What are You Weighting For? Improved Weights for Gaussian Mixture Filtering

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

Andrey A. Popov
Oden Institute for Computational Engineering & Sciences
The University of Texas at Austin
Austin, TX, USA
andrey.a.popov@utexas.edu

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

***Abstract*—Gaussian mixture-type filters have become indispensable tools for modeling intricate and nonlinear systems, offering a departure from traditional Gaussian-centric approaches. This work focuses on the critical aspect of accurate weight computation during the measurement incorporation phase of Gaussian mixture filters. The proposed novel approach computes weights by linearizing the measurement model about each component's posterior estimate rather than the the prior, as traditionally done. This work proves equivalence with traditional methods in linear scenarios and empirically demonstrates improved performance in nonlinear cases. Two illustrative examples, the Avocado and Lorenz '63 models, serve to elucidate the advantages of the new weight computation technique by analyzing filter accuracy and efficiency through varying the number of Gaussian mixture components.**

***Index Terms*—statistical estimation, Gaussian mixture filter, weight update**

I. Introduction

Gaussian mixture-type filters mark a departure from traditional Kalman filtering methods, which prove more effective for systems characterized by Gaussian dynamics and observations [1]–[8]. In real-world scenarios, such as weather tracking and orbit determination, nonlinearity and multimodality are common; challenging the viability of Gaussian assumptions [9]–[18]. Addressing these challenges, Gaussian mixture-type filters tackle the intricacies of non-Gaussian state estimation by representing probability distributions as a weighted sum of Gaussian components [19]–[21].

Accurate weight assignment is crucial as it determines the contribution of each Gaussian component to the overall distribution. Correctly computed weights not only ensure alignment with the true state's distribution, but also enhance filter consistency, enabling reliable tracking of dynamic changes and accommodation of uncertainties. Conversely, inaccurate weight computation can lead to subpar filtering performance, causing divergence, filter degeneracy, and ultimately, inaccurate and inconsistent state estimates. This work introduces a novel method for computing the weights of Gaussian mixture-type filters by linearizing the measurement model about each component's posterior estimate rather than the prior. This approach enhances accuracy in weight computation and requires only minimal computational overhead, maintaining the overall efficiency of the process.

This work is organized as follows: section II gives background information on the differences between these two different linearization techniques. Section III proves that, for linear measurement models, the updated weights using this new approach are equivalent to the traditional method of linearization about the prior. Section IV gives an explanation for improved performance for nonlinear measurement models. And in Section V, we offer elegant alternate forms of the improved weights. Lastly, for nonlinear measurement models, in Section VI, our empirical results for two different examples showcase improved performance compared to traditional methods. The first example involves a single measurement update in two dimensions (referred to as the Avocado example), highlighting the fundamental differences between the new approach and traditional Bayesian methods by attempting to estimate an Avocado-shaped posterior distribution. The second example delves into the Lorenz '63 system serving as a dynamic demonstration of the applicability and superiority of the new weighting approach in handling challenging problems.

II. Background

This work computes the weights of Gaussian mixture-type filters by linearizing the measurement model about each component's posterior estimate rather than the prior. This not only improves accuracy but also ensures minimal additional computational burden. In this section, we explore the distinctions between linearizing the measurement model around the prior and the posterior estimates. Additionally, we delve into why utilizing the posterior estimate can contribute to an improvement in the weight update.

Suppose the measurement model is of the form:

$$y = h(x) + \eta, \tag{1}$$

where, η is zero mean with finite covariance R. Equation (1) relates the system's true state x to the sensor measurements y and in practical scenarios is often nonlinear.

This material is based on research sponsored by the Air Force Office of Scientific Research (AFOSR) under agreement number FA9550-23-1-0646, *Create the Future Independent Research Effort (CFIRE)*.

A non-Gaussian prior probability density function (PDF) can be approximated as a weighted sum of n simpler distributions (such as Gaussians) with associated prior weights w_i^-:

$$p_x(x) \approx \sum_{i=1}^{n} w_i^- \; p_{x_i}(x). \tag{2}$$

From it an approximation to the posterior is computed:

$$p_{x|y}(x|y) \approx \sum_{i=1}^{n} w_i^+ \; p_{x_i|y_i}(x|y), \tag{3}$$

where the updated weights of the i-th component (also referred to as mixand) are given by:

$$w_i^+ = \frac{w_i^- \; p_{y_i}(y)}{p_y(y)}, \tag{4}$$

and,

$$p_y(y) \approx \sum_{j=1}^{n} w_j^- \; p_{y_j}(y), \tag{5}$$

$$p_{y_i}(y) = \mathcal{N}(y;\; h(x_i),\; P_{yy}^{(i)}), \tag{6}$$

where x_i and $P_{yy}^{(i)}$ are the i-th component's current state and measurement innovation covariance, respectively. To compute this covariance, a common technique is to use a first-order Taylor series expansion of the measurement model around the i-th component's prior state estimate $\bar{x}_i$:

$$y \approx \bar{y}_i = h(\bar{x}_i) + \bar{H}_i(x - \bar{x}_i) + \eta, \tag{7}$$

where $\bar{H}_i = H(\bar{x}_i)$ is the Jacobian matrix, which captures the gradient of the measurement model with respect to the state evaluated at the prior estimate. It then follows that $\bar{P}_{yy}^{(i)}$ is calculated [1]:

$$\begin{aligned} \bar{P}_{yy}^{(i)} &= \mathbb{E}_{x,y}[\bar{\epsilon}_i \; \bar{\epsilon}_i^T] \\ &= \mathbb{E}_{x,y}[(y - h(\bar{x}_i)) \; (y - h(\bar{x}_i))^T] \\ &\approx \bar{H}_i \bar{P}_i \bar{H}_i^T + R. \end{aligned} \tag{8}$$

The updated weights of the GMM in (4) are typically computed using this linearization. Linearization around the prior is a common assumption whose implications have not been explored in depth and that can hinder the performance of Gaussian Sum Filters (GSF) [19]–[21], Ensemble Gaussian Mixture Filters (EnGMF) [9], [10], [12]–[15], [18], [22], and other Gaussian Mixture Model (GMM)—type filters [23]–[25].

