AN AUGMENTED GAUSSIAN SUM FILTER THROUGH A MIXTURE DECOMPOSITION

Kostas Tsampourakis[*] *and Víctor Elvira*[*‡]

[*] School of Mathematics, University of Edinburgh, United Kingdom
[‡] The Alan Turing Institute, United Kingdom

ABSTRACT

Bayesian filtering is an approach to the inference problem for state-space models (SSM), arising in many disciplines of science and engineering. The well known Gaussian filters tackle the problem using Gaussian approximations of the distributions of the hidden state in the model. Gaussian filters however are unable to track multimodal distributions that commonly arise in complex dynamical systems. The *Gaussian sum filter* (GSF) addresses this problem by using Gaussian mixture approximations of the filtering distribution. The GSF however has important limitations, requiring small width of the component covariances for good performance. Moreover, in many SSMs the estimates provided by the GSF blow up, due to covariance inflation. In this paper, we propose a way of controlling the covariances of the underlying Gaussian mixture. Our approach relies on a well known Gaussian identity, which helps us break down each component of the GSF, the *parent*, into several *children* components of smaller width. These smaller components are propagated through the nonlinearities by local linearization, which results in a smaller error than in the standard GSF. To reduce the resulting mixture we use resampling. We refer to our novel approach as augmented Gaussian sum filter (AGSF). We demonstrate the advantages of our approach using a toy example, for which the *extended Kalman filter* (EKF) and GSF perform poorly due to covariance inflation.

***Index Terms*—** Nonlinear filtering; State-space models; Gaussian sum filter;

1. INTRODUCTION

The problem of nonlinear filtering has a long history and is of interest to many disciplines of science and engineering. It is a generalization of the linear filtering problem, addressed more than 60 years ago by the celebrated Kalman filter [1], to systems governed by nonlinear dynamics. The nonlinear problem is analytically intractable and approximations are needed. Gaussian filtering, in which filtering distributions are approximated by normal densities, is a common approach [2]. The best-used representatives of this approach are the *extended Kalman filter* (EKF), which utilizes local Taylor approximations [3], the *unscented Kalman filter* (UKF) [4], which approximates the moments using the *unscented transform* [5], the *quadrature* and *cubature* Kalman filters [6–9], based on Gaussian quadrature methods [10, 11], and the Gaussian particle filters [12, 13], based on importance sampling [14].

The complexity of nonlinear systems however cannot be captured accurately by Gaussian distributions. Thus, high-performing filters require going beyond the simple Gaussian paradigm [15–19]. Gaussian sum filters [20] go in that direction by implementing a bank of EKFs to approximate the filtering pdf as a Gaussian mixture. Gaussian sums, or mixtures as they are commonly known, provide a consistent way to represent complex pdfs because of their completeness, since pdfs can be approximated arbitrarily well by a Gaussian mixture [21, 22]. However, the accuracy of GSFs depends on the width of the Gaussian components that are propagated, compared to the characteristic widths of the nonlinearities of the SSM [12]. Their performance is then hindered when the noise covariances are large, while the functions are highly nonlinear. Recent work [23,24] tries to address this limitation by imposing *linear matrix inequalities* to the mixture covariances. This approach requires the complicated machinery of matrix inequalities which is cumbersome to implement.

In this paper, we propose a new way of controlling the covariances of the Gaussian mixtures of the algorithm. Our approach uses the Gaussian convolution identity and Monte-Carlo sampling, to break down the *parent* components of the mixture into *children* Gaussians, placed at random points, with covariances that are parameters of the convolution identity. The *children* components are used for the local linearization, i.e., they are propagated using EKF steps. Since we control their covariance, we have leverage on the linearization error of the EKF step. This decomposition can be applied both in the prediction and update steps, leading to a new Gaussian mixture filtering algorithm which outperforms other Gaussian sum filters.

The rest of the paper is structured as follows. In Section 2, we provide the required background. In Section 3, we introduce the Gaussian augmentation and show how it can be used in the derivation of a novel Gaussian sum filtering algorithm. The paper concludes with numerical results in Section 4.

The work of V. E. is supported by the *Agence Nationale de la Recherche* of France under PISCES (ANR-17-CE40-0031-01), the Leverhulme Research Fellowship (RF-2021-593), and by ARL/ARO under grant W911NF-22-1-0235.

