbol{\mathcal{X}}_2^{(i)}\right), \tag{49}$$

where $\boldsymbol{w}_{\boldsymbol{\mathcal{X}}_1}$ and $\boldsymbol{w}_{\boldsymbol{\mathcal{X}}_2}$ are column vectors that represent the weights of each distribution, such that

$$\boldsymbol{w}_{\boldsymbol{\mathcal{X}}_1} = \begin{bmatrix} w_{\boldsymbol{\mathcal{X}}_1}^{(1)} & w_{\boldsymbol{\mathcal{X}}_1}^{(2)} & \cdots & w_{\boldsymbol{\mathcal{X}}_1}^{(N)} \end{bmatrix}^{\mathrm{T}}, \tag{50}$$

$$\boldsymbol{w}_{\boldsymbol{\mathcal{X}}_2} = \begin{bmatrix} w_{\boldsymbol{\mathcal{X}}_2}^{(1)} & w_{\boldsymbol{\mathcal{X}}_2}^{(2)} & \cdots & w_{\boldsymbol{\mathcal{X}}_2}^{(M)} \end{bmatrix}^{\mathrm{T}}, \tag{51}$$

and $\boldsymbol{\mathcal{X}}_1$, $\boldsymbol{\mathcal{X}}_2$, are the matrices representing the set of points in the Dirac mixture,

$$\boldsymbol{\mathcal{X}}_1 = \begin{bmatrix} \boldsymbol{\mathcal{X}}_1^{(1)} & \boldsymbol{\mathcal{X}}_1^{(2)} & \cdots & \boldsymbol{\mathcal{X}}_1^{(N)} \end{bmatrix}, \tag{52}$$

$$\boldsymbol{\mathcal{X}}_2 = \begin{bmatrix} \boldsymbol{\mathcal{X}}_2^{(1)} & \boldsymbol{\mathcal{X}}_2^{(2)} & \cdots & \boldsymbol{\mathcal{X}}_2^{(M)} \end{bmatrix}. \tag{53}$$

Note that $\boldsymbol{\mathcal{X}}_1$ is a $N_s \times N$ matrix and $\boldsymbol{\mathcal{X}}_2$ is a $N_s \times M$ matrix. With this notation, the MCVMD for these two Dirac Mixtures can be written as

$$\begin{aligned} D_{\text{MCVM}} = {} & \boldsymbol{w}_{\boldsymbol{\mathcal{X}}_1}^{\mathrm{T}} \boldsymbol{M}_{\boldsymbol{\mathcal{X}}_1\boldsymbol{\mathcal{X}}_1} \boldsymbol{w}_{\boldsymbol{\mathcal{X}}_1} - 2\boldsymbol{w}_{\boldsymbol{\mathcal{X}}_1}^{\mathrm{T}} \boldsymbol{M}_{\boldsymbol{\mathcal{X}}_1\boldsymbol{\mathcal{X}}_2} \boldsymbol{w}_{\boldsymbol{\mathcal{X}}_2} \\ & + \boldsymbol{w}_{\boldsymbol{\mathcal{X}}_2}^{\mathrm{T}} \boldsymbol{M}_{\boldsymbol{\mathcal{X}}_2\boldsymbol{\mathcal{X}}_2} \boldsymbol{w}_{\boldsymbol{\mathcal{X}}_2} + K \cdot ||\boldsymbol{\mathcal{X}}_1 \boldsymbol{w}_{\boldsymbol{\mathcal{X}}_1} - \boldsymbol{\mathcal{X}}_2 \boldsymbol{w}_{\boldsymbol{\mathcal{X}}_2}||_2^2, \end{aligned} \tag{54}$$

where $K > 0$ is a constant and,

$$\boldsymbol{M}_{\boldsymbol{\mathcal{X}}_1\boldsymbol{\mathcal{X}}_1} = \operatorname{xlog}\left(\boldsymbol{D}_{\boldsymbol{\mathcal{X}}_1\boldsymbol{\mathcal{X}}_1}\right), \tag{55}$$