Better weights can be computed by linearizing the measurement model about each component's posterior estimate $\hat{x}_i$ rather than the prior $\bar{x}_i$:

$$y \approx \hat{y}_i = h(\hat{x}_i) + \hat{H}_i(x - \hat{x}_i) + \eta, \tag{9}$$

where $\hat{H}_i = H(\hat{x}_i)$ is the Jacobian matrix, which now captures the gradient of the measurement model with respect to the state evaluated at the posterior estimate. It then follows that $\hat{P}_{yy}^{(i)}$ is calculated [2]:

$$\begin{aligned} \hat{P}_{yy}^{(i)} &= \mathbb{E}_{x,y}[\hat{\epsilon}_i \; \hat{\epsilon}_i^T] \\ &= \mathbb{E}_{x,y}[(y - h(\hat{x}_i)) \; (y - h(\hat{x}_i))^T] \\ &\approx \hat{H}_i \hat{P}^{(i)} \hat{H}_i^T + R - \hat{H}_i K_i R - (\hat{H}_i K_i R)^T, \end{aligned} \tag{10}$$

where $K_i = \bar{P}^{(i)} \bar{H}_i^T \bar{P}_{yy}^{(i)^{-1}}$.

Using the covariances from (8) and (10), two approximations, $p_{\hat{y}}(y)$ and $p_{\bar{y}}(y)$, are explored in lieu of the elusive true distribution $p_y(y)$:

$$p_{\hat{y}}(y) \approx \sum_{j=1}^{n} w_j^- \; p_{\hat{y}_j}(y), \tag{11}$$

$$p_{\hat{y}_i}(y) = \mathcal{N}(y;\; h(\hat{x}_i),\; \hat{P}_{yy}^{(i)}), \tag{12}$$

and,

$$p_{\bar{y}}(y) \approx \sum_{j=1}^{n} w_j^- \; p_{\bar{y}_j}(y), \tag{13}$$

$$p_{\bar{y}_i}(y) = \mathcal{N}(y;\; h(\bar{x}_i),\; \bar{P}_{yy}^{(i)}). \tag{14}$$

This work proposes employing $p_{\hat{y}}(y)$ to compute GMM weights and proves that, in linear cases, both approximations yield equivalent results. Additionally, this work empirically demonstrates, in nonlinear cases, improved GMM filter performance when linearizing the measurement model about the posterior estimates. The improvement is observed when linearizing the measurement model about the posterior estimates offers a more accurate approximation of the truth compared to linearizing about the prior.

III. Proof of Equivalence for Linear Models

This section proves, for linear measurement models, that using the posterior estimate in the weight update leads to weights that are equivalent to those obtained by using the prior.

Theorem III.1. *(Equivalent Weights Under Linear Measurement Models)*

Given the prior weights w_i^- and considering the Gaussian measurement probability distributions $p_{\hat{y}}(y)$ and $p_{\hat{y}_i}(y)$ as computed from (11) and (12), respectively, based on the posterior estimates, and $p_{\bar{y}}(y)$ and $p_{\bar{y}_i}(y)$ as computed from (13) and (14), respectively, using the prior estimates; the traditional weights take the following form:

$$\bar{w}_i^+ = \frac{w_i^- \; p_{\bar{y}_i}(y)}{p_{\bar{y}}(y)}, \tag{15}$$

and the improved weights have the form:

$$\hat{w}_i^+ = \frac{w_i^- \; p_{\hat{y}_i}(y)}{p_{\hat{y}}(y)}, \tag{16}$$

where both $\bar{w}_i^+$ and $\hat{w}_i^+$ share the same prior weights w_i^-. The linear case yields $y = Hx_i + \eta$, $H(\hat{x}_i) = H(\bar{x}_i) = H$ and will give:

$$\hat{w}_i^+ = \frac{w_i^- \; p_{\hat{y}_i}(y)}{p_{\hat{y}}(y)} = \frac{w_i^- \; p_{\bar{y}_i}(y)}{p_{\bar{y}}(y)} = \bar{w}_i^+. \tag{17}$$

[1] See A.38 in the Appendix for expansion of this calculation.
[2] See A.39 in the Appendix for expansion of this calculation.