2. BACKGROUND

2.1. Bayesian Filtering

Consider an additive, nonlinear state space model, defined by

$$\mathbf{x}_t = \mathbf{f}(\mathbf{x}_{t-1}) + \mathbf{q}_t, \quad (1)$$

$$\mathbf{y}_t = \mathbf{g}(\mathbf{x}_t) + \mathbf{r}_t, \quad (2)$$

where $\mathbf{x}_t \in \mathbb{R}^{d_x}$ and $\mathbf{y}_t \in \mathbb{R}^{d_y}$ are the state and observation vectors, while $\mathbf{q}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{Q})$ and $\mathbf{r}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$ are additive noise vectors. The goal of *Bayesian filtering* is to compute the marginal posterior distribution, or *filtering distribution* (FD) $p(\mathbf{x}_t|\mathbf{y}_{1:t})$ of the state $\mathbf{x}_t$ given the observation history up to time t. This can be done recursively, and the FD at time t can be computed in two steps, given the FD at time $t-1$ and the new observation $\mathbf{y}_t$. These steps are, (i) the computation of the predictive distribution of the state $\mathbf{x}_t$ given observations up to time $t-1$ (*prediction step*),

$$p(\mathbf{x}_t|\mathbf{y}_{1:t-1}) = \int p(\mathbf{x}_t|\mathbf{x}_{t-1}) p(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1}) d\mathbf{x}_{t-1}, \quad (3)$$

and (ii) the Bayesian update of this distribution upon receiving the new observation at time t (*update step*):

$$p(\mathbf{x}_t|\mathbf{y}_{1:t}) = \frac{p(\mathbf{y}_t|\mathbf{x}_t) p(\mathbf{x}_t|\mathbf{y}_{1:t-1})}{\int p(\mathbf{y}_t|\mathbf{x}_t) p(\mathbf{x}_t|\mathbf{y}_{1:t-1}) d\mathbf{x}_t}. \quad (4)$$

The prediction and update steps can be computed exactly for a linear state space model. When the functions are nonlinear we define approximations of the FD, and compute approximately the prediction and update steps of Eqs. (3)-(4).

2.2. Gaussian Filters

Gaussian filtering methods approximate the filtering distribution by a Gaussian [2]. This approach has proved very successful and has been extensively used, e.g., in the form of the EKF.

These approximations match the first and second moments of the approximate normal distribution to the true underlying distributions. For a given pair of random variables $\mathbf{a}$ and $\mathbf{b}$, where $\mathbf{a}$ is assumed to be Gaussian, $\mathbf{a} \sim \mathcal{N}(\boldsymbol{\mu}_\mathbf{a}, \boldsymbol{\Sigma}_\mathbf{a})$, and $\mathbf{b}$ is assumed to have well defined mean $\mathbb{E}[\mathbf{b}]$, and covariance $\mathrm{Var}[\mathbf{b}]$, the Gaussian moment matching approximation of the joint is

$$\mathcal{N}\left(\begin{pmatrix} \boldsymbol{\mu}_\mathbf{a} \\ \boldsymbol{\mu}_\mathbf{b} \end{pmatrix}, \begin{pmatrix} \boldsymbol{\Sigma}_\mathbf{a} & \mathbf{C}_\mathbf{ab} \\ \mathbf{C}_\mathbf{ab}^T & \mathbf{S}_\mathbf{b} \end{pmatrix} \right), \quad (5)$$

where $\boldsymbol{\mu}_\mathbf{b} \approx \mathbb{E}[\mathbf{b}], \mathbf{S}_\mathbf{b} \approx \mathrm{Var}[\mathbf{b}], \mathbf{C}_\mathbf{ab} \approx \mathrm{Cov}[\mathbf{a}, \mathbf{b}]$. In the prediction step, $\mathbf{a} = \mathbf{x}_{t-1}$ and $\mathbf{b} = \mathbf{x}_t$, while in the update step $\mathbf{a} = \mathbf{x}_t$ and $\mathbf{b} = \mathbf{y}_t$.

There is a large variety of ways to approximate the moments $\boldsymbol{\mu}_\mathbf{b}, \boldsymbol{\Sigma}_\mathbf{b}$, and $\mathbf{C}_\mathbf{ab}$, leading to a large variety of Gaussian filtering methods. The EKF, UKF, and Cubature KF are all Gaussian filters.