$$\boldsymbol{M}_{\boldsymbol{\mathcal{X}}_1\boldsymbol{\mathcal{X}}_2} = \operatorname{xlog}\left(\boldsymbol{D}_{\boldsymbol{\mathcal{X}}_1\boldsymbol{\mathcal{X}}_2}\right), \tag{56}$$

with $\operatorname{xlog}(z) = z \cdot \log(z)$ and $\boldsymbol{D}_{\boldsymbol{\mathcal{X}}_i\boldsymbol{\mathcal{X}}_j}$ representing the Euclidean distance matrix between $\boldsymbol{\mathcal{X}}_i$ and $\boldsymbol{\mathcal{X}}_j$. The gradient with respect to $\boldsymbol{\mathcal{X}}_1$ or $\boldsymbol{\mathcal{X}}_2$ is not presented for brevity, although it also has a closed-form solution that can be obtained by differentiating (54).

Optimal Sampling of Gaussian Mixtures

With the LCD and MCVMD defined, the process of obtaining high-quality grids can be studied. As mentioned before, the first step is to obtain equally weighted deterministic samples from each Gaussian component, such that

$$\mathcal{N}\left(\boldsymbol{x}_k; \tilde{\boldsymbol{\mathcal{X}}}^{(i)}_{k|k}, \tilde{\mathcal{P}}^{(i)}_{k|k}\right) \approx \frac{1}{D}\sum_{j=1}^{D} \delta\left(\boldsymbol{x}_k - \boldsymbol{\mathcal{D}}^{(j)}_{k|k}\right), \tag{57}$$

where D are the total number of points to be sampled from each Gaussian component and $\boldsymbol{\mathcal{D}}^{(j)}_{k|k}$ are the deterministic samples. With this approximation, the Gaussian mixture obtained with the GSF update can be expressed as

$$p\left(\boldsymbol{x}_k|\boldsymbol{y}_k\right) \approx \sum_{i=1}^{N} \frac{\tilde{w}^{(i)}_{k|k}}{D} \sum_{j=1}^{D} \delta\left(\boldsymbol{x}_k - \boldsymbol{\mathcal{D}}^{(j)}_{k|k}\right). \tag{58}$$

To find the discretization points, the previously mentioned Fibonacci grids are used. A set of deterministic samples $\boldsymbol{\mathcal{D}}$, representing the Fibonacci grid of a standard multivariate Gaussian (i.e., zero mean and identity covariance), is first obtained. Once the deterministic samples have been obtained, they can be scaled and shifted by the corresponding mean and covariance,

$$\boldsymbol{\mathcal{D}}_{k|k} = \left[\tilde{\mathcal{P}}^{(i)}_{k|k}\right]^{1/2} \boldsymbol{\mathcal{D}} + \tilde{\boldsymbol{\mathcal{X}}}^{(i)}_{k|k}. \tag{59}$$

For this work, the block Schur algorithm is used to compute the square root of the matrix.[24] Note that the whole process of obtaining the unscaled deterministic samples $\boldsymbol{\mathcal{D}}$ can be performed offline, thus not increasing the overall complexity of the filtering algorithm.

Once the Gaussian mixture has been represented as a union of Dirac mixtures, the final step is to optimally reduce the approximation to obtain the desired high-quality grid. This final step can be expressed as the approximation of

$$\sum_{i=1}^{N} \frac{\tilde{w}^{(i)}_{k|k}}{D} \sum_{j=1}^{D} \delta\left(\boldsymbol{x}_k - \boldsymbol{\mathcal{D}}^{(j)}_{k|k}\right) \approx \frac{1}{M}\sum_{j=1}^{M} \delta\left(\boldsymbol{x}_k - \boldsymbol{\mathcal{X}}^{(j)}_{k|k}\right). \tag{60}$$

In this representation, the discrete density obtained in the previous step is reduced to M equally weighted discretization points, representing the optimal grid. For simplicity, this sampling solution uses $M = N$. In this work, the new discretization points are determined by minimizing the

MCVMD between the two Dirac mixtures using Matlab's `fminunc` function, which uses the BFGS quasi-Newton method with a cubic line search procedure.