Proof. Starting from (17), the prior weights cancel on both sides:

$$\frac{p_{\hat{y}_i}(y)}{p_{\hat{y}}(y)} \;=\; \frac{p_{\bar{y}_i}(y)}{p_{\bar{y}}(y)}, \tag{18}$$

then taking the log of both sides yields,

$$\log p_{\hat{y}_i}(y) \;-\; \log p_{\hat{y}}(y) \;=\; \log p_{\bar{y}_i}(y) \;-\; \log p_{\bar{y}}(y). \tag{19}$$

Since the measurement Jacobians are not state dependent, this results in constant measurement innovation covariances such that $\hat{P}_{yy}^{(i)} = \hat{P}_{yy}$ and $\bar{P}_{yy}^{(i)} = \bar{P}_{yy}$. Equation (19) simplifies to:

$$\begin{aligned} &\frac{1}{2}\,\hat{\epsilon}_i^T\;\hat{P}_{yy}^{-1}\;\hat{\epsilon}_i \;+\; \log\sum_{j=1}^{n} w_j^-\exp-\frac{1}{2}\,\hat{\epsilon}_j^T\;\hat{P}_{yy}^{-1}\;\hat{\epsilon}_j \\ =\;&\frac{1}{2}\,\bar{\epsilon}_i^T\;\bar{P}_{yy}^{-1}\;\bar{\epsilon}_i \;+\; \log\sum_{j=1}^{n} w_j^-\exp-\frac{1}{2}\,\bar{\epsilon}_j^T\;\bar{P}_{yy}^{-1}\;\bar{\epsilon}_j. \end{aligned} \tag{20}$$

So now, we only need to prove the following to prove (17):

$$\hat{\epsilon}_i^T\;\hat{P}_{yy}^{-1}\;\hat{\epsilon}_i \;=\; \bar{\epsilon}_i^T\;\bar{P}_{yy}^{-1}\;\bar{\epsilon}_i. \tag{21}$$

Reference [8] has previously established the proof of (21) for linear measurement models. Inherently, this also serves as a natural validation for the proof of (17) in the contributions presented in this work. □

For linear models, we can now conclude that calculating the weights of the posterior distribution from the posterior Gaussian components is equivalent to the traditional method of using the prior Gaussian components.

IV. Nonlinear Models

The nonlinear case $y = h(x_i) + \eta$ results in $\hat{P}_{yy}^{(i)} \neq$ const., $\bar{P}_{yy}^{(i)} \neq$ const., and $H(\hat{x}_i) \neq H(\bar{x}_i)$. This means terms do not simplify as they did for the linear case and we can only conclude that:

$$\hat{w}_i^+ \;=\; \frac{w_i^-\;p_{\hat{y}_i}(y)}{p_{\hat{y}}(y)} \;\neq\; \frac{w_i^-\;p_{\bar{y}_i}(y)}{p_{\bar{y}}(y)} \;=\; \bar{w}_i^+. \tag{22}$$

However, we can demonstrate empirically that:

$$\hat{w}_i^+ \;=\; \frac{w_i^-\;p_{\hat{y}_i}(y)}{p_{\hat{y}}(y)} \;\;\textit{i.b.t}\;\; \frac{w_i^-\;p_{\bar{y}_i}(y)}{p_{\bar{y}}(y)} \;=\; \bar{w}_i^+, \tag{23}$$

where '*i.b.t*' (*is better than*) is used here to mean the new weights $\hat{w}_i^+$ are improved or better than the traditional weights $\bar{w}_i^+$. [3] This is because the posterior is assumed to be a better approximation of the truth, thus the linearization and by extension the weights are a more accurate representation of the truth. The set of examples in this work provide empirical evidence for improving the weights independently for each GMM component; improving the precision of the overall conditional mean $\mathbb{E}_{x|y}[x|y]$.

[3] This does not imply that the improved weights are greater than the traditional weights, but only that they have been improved.

V. Computing the Improved Weights

Some alternate equivalent forms of $\hat{P}_{yy}^{(i)}$ for numerical stability:

$$\begin{aligned} \hat{P}_{yy}^{(i)} &= \hat{H}_i\hat{P}^{(i)}\hat{H}_i^T \;+\; R \;-\; \hat{H}_iK_iR \;-\; (\hat{H}_iK_iR)^T && \textcircled{1}\\ \hat{P}_{yy}^{(i)} &= (\hat{H}_i-\bar{H}_i)\hat{P}^{(i)}(\hat{H}_i-\bar{H}_i)^T \;+\; R\bar{P}_{yy}^{(i)^{-1}}R^T && \textcircled{2}\\ \hat{P}_{yy}^{(i)} &= (\hat{H}_i-\bar{H}_i)\hat{P}^{(i)}(\hat{H}_i-\bar{H}_i)^T && \textcircled{3}\\ &\quad+\;(I-\bar{H}_iK_i)\bar{P}_{yy}^{(i)}(I-\bar{H}_iK_i)^T \end{aligned}$$

Notice that ① is not necessarily enforcing semi-positive definiteness, but is enforcing symmetry. Both ② and ③ enforce semi-positive definiteness and symmetry, however, ② requires an explicit inversion of $\bar{P}_{yy}$. Therefore, the recommended most numerically stable option is ③ which this work is coining the "Joseph Form" [26] of the measurement innovation covariance for the improved weights.

Finally, the improved weights $\hat{w}_i^+$ are:

$$\boxed{\hat{w}_i^+ \;\propto\; w_i^-\;\mathcal{N}(y;\;h(\hat{x}_i),\;\hat{P}_{yy}^{(i)})} \tag{24}$$

And to explicitly compute them:

$$\begin{aligned} \hat{w}_i^+ \;&=\; \frac{w_i^-\;\mathcal{N}(y;\;h(\hat{x}_i),\;\hat{P}_{yy}^{(i)})}{p_{\hat{y}}(y)} \\ &\approx\; \frac{w_i^-\;\mathcal{N}(y;\;h(\hat{x}_i),\;\hat{P}_{yy}^{(i)})}{\sum_{j=1}^{n} w_j^-\;\mathcal{N}(y;\;h(\hat{x}_j),\;\hat{P}_{yy}^{(j)})}. \end{aligned} \tag{25}$$

The computational expense associated with the above is comparable to computing the weights with the prior. In essence, the computational overhead is minimal, making it practically negligible.