Moreover, multiple Gaussian filters can be run in parallel to construct Gaussian mixture approximations of the FD. Such filters are known as Gaussian sum filters (GSF) [12, 21].

3. AUGMENTED GAUSSIAN SUM FILTER

We propose a novel approach based on the augmentation of the Gaussian integrals that appear in the Bayesian filtering equations, Eqs. (3), (4), with auxiliary latent variables. The augmentation allows us to break down the components of a GSF into *children* components that have smaller covariance. We use this decomposition to construct a novel GSF.

3.1. Augmenation of a Gaussian random variable

The augmentation is based on the Gaussian integral identity,

$$\mathcal{N}(\mathbf{a}|\boldsymbol{\mu}_\mathbf{a}, \boldsymbol{\Sigma}_\mathbf{a}) = \int \mathcal{N}(\mathbf{a}|\boldsymbol{\Gamma}\mathbf{z} + \mathbf{c}, \boldsymbol{\Delta}) \mathcal{N}(\mathbf{z}|\boldsymbol{\mu}_\mathbf{z}, \boldsymbol{\Sigma}_\mathbf{z}) d\mathbf{z}, \quad (6)$$

where $\mathbf{a} \in \mathbb{R}^{d_a}$ and $\mathbf{z} \in \mathbb{R}^{d_z}$ is an auxiliary variable that has been introduced, in relation to $\mathbf{a}$. For the identity to hold, we must have $\boldsymbol{\mu}_\mathbf{a} = \boldsymbol{\Gamma}\boldsymbol{\mu}_\mathbf{z} + \mathbf{c}$, $\boldsymbol{\Sigma}_\mathbf{a} = \boldsymbol{\Gamma}\boldsymbol{\Sigma}_\mathbf{z}\boldsymbol{\Gamma}^T + \boldsymbol{\Delta}$ and also $\boldsymbol{\Delta} \succeq \mathbf{0}, \boldsymbol{\Sigma}_\mathbf{z} \succeq \mathbf{0}$. Without loss of generality, we make the choice $d_z = d_a, \mathbf{c} = \mathbf{0}, \boldsymbol{\mu}_\mathbf{z} = \boldsymbol{\mu}_\mathbf{a}, \boldsymbol{\Gamma} = \mathbf{I}$, and $\boldsymbol{\Sigma}_\mathbf{z} = \boldsymbol{\Sigma}_\mathbf{a} - \boldsymbol{\Delta}$ which we justify in Section 3.3. With this choice the only free parameter is the covariance matrix $\boldsymbol{\Delta}$, which must satisfy $\boldsymbol{\Sigma}_\mathbf{a} \succeq \boldsymbol{\Delta} \succeq \mathbf{0}$. We use the augmentation of Eq. (6) to make a Monte-Carlo based, Gaussian mixture approximation of $\mathcal{N}(\mathbf{a}|\boldsymbol{\mu}_\mathbf{a}, \boldsymbol{\Sigma}_\mathbf{a})$ as follows,

$$\mathcal{N}(\mathbf{a}|\boldsymbol{\mu}_\mathbf{a}, \boldsymbol{\Sigma}_\mathbf{a}) \simeq \frac{1}{N} \sum_{n=1}^{N} \mathcal{N}(\mathbf{a}|\mathbf{z}_n, \boldsymbol{\Delta}), \quad (7)$$

where $\mathbf{z}_n \sim \mathcal{N}(\boldsymbol{\mu}_\mathbf{z}, \boldsymbol{\Sigma}_\mathbf{z})$.

3.2. Augmented Gaussian sum filter

We now derive a novel Gaussian sum filtering algorithm by using the augmentation of the previous section both in the prediction and update steps.

The new filter is summarized in Alg. 1. It is initialized as a mixture of M Gaussians and iterates over prediction and update steps, whose derivation is sketched below. Note that at each step the number of components of the original mixture is multiplied. To ensure that our algorithm will run on fixed memory we do a resampling step at the end of the update step.