Note that this step has to be performed online as it requires the current GSF-approximated posterior. However, if $D \cdot N$ is the total number of deterministic samples obtained from all the Gaussian components in the Gaussian mixture and $D \cdot N \gg M$, then the complexity of evaluating the MCVMD is linear in both $D \cdot N$ and M, i.e., $\mathcal{O}(D \cdot N \cdot M \cdot N_s)$,[23] where N_s is the state dimension, thus not overly increasing the complexity of the filtering algorithm.

Figure 5 shows a two-dimensional example of the optimal point sampling procedure discussed. The underlying distribution is a Gaussian mixture with four equally weighted components, each representing a petal of the distribution in the form of a flower. For each Gaussian component, 144 deterministic samples are obtained using scaled Fibonacci grids. These points, shown in gold in the figure, are the same standard points used for each component, but each set is shifted by its corresponding mean and covariance. The samples of each Gaussian component are joined to form the first Dirac mixture approximation to the Gaussian mixture. Finally, the Dirac mixture is optimally reduced to 50 equally weighted points, shown in purple.

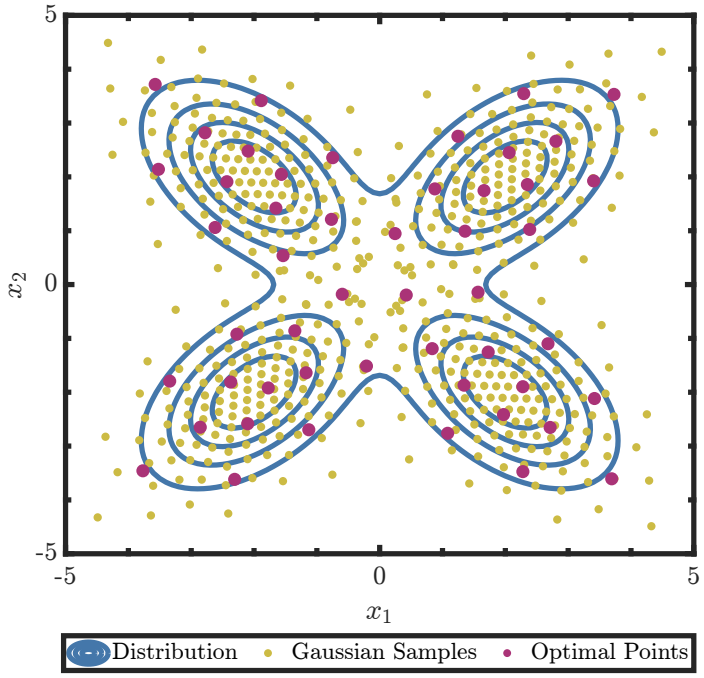

Figure 5. Optimal point sampling of a Gaussian mixture. The distribution is shown with solid blue lines, the first Dirac approximation is shown with yellow points, and the optimally reduced approximation is shown with purple points.

Returning to the dual banana problem, Figure 6 shows the grid obtained by the optimal point sampling solution. As each of these new grid points is equally weighted, $M_{\text{eff}} = 1$ for this case. From this figure, it can be seen that the optimal grid is able to capture the entirety of the posterior distribution, showing a clear improvement over both the standard solution and the clustering approaches. It is important to note that these points are completely deterministic, meaning that no random number generator is used, unlike the various resampling strategies commonly used in particle filters.