VI. Numerical Experiments

The following examples will demonstrate empirically that the new GMM weights proposed can lead to improved performance compared to using the traditional weights.

In the first example, the compared filters will execute a single measurement update in two dimensions. This approach aims to demonstrate the effectiveness of the new weight update scheme, showcasing its broad applicability across all GMM-type filters and decoupling it from the effects of resampling, pruning, propagation, and so forth. As such, the results presented are relevant for GSF, EnGMF, and other GMM-type filters of a similar nature.

In the second example, the compared filters will grapple with the dynamic complexities of the Lorenz '63 system. This system is notorious for its chaotic nature, which means it's highly sensitive to initial conditions. Even minor errors in the initial state or measurement data can trigger substantial divergence in the estimated state. Consequently, accurately estimating the system's state over time becomes a formidable challenge. Adding to the complexity, the Lorenz '63 system is inherently nonlinear. This nonlinearity can result in non-Gaussian probability distributions for the state variables. As a result, conventional filters tailored for linear and Gaussian

systems prove inadequate for providing precise estimates in this scenario. To address these challenges, more advanced estimation techniques like GMM-type filters come into play. In this sensitive dynamic context, the EnGMF stands out as a robust choice when contrasted with a basic Gaussian Sum Filter (GSF). Its ability to handle the chaotic, nonlinear, and non-Gaussian characteristics of the Lorenz '63 system positions it as the preferred option. The EnGMF has been selected, in this example, as the preferred representative filter for conducting a comparative analysis of the different weight updates for GMM-type filters.

A. *Example: Avocado Distribution*

We now demonstrate the improved weights through a single update, two dimensional example visualized by Fig. 1. The results are expressed as *Gaussian Mixture Filter* (GMF), because they are applicable for the GSF, EnGMF, and other GMM-type filters of like kind due to the fact that there isn't any resampling, pruning, propagation, etc.

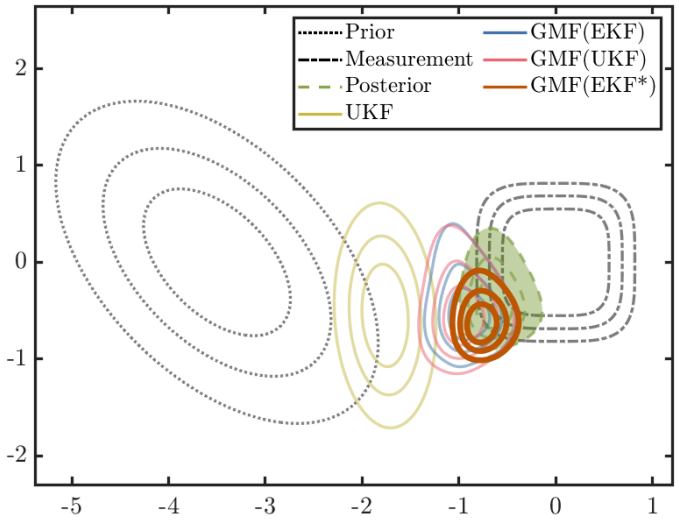

Fig. 1. A plot of the single update, two dimensional example. Comparing the posterior PDF estimates of the UKF, GMF(EKF), GMF(UKF), and GMF(EKF*) averaged over 100 Monte Carlo simulations. The GMM-type filters use 100 components or mixands.

The prior is expressed as a Gaussuan distribution:

$$\mathcal{N}\left(\begin{bmatrix}-3.5\\0\end{bmatrix},\begin{bmatrix}1 & -\frac{1}{2}\\-\frac{1}{2} & 1\end{bmatrix}\right), \tag{26}$$

which is off-center from the origin with the two variables correlated. The nonlinear measurement mapping is of the form

$$h(x) \;=\; \begin{bmatrix}(x_1)^2\\(x_2)^2\end{bmatrix}, \tag{27}$$

with a measured value of

$$y \;=\; \begin{bmatrix}0\\0\end{bmatrix}, \tag{28}$$

which means the measured value was poorly predicted by our prior knowledge. Additionally there is high confidence in the measurement:

$$R \;=\; (0.4^2)I_{2\times 2}, \tag{29}$$

where $I_{2\times 2}$ is the two dimension identity matrix. This example compares the GMF with the new improved weights against the traditional weights. For fair comparison, each GMF has the same number of components, N, with uniform prior weights $w_i^- = 1/N$. For notation, the GMF(EKF) is the GSF/EnGMF/GMM-type filter performing individual EKF updates for each component and is using the traditional weight update described by (15). Conversely, the GMF(EKF*) is the GMM-type filter performing individual EKF updates for each component, but is using the new improved weight update described by (16) and more explicitly by (25). We include the UKF for good measure with tuning parameters $\alpha = 1$, $\beta = 2$, and $\kappa = 3 - m$ ($m = 2$; size of state-space) [3]. Of course UKF's can be tuned to fail, which is why we include the GMF(UKF) that has the same tuning specs as the UKF. It is also using the traditional weight update. The GMF filters (with a sufficient number of components) will outlast a typical UKF for non-Gaussian distributions.