Prediction. We assume that the FD at time $t-1$ is given by a Gaussian mixture, of M components $\mathcal{N}(\boldsymbol{\mu}_{t-1}^{(m)}, \boldsymbol{\Sigma}_{t-1}^{(m)})$ with corresponding weights $w_{t-1}^{(m)}$. Thus using Eq. (3), the predictive distribution of $\mathbf{x}_t$ is approximated as the mixture

$$p(\mathbf{x}_t|\mathbf{y}_{1:t-1}) \simeq \sum_{m=1}^{M} w_{t-1}^{(m)} p_m(\mathbf{x}_t), \quad (8)$$

where

$$p_m(\mathbf{x}_t) = \int \mathcal{N}(\mathbf{x}_t|\mathbf{f}(\mathbf{x}_{t-1}), \mathbf{Q})\mathcal{N}(\mathbf{x}_{t-1}|\boldsymbol{\mu}_{t-1}^{(m)}, \boldsymbol{\Sigma}_{t-1}^{(m)})d\mathbf{x}_{t-1}. \tag{9}$$

The EKF (GSF) approximates this integral using a Taylor expansion of $\mathbf{f}$ around $\boldsymbol{\mu}_{t-1}^{(m)}$. We delay this step and at this stage, using the augmentation, we approximate this integral by a Gaussian mixture, with smaller covariance matrices. Unlike the GSF, we first decompose the *parent* components, $\mathcal{N}(\boldsymbol{\mu}_{t-1}^{(m)}, \boldsymbol{\Sigma}_{t-1}^{(m)})$, using the augmentation in Sec. 3.1 and then we apply an EKF step, i.e., linearization.

More precisely, we express $p_m(\mathbf{x}_t)$ of Eq. (9) as a Gaussian mixture with components of covariance $\boldsymbol{\Delta}_m$, placed at random points (see Fig. 2). Finally we apply an EKF prediction step to the resulting Gaussian components, to approximate the predictive pdf of Eq. (8) by Eq. (11) of Alg. 1. The corresponding step of the GSF is illustrated in Fig. 1

Update. Using the update equation, Eq. (4), for a prior given by Eq. (11), we find that the posterior of $\mathbf{x}_t$ is proportional to

$$\sum_{m=1}^{M}\sum_{n=1}^{N_m} w_{mn}\mathcal{N}(\mathbf{y}_t|\mathbf{g}(\mathbf{x}_t), \mathbf{R})\mathcal{N}(\mathbf{x}_t|\boldsymbol{\mu}_{mn}^{-}, \boldsymbol{\Sigma}_{mn}^{-}). \tag{10}$$

The EKF approximates the terms of this mixture by the posterior of a joint Gaussian linear approximation. We instead use again the augmentation of Sec. 3.1. For each *parent* Gaussian $\mathcal{N}(\boldsymbol{\mu}_{mn}^{-}, \boldsymbol{\Sigma}_{mn}^{-})$, we introduce samples $\mathbf{s}_{mn\ell}, \ell = 1, \ldots, L_{mn}$ and covariance matrices $\boldsymbol{\Lambda}_{mn}$, and write the original Gaussian as a mixture of *children* components placed at the points $\mathbf{s}_{mn\ell}$ with covariance $\boldsymbol{\Lambda}_{mn}$. Finally to the components indexed by the triple $(mn\ell)$, we apply linearization in the update step to approximate the FD as the mixture in Eq. (14) in Alg. 1.

Resampling The recursive part of the prediction/update Alg. 1 takes as input a mixture of M Gaussians and returns a mixture of $\sum_{m=1}^{M}\sum_{n=1}^{N_m} L_{mn} > M$ components. Hence if used recursively it would produce a growing mixture which is impractical. To have a practical algorithm we need to reduce the number of components to a chosen number. There are many possibilities for decreasing the number of components of a Gaussian mixture [25]. In our implementation we use multinomial resampling at the end of each update.

3.3. Selection of parameters and discussion

Here we address the choice of the augmentation parameters introduced in Section 3.1. Note that the identity of Eq. (6) holds for any choice of the parameters, $d_z, \mathbf{c}, \boldsymbol{\Gamma}, \boldsymbol{\Delta}, \boldsymbol{\mu}_{\mathbf{z}}$, and $\boldsymbol{\Sigma}_{\mathbf{z}}$ that satisfy the constraints. There is however redundancy in this set of parameters, since if we allow for degenerate covariance matrices $\boldsymbol{\Delta}, \boldsymbol{\Sigma}_{\mathbf{z}}$, we effectively affect the dimensionality d_z as well as the matrix $\boldsymbol{\Gamma}$. Therefore there is no loss of generality in the choices we make in Section 3.1. Moreover, we have not specified a way of selecting the covariance $\boldsymbol{\Delta}$. In general, criteria can be derived that relate the choice of $\boldsymbol{\Delta}$ with the local geometry of the nonlinearity. Such considerations will be made in an extended version of this work.