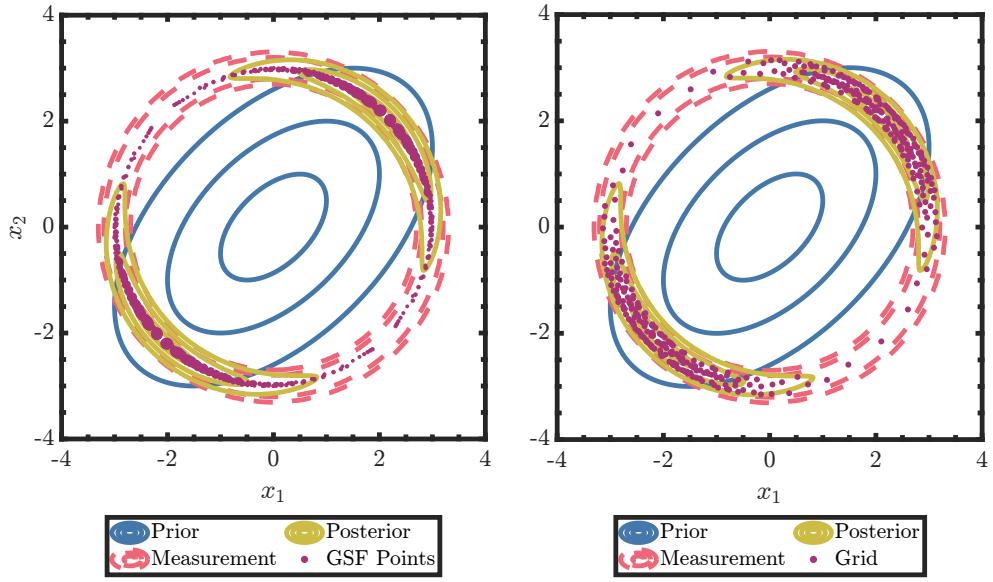

Figure 6. GSF-updated points in purple (left) and new optimal grid in purple (right). The prior distribution is shown in blue, the measurement likelihood in red with dashed lines, and the posterior distribution in solid yellow lines.

SEQUENTIAL FILTERING WITH THE IKEDA MAP

To evaluate the performance of the proposed approaches in a sequential filtering problem, the Ikeda map is used. This function represents a discrete-time dynamical system, where

$$x_{k+1}^{(1)} = 1 + u\left(x_k^{(1)}\cos t_k - x_k^{(2)}\sin t_k\right), \tag{61}$$

$$x_{k+1}^{(2)} = u\left(x_k^{(1)}\sin t_k + x_k^{(2)}\cos t_k\right), \tag{62}$$

$$t_k = 0.4 - \frac{6}{1+\left(x_k^{(1)}\right)^2 + \left(x_k^{(2)}\right)^2}. \tag{63}$$

This highly nonlinear function is designed to model light circulating within a nonlinear optical resonator.[25,26] To induce a chaotic behavior and have a more difficult filtering problem, the parameter u can be set to $u \geq 0.6$. Figure 7 shows 1,000 trajectories for different initial states propagated using $u = 0.9$.

For this sequential filtering example, the parameter u is set to $u = 0.9$ and the dynamics are propagated with additive white Gaussian process noise with covariance matrix $Q = 1 \times 10^{-2} I_{2\times 2}$. In addition, a range measurement is assumed,

$$y_k = \sqrt{\left(x_k^{(1)}\right)^2 + \left(x_k^{(2)}\right)^2} + \eta_k, \tag{64}$$

where η_k represents white Gaussian measurement noise with scalar covariance matrix $R = 1$. For this example, the three strategies presented in this work are used and evaluated. The three strategies are summarized below.

1. **SMF:** This represents the standard SMF grid design, where the new grid is placed at the mean of the GSF-updated points and aligned with the updated covariance.