To assess filter performance, the root mean square error (RMSE) and the Kullback–Leibler divergence (KLD) or relative entropy are used. RMSE is computed by

$$\text{RMSE} \;=\; \sqrt{\frac{1}{m}(x-\hat{x})^T(x-\hat{x})}, \tag{30}$$

where m is the size of the state-space, x is the truth, and $\hat{x}$ is the posterior state estimate. KLD is computed by

$$D_{\text{KL}}(P||Q) \;=\; \frac{1}{n_x}\sum_x \frac{1}{2}(\log P(x|y) \;-\; \log Q(x|y))^2, \tag{31}$$

where n_x is the number of discrete values of x on the grid, and $P(x|y)$ and $Q(x|y)$ are the approximated and true posterior PDF's, respectively. In this example, $P(x|y)$ is the estimated PDF produced by the filter after the update represented by a contour PDF in Fig. 1 and $Q(x|y)$ is the truth represented by the green shaded PDF in Fig. 1.

RMSE is a measure of how accurate the state estimate is with respect to the truth. A lower RMSE indicates a more accurate filter. In the context of Table I, using the improved weights, GMF(EKF*), gives better accuracy when compared to the other filters including the traditional weight update, GMF(EKF); suggesting improved performance.

KLD, on-the-other-hand, is a measure of the dissimilarity between two probability distributions. Though it is not a true distance metric as it is not symmetric and does not satisfy the triangle inequality, it is useful in quantifying how the true probability distribution differs from the approximating distribution. A lower KLD indicates there is less information needed to match the approximated distribution to the truth. Table I shows that using the improved weights can substantially decrease KLD; again improving filter performance.

TABLE I

NUMERICAL OUTPUT OF THE SINGLE UPDATE, TWO DIMENSIONAL AVOCADO EXAMPLE. COMPARING THE ROOT MEAN SQUARE ERROR AND THE KULLBACK–LEIBLER DIVERGENCE FOR THE UKF, GMF(EKF), GMF(UKF), AND GMF(EKF*). AVERAGED OVER 100 MONTE CARLO SIMULATIONS. THE GMM-TYPE FILTERS USE 100 COMPONENTS OR MIXANDS.

	RMSE	KLD
UKF	0.9613	—
GMF(EKF)	0.2899	12.594
GMF(UKF)	0.3113	16.640
GMF(EKF*)	**0.2378**	**0.8226**

To aid in our justification, we varied the number of GMM components in the GMF. The RMSE and KLD results vs number of components are shown in Fig. 2 and Fig. 3, respectively. Noticeably, the UKF suffers and is not visible in the plotting frames due to it's approximation of Gaussian distributions. This was understood previously from it's poor performance in Fig. 1 and Table I. For the GMM-type filters, with smaller number of components, there is a clear indication of improved performance when using the new weights over the traditional ones. As the number of components increases, a noteworthy trend emerges: the filters utilizing traditional weights, namely GMF(EKF) and GMF(UKF), appear to gradually converge towards the performance exhibited by the filter employing the improved weights, denoted as GMF(EKF*). As the number grows, then the prior covariance becomes smaller, the update becomes smaller, and hence the difference between prior and posterior also becomes smaller. In the limit as the number of components goes to infinity, the two should be identical.

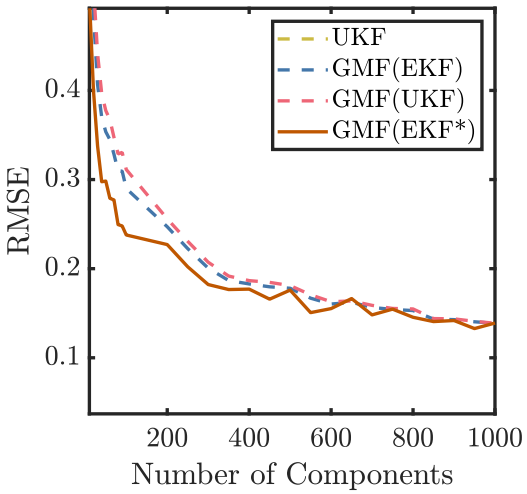

Fig. 2. Comparing the impact on state estimate accuracy with-respect-to the truth, represented by root mean square error (RMSE), by varying the number of GMM components in the single update Avocado example. Featuring the UKF, GMF(EKF), GMF(UKF), and GMF(EKF*). The UKF results are plotted out of frame. Plots are averaged across 100 Monte Carlo simulations for each GMM component test case.

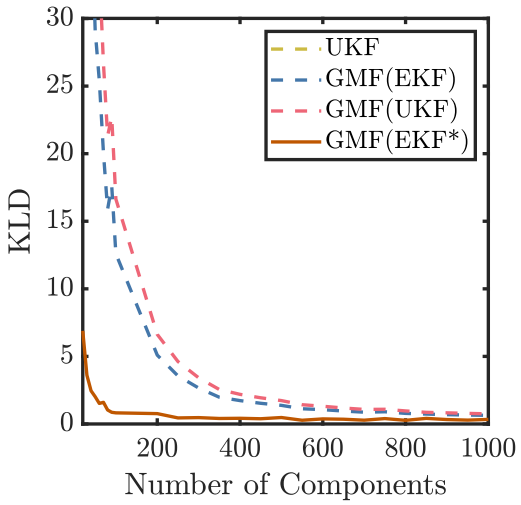

Fig. 3. Comparing the impact on PDF accuracy with-respect-to the truth, represented by Kullback–Leibler divergence (KLD), by varying the number of GMM components in the single update Avocado example. Featuring the UKF, GMF(EKF), GMF(UKF), and GMF(EKF*). The UKF results are plotted out of frame. Plots are averaged across 100 Monte Carlo simulations for each GMM component test case.