Algorithm 1 Augmented Gaussian sum filter

1: **Initialization:** $\{w_0^{(m)}, \boldsymbol{\mu}_0^{(m)}, \boldsymbol{\Sigma}_0^{(m)}\}_{m=1}^{M}$.
2: for $t = 1, \ldots, T$:
3: **Prediction**

$$\widehat{p}(\mathbf{x}_t|\mathbf{y}_{1:t-1}) = \sum_{m=1}^{M}\sum_{n=1}^{N_m} w_{mn}\mathcal{N}(\mathbf{x}_t|\boldsymbol{\mu}_{mn}^{-}, \boldsymbol{\Sigma}_{mn}^{-}), \tag{11}$$

where

$$\boldsymbol{\mu}_{mn}^{-} = \mathbf{f}(\mathbf{z}_{mn}), \tag{12}$$
$$\boldsymbol{\Sigma}_{mn}^{-} = \nabla\mathbf{f}(\mathbf{z}_{mn})\boldsymbol{\Delta}_m\nabla\mathbf{f}(\mathbf{z}_{mn})^T + \mathbf{Q}, \tag{13}$$

for $\mathbf{z}_{mn} \sim \mathcal{N}(\boldsymbol{\mu}_{t-1}^{(m)}, \boldsymbol{\Sigma}_{t-1}^{(m)} - \boldsymbol{\Delta}_m), w_{mn} = w_{t-1}^{(m)}/N_m$, $n = 1, \ldots, N_m; m = 1, \ldots, M$.

4: **Update**

$$\widehat{p}(\mathbf{x}_t|\mathbf{y}_{1:t}) = \sum_{m=1}^{M}\sum_{n=1}^{N_m}\sum_{\ell=1}^{L_{mn}} w_{mnl}\mathcal{N}(\mathbf{x}_t|\boldsymbol{\mu}_{mn\ell}, \boldsymbol{\Sigma}_{mn\ell}), \tag{14}$$

where

$$w_{mnl} \propto (w_{mn}/L_{mn})\mathcal{N}(\mathbf{y}_t|\boldsymbol{\mu}_{\mathbf{y},mn\ell}, \mathbf{S}_{\mathbf{y},mn\ell}), \tag{15}$$
$$\boldsymbol{\mu}_{mn\ell} = \mathbf{s}_{mn\ell} + \mathbf{G}_{mn\ell}(\mathbf{y}_t - \boldsymbol{\mu}_{\mathbf{y},mn\ell}), \tag{16}$$
$$\boldsymbol{\Sigma}_{mn\ell} = \boldsymbol{\Lambda}_{mn} - \mathbf{G}_{mn\ell}\mathbf{S}_{\mathbf{y},mn\ell}\mathbf{G}_{mn\ell}^T, \tag{17}$$

with

$$\boldsymbol{\mu}_{\mathbf{y},mn\ell} = \mathbf{g}(\mathbf{s}_{mn\ell}), \tag{18}$$
$$\mathbf{S}_{\mathbf{y},mn\ell} = \nabla\mathbf{g}(\mathbf{s}_{mn\ell})\boldsymbol{\Lambda}_{mn}\nabla\mathbf{g}(\mathbf{s}_{mn\ell})^T + \mathbf{R}, \tag{19}$$
$$\mathbf{G}_{mn\ell} = \boldsymbol{\Lambda}_{mn}\nabla\mathbf{g}(\mathbf{s}_{mn\ell})^T\mathbf{S}_{\mathbf{y},mn\ell}^{-1}, \tag{20}$$

where $\mathbf{s}_{mn\ell} \sim \mathcal{N}(\mathbf{s}|\boldsymbol{\mu}_{mn}^{-}, \boldsymbol{\Sigma}_{mn}^{-} - \boldsymbol{\Lambda}_{mn})$ for $m = 1, \ldots, M, n = 1, \ldots, N_m$ and $\ell = 1, \ldots, L_{mn}$.