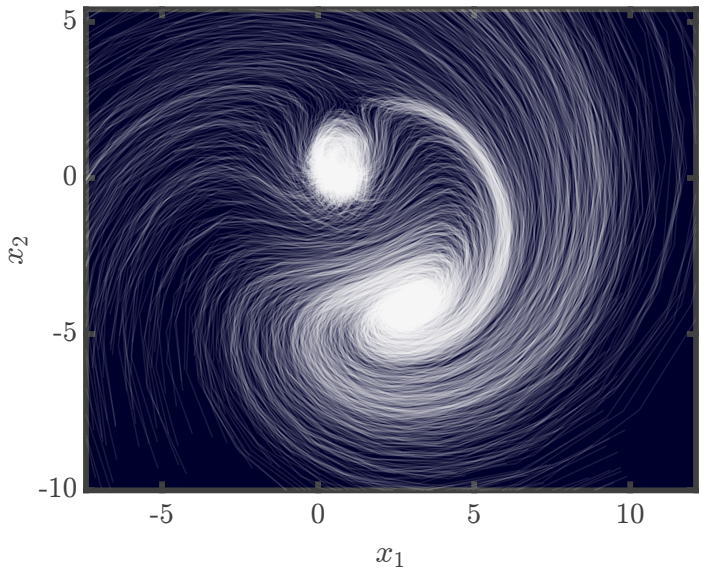

Figure 7. Multiple trajectories for different initial states of the Ikeda map with $u = 0.9$.

2. **SMF-DBS:** This strategy involves the clustering methodology, where the GSF-updated points are clustered using DBSCAN, and a Fibonacci grid is placed on top of each cluster.
3. **SMF-OPS:** This strategy uses the optimal point sampling approach, where deterministic points are first sampled from each Gaussian component represented by the GSF-updated points, and then optimally reduced by minimizing the MCVMD.

Each strategy is used to estimate the state in 1,000 distinct trajectories simulated for 50 time units, with an initial true state defined as $\boldsymbol{x}_0 \sim \mathcal{N}(0_{2\times1}, I_{2\times2})$. All filters use grids with 25 grid points ($N = 25$), and start with a regular grid centered on the initial state estimate. The regular grids in the standard SMF are set to expand up to 3σ in each direction. The Fibonacci grids used in the SMF-DBS for each cluster are stretched to expand up to 6σ in each direction, with `epsilon` $= 0.275$ and `minPts` $= 10$. The Fibonacci grids used in the SMF-OPS to sample each Gaussian component are composed of 5 points, with no stretching applied. The scaling factor used to scale Silverman's rule of thumb for the approximation of the predictive distribution is set to 0.3 for all strategies ($\alpha = 0.3$).

To compare each strategy, two metrics are used. The root mean squared error (RMSE) is used to assess the accuracy of the filter. For this work, the RMSE is defined as follows

$$\text{RMSE}(k) = \sum_{j=1}^{N_m} \frac{1}{N_m} \sqrt{\frac{1}{N_s} \sum_{i=1}^{N_s} \left(x_{k,j}^{(i)} - \hat{x}_{k|k,j}^{(i)} \right)^2}, \tag{65}$$

where N_m is the number of independent trajectories, N_s is the state dimension, $x_{k,j}^{(i)}$ is the true state, and $\hat{x}_{k|k,j}^{(i)}$ is the estimated state. To evaluate the consistency of the filter, the scaled normalized estimation error squared (SNEES) is used,

$$\text{SNEES}(k) = \frac{1}{N_s N_m} \sum_{j=1}^{N_m} \left(\boldsymbol{x}_{k,j} - \hat{\boldsymbol{x}}_{k|k,j} \right)^{\mathrm{T}} \left(P_{k|k,j} \right)^{-1} \left(\boldsymbol{x}_{k,j} - \hat{\boldsymbol{x}}_{k|k,j} \right). \tag{66}$$

A consistent filter will obtain a SNEES close to one. If the SNEES value exceeds one, it suggests overconfidence in the estimator. Conversely, if the value is considerably lower than one, the estimator is overly conservative.[1]