B. Example: Lorenz '63

This dynamic example will now demonstrate the strength of the improved GMM weighting scheme. The EnGMF is being used to tackle the complicated Lorenz '63 system [27] to compare the improved weights against the traditional weights. We again include the UKF for good measure with tuning parameters $\alpha = 1$, $\beta = 2$, and $\kappa = 3 - m$ ($m = 3$; size of state-space) [3].

The Lorenz '63 dynamics used are

$$\begin{aligned} \dot{x}_1 &= 10(x_2 - x_1) + \nu_1 \\ \dot{x}_2 &= x_1(28 - x_3) - x_2 + \nu_2 \\ \dot{x}_3 &= x_1 x_2 - \frac{8}{3} x_3 + \nu_3, \end{aligned} \tag{32}$$

where process noise $\nu = [\nu_1 \ \nu_2 \ \nu_3]^T \sim \mathcal{N}(\mathbf{0}, Q)$ and

$$Q = 4 \times 10^{-3} \begin{bmatrix} 0.86 & 0.86 & -0.01 \\ 0.86 & 1.1 & -0.01 \\ -0.01 & -0.01 & 1.02 \end{bmatrix}. \tag{33}$$

The states are propagated using (32) in the classical 4th-order Runge-Kutta integrator. The initial states are

$$x_0 = [0 \ 1 \ 0]^T. \tag{34}$$

The initial uncertainty is

$$P_0 = I_{3\times 3}. \tag{35}$$

For the nonlinear measurement, we take the range from one of the equilibrium points:

$$y = \sqrt{(x_1 - 6\sqrt{2})^2 + (x_2 - 6\sqrt{2})^2 + (x_3 - 27)^2} + \eta \tag{36}$$

where measurement noise $\eta \sim \mathcal{N}(\mathbf{0}, R)$ and

$$R = \frac{1}{100}. \tag{37}$$

A measurement is recorded every $\Delta t = 0.5$ time units. For analysis, the first 100 estimates are discarded since the initial conditions are not yet on the attractor. Fig. 4 shows the trajectory for the simulation.

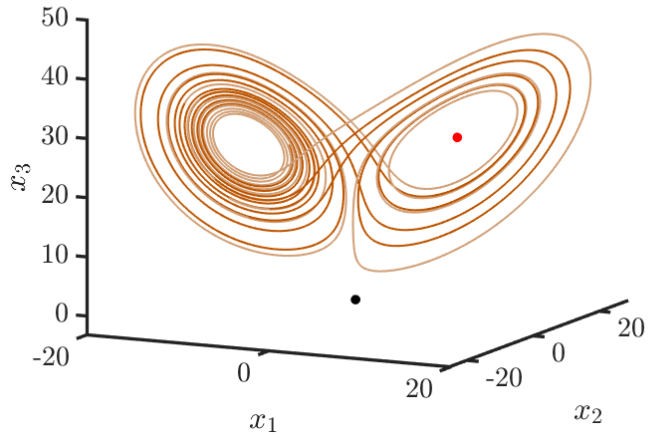

Fig. 4. A plot of Lorenz '63 trajectory with the black marker as the initial state x_0 and the red marker as the equilibrium point where the range is mapped to. The initial state is not on the attractor which motivates discarding the first 100 estimates during analysis.

The dynamics form a distinct shape like wings on a butterfly. Similar to the example in Section VI-A, this is a challenging scenario for linear filters like the UKF. The long propagation time step enables the states to spread out between measurement updates. The prior covariance is large, but the measurements are highly accurate.

Again, we varied the number of GMM components (or *ensemble sizes* to be pedantic for the EnGMF). The RMSE verses the number of components and wall-clock-time verses RMSE results are shown in Fig. 5 and Fig. 6, respectively. Again, the UKF suffers and is not visible in the plotting frames due to it's approximation of Gaussian distributions and the system's large propagation time step.

It is evident that the EnGMF(EKF*) outperforms the other filters in terms of both accuracy and efficiency. Notably, its computational performance surpasses that of the EnGMF(UKF), offering a more cost-effective alternative, especially considering the lack of tuning requirements for the EnGMF(EKF*). This finding underscores its practical advantages in terms of efficiency and resource utilization.

Moreover, the study reveals nuanced insights into the comparative performance of the different EnGMF's concerning the number of components. In scenarios with smaller numbers, the EnGMF(EKF*) demonstrates improved accuracy compared to its counterparts. However, as the number of components expands, the accuracy of all three filters converges just like in Section VI-A. This observation suggests that, under certain conditions, the performance gains of EnGMF(EKF*) may diminish with more components, warranting further investigation into the changes of the number of components and filter accuracy. These nuanced findings contribute valuable insights to the understanding of filter performance in varying ensemble configurations.