Figure 8 shows the RMSE as a function of time for the three strategies presented. The green dashed line represents the results obtained with a regularized particle filter with 10,000 particles, providing a sense of the best achievable performance in this problem setting. As seen in the figure, the standard SMF yields the highest RMSE, indicating the lowest accuracy in the estimates. The SMF-DBS increases accuracy compared to the standard SMF, approaching the SIR $\rightarrow \infty$ line. Regardless, by not considering the probability mass of the GSF-updated points, accuracy is lost as the new grids are not directly sampling the Gaussian mixture represented by the updated points. In this sense, since the SMF-OPS is optimally sampling the Gaussian mixture, it achieves the best accuracy among the three strategies, closely following the best achievable performance line, converging to Bayesian inference.

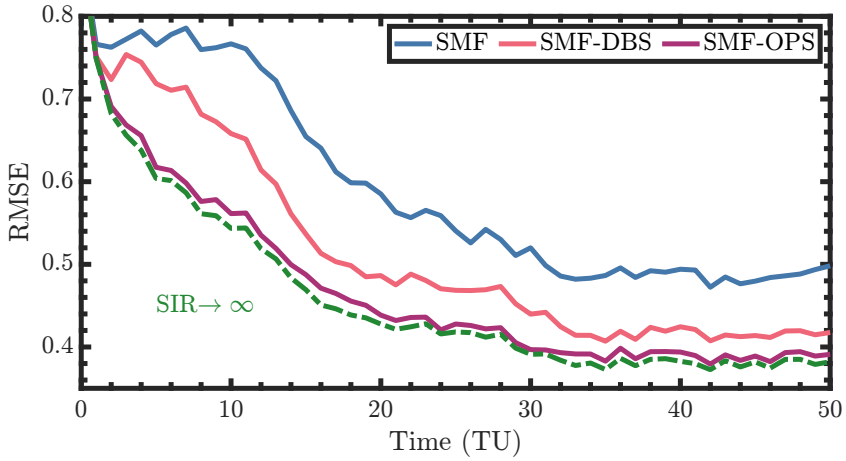

Figure 8. RMSE as a function of time for the three strategies presented. The green dashed line shows the results obtained with a regularized particle filter with 10,000 particles.

In figure 9, the SNEES is shown as a function of time for the three strategies. In this case, it can be observed that the standard SMF has high SNEES values, suggesting overconfidence in its estimates. This is mainly due to the low number of grid points used in this problem setting. With an increasing number of grid points, the standard SMF has been shown to produce an accurate and consistent estimate.[9] However, both the SMF-DBS and SMF-OPS outperform the standard SMF in low grid point settings. The SMF-DBS achieves a SNEES value closer to one compared to the standard SMF, though not exactly one. The SMF-OPS obtains a SNEES very close to one at all time steps, indicating both an accurate and consistent filter. It is important to note that the factor α can be increased to obtain better consistency with both the SMF and the SMF-DBS, but this would not increase the accuracy of the filter.[9]

From these results, it is important to note that each strategy has a different level of implementation complexity. The grid design in the standard SMF is by far the simplest, as its implementation is straightforward and only requires the tuning of two parameters, specifically the extent of the grid and the bandwidth parameter α. The SMF-DBS is more complex to tune, as the clustering algorithm is highly sensitive to the two tuning parameters, `epsilon` and `minPts`. The SMF-OPS

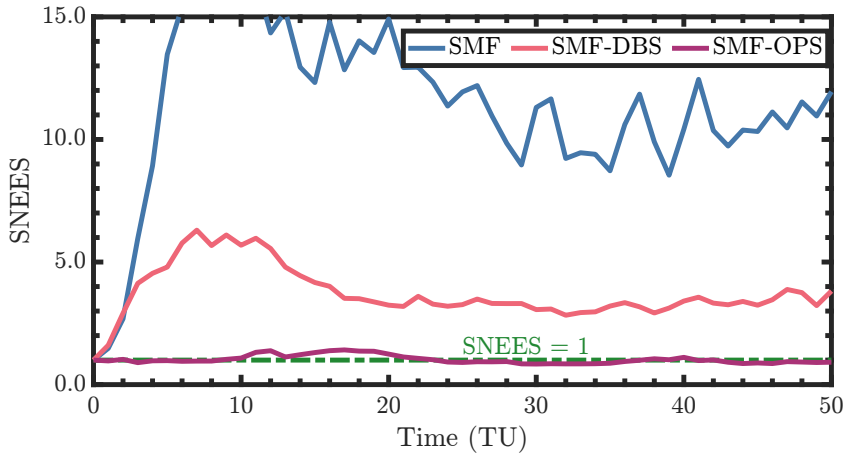

Figure 9. SNEES as a function of time for the three strategies presented. The green dashed line shows a SNEES of one.

is less complicated in terms of tuning, as the only parameters that need adjustment are those in the optimization scheme used to minimize the MCVMD and reduce the set of deterministic points. These are additional factors that must be taken into account when comparing the three strategies.

CONCLUSIONS

In this work, two new techniques are studied for generating a new grid in the Silverman mass filter, a variation of the point mass filter. In the Silverman mass filter, a Gaussian sum filter update is performed before generating a new grid and continuing with the recursion. The standard approach places a new grid at the mean and covariance of these updated points. In contrast, the two new techniques leverage the additional information provided by the updated points to generate a better placement of the new grid.

The first technique uses a density-based clustering approach in which the updated points are first clustered. Once the points have been clustered, the individual mean and covariance of each cluster are obtained. From the mean and covariance of each cluster, a new grid is placed, with the union of all grids considered as the new grid for the next iteration. This technique only considers the location of the updated points to generate the new grid, thus not using the full information provided by the Gaussian sum filter update.

The second technique takes advantage of the full information provided by the update. Since each updated point carries a mean, covariance, and probability mass, a modified distance measure is used to optimally generate a grid from these points. First, deterministic points are sampled from each updated point (representing a Gaussian distribution), and the union of all sampled points is then optimally reduced by minimizing a distance measure. This technique results in better grid placement as it considers the full information from the Gaussian sum update.

The two techniques are evaluated in a static problem using a bimodal posterior distribution. Both techniques are shown to describe the posterior distribution better than the standard approach in the Silverman mass filter. Additionally, the new techniques are evaluated in a sequential filtering problem with the Ikeda map. The clustering-based approach is shown to outperform the standard technique, while the optimal sampling approach was shown to converge to Bayesian inference, producing the most accurate and consistent state estimate among the three approaches considered.

ACKNOWLEDGMENT

This material is based on research sponsored by Air Force Office of Scientific Research (AFOSR) under award number: FA9550-22-1-0419, and by the J. Tinsley Oden Faculty Fellowship Research (Visitors) Program.

REFERENCES

[1] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. John Wiley & Sons, Inc, 2001.

[2] R. Bucy and K. Senne., "Digital synthesis of non-linear filters," *Automatica*, Vol. 7, No. 3, 1971, pp. 287–298.

[3] N. Bergman, *Recursive Bayesian Estimation: Navigation and Tracking Applications*. PhD thesis, Linköping University, 1999.

[4] M. Šimandl, J. Královec, and T. Söderström, "Anticipative grid design in point-mass approach to nonlinear state estimation," *IEEE Transactions on Automatic Control*, Vol. 47, No. 4, 2002, pp. 699–702.

[5] M. Šimandl, J. Královec, and T. Söderström, "Advanced point-mass method for nonlinear state estimation," *Automatica*, Vol. 42, No. 7, 2006, pp. 1133–1145.

[6] J. Duník, O. Straka, and J. Matoušek, "Conditional Density Driven Grid Design in Point-Mass Filter," *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 9180–9184.

[7] Y. Choe and C. G. Park, "Point-Mass Filtering With Boundary Flow and Its Application to Terrain Referenced Navigation," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 57, No. 6, 2021, pp. 3600–3613.