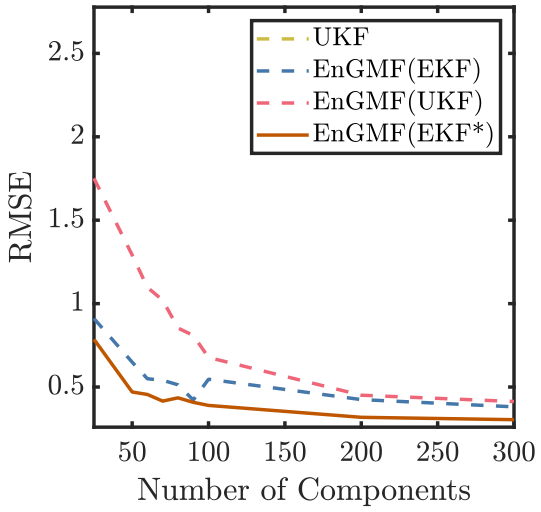

Fig. 5. Comparing the impact on state estimate accuracy with-respect-to the truth, represented by root mean square error (RMSE), by varying the number of component members in the Lorenz '63 example. Featuring the UKF, EnGMF(EKF), EnGMF(UKF), and EnGMF(EKF*). The UKF results are plotted out of frame. Plots are averaged across 100 Monte Carlo simulations for each GMM component test case.

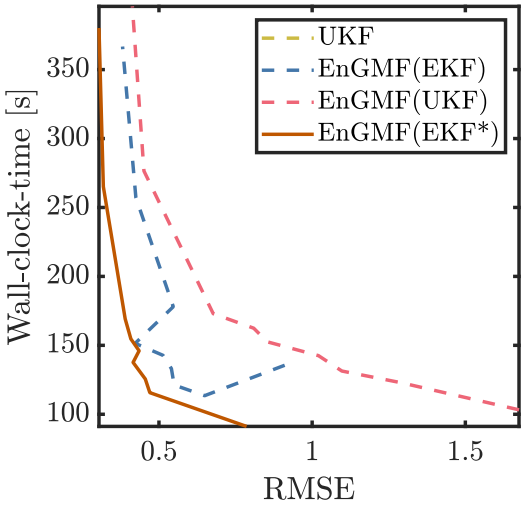

Fig. 6. Comparing the impact on computational performance, represented by the wall-clock-time in seconds, against the root mean square error (RMSE) by varying the number of component members in the Lorenz '63 example. Featuring the UKF, EnGMF(EKF), EnGMF(UKF), and EnGMF(EKF*). The UKF results are plotted out of frame. Plots are averaged across 100 Monte Carlo simulations for each GMM component test case.

VII. Conclusion

This work establishes a novel method for computing weights in Gaussian mixture-type filters, showcasing its equivalence to traditional methods in linear scenarios and improved performance in nonlinear cases. The empirical evaluation using the Avocado and Lorenz '63 models demonstrates the practical advantages of the proposed approach, notably outperforming traditional methods in terms of accuracy and efficiency. The exploration of varying the number of Gaussian mixture components adds valuable insights into filter performance under different configurations. This work demonstrates improved accuracy and computational efficiency, especially in scenarios with fewer components, underlining its practical applicability and resource effectiveness.

Further research could explore the validity of (23), which hinges on the assumption that the linearization about the posterior is a better approximation of the truth compared to the prior. It would be interesting to examine the changes in performance when this assumption is not met.

Appendix

Some of the calculations in this work are too long to fit nicely within the main body. For the reader's reference, these are the step-by-step expanded derivations of the measurement innovation covariances for the i-th Gaussian component.

Expansion of $\bar{P}_{yy}^{(i)}$:

$$\begin{aligned}
\bar{P}_{yy}^{(i)} &= \mathbb{E}_{x,y}[\bar{\epsilon}_i\ \bar{\epsilon}_i^T] \\
&= \mathbb{E}_{x,y}[(y - h(\bar{x}_i))\ (y - h(\bar{x}_i))^T] \\
&= \mathbb{E}_{x,y}[(\bar{H}_i(x - \bar{x}_i) + \eta)\ (\bar{H}_i(x - \bar{x}_i) + \eta)^T] \\
&= \mathbb{E}_{x,y}[\bar{H}_i(x - \bar{x}_i)(x - \bar{x}_i)^T \bar{H}_i^T \\
&\quad + 2\bar{H}_i(x - \bar{x}_i)\eta^T + \eta\eta^T] \\
&\approx \bar{H}_i \bar{P}_i \bar{H}_i^T + R.
\end{aligned} \tag{A.38}$$

Expansion of $\hat{P}_{yy}^{(i)}$:

$$\begin{aligned}
\hat{P}_{yy}^{(i)} &= \mathbb{E}_{xy}[\hat{\epsilon}_i\ \hat{\epsilon}_i^T] \\
&= \mathbb{E}_{xy}[(y - h(\hat{x}_i))\ (y - h(\hat{x}_i))^T] \\
&= \mathbb{E}_{xy}[\hat{H}_i(x - \hat{x}_i) + \eta)(\hat{H}_i(x - \hat{x}_i) + \eta)^T] \\
&= \mathbb{E}_{xy}[(\hat{H}_i(x - \hat{x}_i)(x - \hat{x}_i)^T \hat{H}_i^T \\
&\quad + \hat{H}_i(x - \hat{x}_i)\eta^T \\
&\quad + \eta(\hat{H}_i(x - \hat{x}_i))^T + \eta\eta^T] \\
&= \hat{H}_i \mathbb{E}_{xy}[(x - \hat{x}_i)(x - \hat{x}_i)^T] \hat{H}_i^T \\
&\quad + \hat{H}_i \mathbb{E}_{xy}[(x - \hat{x}_i)\eta^T] \\
&\quad + \mathbb{E}_{xy}[\eta(x - \hat{x}_i)^T] \hat{H}_i^T + \mathbb{E}_{xy}[\eta\eta^T] \\
&\approx \hat{H}_i \hat{P}^{(i)} \hat{H}_i^T + R - \hat{H}_i K_i R - (\hat{H}_i K_i R)^T,
\end{aligned} \tag{A.39}$$