[8] F. Giraldo-Grueso, A. A. Popov, and R. Zanetti, "Gaussian Mixture-Based Point Mass Filtering," *Proceedings of the 2024 27th International Conference on Information Fusion (FUSION)*, Venice, Italy, July 2024.

[9] F. Giraldo-Grueso, A. A. Popov, and R. Zanetti, "Gaussian Mixture-Based Point Mass Filtering with Applications to Terrain Relative Navigation," *Under Review. IEEE Transactions on Aerospace and Electronic Systems*, 2024, pp. 1–15.

[10] 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, 2022, pp. 4179–4197.

[11] A. A. Popov and R. Zanetti, "An Adaptive Covariance Parameterization Technique for the Ensemble Gaussian Mixture Filter," *arXiv preprint arXiv:2212.10323*, 2022.

[12] A. A. Popov and R. Zanetti, "Ensemble gaussian mixture filtering with particle-localized covariances," *Proceedings of the 2023 26th International Conference on Information Fusion (FUSION)*, Charleston, South Carolina, July 2023.

[13] A. A. Popov and R. Zanetti, "Ensemble-localized Kernel Density Estimation with Applications to the Ensemble Gaussian Mixture Filter," *arXiv preprint arXiv:2308.14143*, Vol. 69, 2023.

[14] K. Michaelson, A. A. Popov, R. Zanetti, and K. DeMars, "Particle Flow with a Continuous Formulation of the Nonlinear Measurement Update," *Proceedings of the 2024 27th International Conference on Information Fusion (FUSION)*, Venice, Italy, July 2024.

[15] H. Sorenson and D. Alspach, "Recursive bayesian estimation using gaussian sums," *Automatica*, Vol. 7, No. 4, 1971, pp. 465–479.

[16] D. Alspach and H. Sorenson, "Nonlinear Bayesian estimation using Gaussian sum approximations," *IEEE Transactions on Automatic Control*, Vol. 17, No. 4, 1972, pp. 439–448.

[17] B. W. Silverman, *Density estimation for statistics and data analysis*. Routledge, 2018.

[18] N. V. P. Janssen, J. S. Marron and W. Sarle, "Scale measures for bandwidth selection," *Journal of Nonparametric Statistics*, Vol. 5, No. 4, 1995, pp. 359–380.

[19] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, Portland, Oregon, 1996, p. 226–231.

[20] D. Frisch and U. D. Hanebeck, "Deterministic Gaussian Sampling With Generalized Fibonacci Grids," *Proceedings of the 2021 IEEE International Conference on Information Fusion (FUSION)*, Sun City, South Africa, Nov. 2021.

[21] U. D. Hanebeck and V. Klumpp, "Localized Cumulative Distributions and a multivariate generalization of the Cramér-von Mises distance," *2008 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, 2008, pp. 33–39.

[22] I. Gilitschenski, J. Steinbring, U. D. Hanebeck, and M. Simandl, "Deterministic Dirac mixture approximation of Gaussian mixtures," *17th International Conference on Information Fusion (FUSION)*, 2014, pp. 1–7.

[23] U. D. Hanebeck, "Optimal Reduction of Multivariate Dirac Mixture Densities," *arXiv preprint arXiv:1411.4586*, 2014.

[24] E. Deadman, N. J. Higham, and R. Ralha, "Blocked Schur Algorithms for Computing the Matrix Square Root," *Applied Parallel and Scientific Computing*, 2013, pp. 171–182.

[25] K. Ikeda, "Multiple-valued stationary state and its instability of the transmitted light by a ring cavity system," *Optics Communications*, Vol. 30, No. 2, 1979, pp. 257–261.

[26] K. Ikeda, H. Daido, and O. Akimoto, "Optical Turbulence: Chaotic Behavior of Transmitted Light from a Ring Cavity," *Phys. Rev. Lett.*, Vol. 45, 1980, pp. 709–712.