this results from solving

$$\begin{aligned}
\mathbb{E}_{xy}[(x - \hat{x}_i)(x - \hat{x}_i)^T] &\approx \hat{P}^{(i)} \\
&= \bar{P}^{(i)} - \bar{P}^{(i)} \bar{H}_i^T \bar{P}_{yy}^{(i)^{-1}} \bar{H}_i \bar{P}^{(i)} \\
&= \bar{P}^{(i)} - K_i \bar{H}_i \bar{P}^{(i)},
\end{aligned} \tag{A.40}$$

$$\begin{aligned}
&\mathbb{E}_{xy}[(x - \hat{x}_i)\eta^T] \\
&\quad = \mathbb{E}_{xy}[(x - \bar{x}_i - K_i(y - h(\bar{x}_i)))\eta^T] \\
&\quad \approx -K_i R,
\end{aligned} \tag{A.41}$$

$$\mathbb{E}_{xy}[\eta\eta^T] = R. \tag{A.42}$$

References

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

[2] 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.

[3] R. van der Merwe and E. Wan, "Sigma-point Kalman filters for probabilistic inference in dynamic state-space models," Ph.D. dissertation, The faculty of the OGI School of Science & Engineering at Oregon ..., 2004.

[4] A. Gelb *et al.*, *Applied optimal estimation*. MIT press, 1974.

[5] T. Lefebvre, H. Bruyninckx, and J. De Schuller, "Comment on" a new method for the nonlinear transformation of means and covariances in filters and estimators"[with authors' reply]," *IEEE transactions on automatic control*, vol. 47, no. 8, pp. 1406–1409, 2002.

[6] I. Arasaratnam, S. Haykin, and R. J. Elliott, "Discrete-time nonlinear filtering algorithms using Gauss–Hermite quadrature," *Proceedings of the IEEE*, vol. 95, no. 5, pp. 953–977, 2007.

[7] I. Arasaratnam and S. Haykin, "Cubature Kalman filters," *IEEE Transactions on automatic control*, vol. 54, no. 6, pp. 1254–1269, 2009.

[8] R. Zanetti, "Adaptable recursive update filter," *Journal of Guidance, Control, and Dynamics*, vol. 38, no. 7, pp. 1295–1300, 2015.

[9] 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.

[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, pp. 4179–4197, 2022.

[11] S. Yun, N. Ravago, B. L. Reifler, R. Zanetti, and B. A. Jones, "Generalized labeled multi-Bernoulli filter with kernel-based ensemble Gaussian mixture filtering for orbit determination with sparse data," in *AMOS Conf. Proc*, 2022.

[12] A. A. Popov and R. Zanetti, "Ensemble Gaussian mixture filtering with particle-localized covariances," in *2023 26th International Conference on Information Fusion (FUSION)*. IEEE, 2023, pp. 1–7.

[13] ——, "An adaptive covariance parameterization technique for the ensemble Gaussian mixture filter," *arXiv preprint arXiv:2212.10323*, 2022.

[14] ——, "Ensemble-localized kernel density estimation with applications to the ensemble Gaussian mixture filter," *arXiv preprint arXiv:2308.14143*, 2023.

[15] D. Durant, A. A. Popov, and R. Zanetti, "MCMC EnGMF for sparse data orbit determination," in *Astrodynamics Specialist Conference, Big Sky, MT*, no. 23(356). AAS/AIAA, 2023.

[16] Z. Li, "Applications of Gaussian mixture model to weather observations," Ph.D. dissertation, The faculty of the School of Electrical & Computer Engineering at The University of Oklahoma ..., 2011.

[17] 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.

[18] B. L. Reifler, A. A. Popov, B. A. Jones, and R. Zanetti, "Large-scale space object tracking in a proliferated LEO scenario," in *2023 26th International Conference on Information Fusion (FUSION)*. IEEE, 2023, pp. 1–8.

[19] H. W. Sorenson and D. L. Alspach, "Recursive Bayesian estimation using Gaussian sums," *Automatica*, vol. 7, no. 4, pp. 465–479, 1971.

[20] D. Alspach and H. Sorenson, "Nonlinear Bayesian estimation using Gaussian sum approximations," *IEEE transactions on automatic control*, vol. 17, no. 4, pp. 439–448, 1972.

[21] S. Yun and R. Zanetti, "Sequential Monte Carlo filtering with Gaussian mixture sampling," *Journal of Guidance, Control, and Dynamics*, vol. 42, no. 9, pp. 2069–2077, 2019.

[22] B. W. Silverman, *Density estimation for statistics and data analysis*. Routledge, 1998.

[23] A. S. Stordal, H. A. Karlsen, G. Nævdal, H. J. Skaug, and B. Vallès, "Bridging the ensemble Kalman filter and particle filters: the adaptive Gaussian mixture filter," *Computational Geosciences*, vol. 15, pp. 293–305, 2011.
[24] D. Raihan and S. Chakravorty, "Particle Gaussian mixture filters-i," *Automatica*, vol. 98, pp. 331–340, 2018.
[25] ——, "Particle Gaussian mixture filters-ii," *Automatica*, vol. 98, pp. 341–349, 2018.
[26] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons, 2001.
[27] E. N. Lorenz, "Deterministic nonperiodic flow," *Journal of atmospheric sciences*, vol. 20, no. 2, pp. 130–141, 1963.