5: **Resampling:** For $m' = 1, \ldots, M$, sample a triplet $(mn\ell)$ with probability $w_{mn\ell}$ and set

$$\boldsymbol{\mu}_t^{(m')} = \boldsymbol{\mu}_{mn\ell}\ ;\ \boldsymbol{\Sigma}_t^{(m')} = \boldsymbol{\Sigma}_{mn\ell}. \tag{21}$$

Set $w_t^{(m)} = 1/M$.

Finally we note that in general the points $\mathbf{z}_n$ can be selected using any rule, random or deterministic [26], such as quadrature rules, importance sampling [27] or quasi-Monte Carlo rules [28]. This paves the way to other novel algorithms that are not explored in this paper due to space limitations.

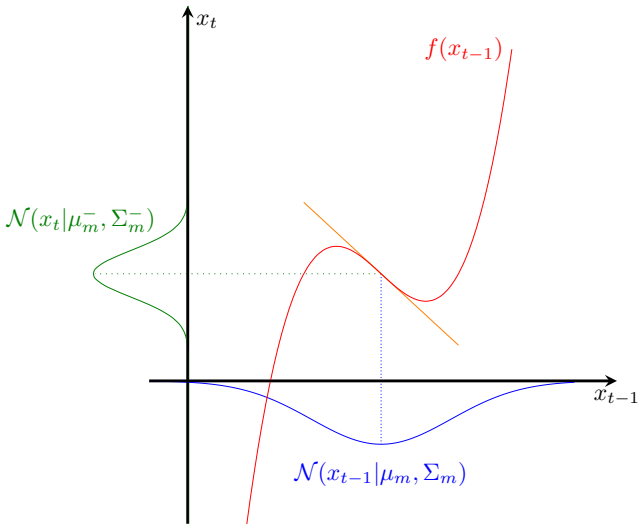

Fig. 1: Graphical illustration of the prediction step for a component of the GSF. The original parent component (blue) $\mathcal{N}(x_{t-1}|\mu_m, \Sigma_m)$, propagated through the nonlinearity (red), via local linearization (the tangent line in orange), to give the components of the predictive mixture (green).

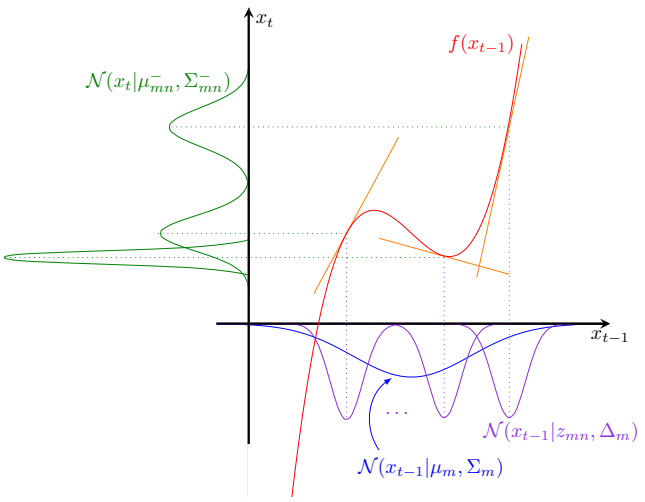

Fig. 2: Graphical illustration of the AGSF prediction step for the same component as in Fig. 1. The original parent component (blue), is broken down into children components (violet). Each children is propagated through the nonlinearity (red), via different linearization (orange lines), to give the components of the predictive mixture (green).

4. NUMERICAL EXPERIMENT

In this section, we present a numerical example to compare the performance of our approach (AGSF) with the EKF and GSF. We show that the augmentation can be used to give numerically stable estimates when the EKF and GSF fail.

We consider the one-dimensional, nonlinear state space model given by the equations, $x_t = \sin(10x_{t-1}) + q_t$ and $y_t = ax_t^2 + r_t$, where $x_t, y_t \in \mathbb{R}$, $q_t \sim \mathcal{N}(0, Q)$ and $r_t \sim \mathcal{N}(0, R)$. For each value of $a \in \{0.01, 0.1, 1\}$ and for $Q = 1, R = 1$, we perform $N_s = 100$ simulations of the SSM for $T = 100$ time steps. For the GSF algorithm, we used 10 components. To run the AGSF Alg. 1, we used $M = 5$, $N = 5$, and $L = 5$ components. We set the auxiliary covariances to be fixed and equal to $\mathbf{\Lambda}_{mn} = 1, \mathbf{\Delta}_m = 0.2$, for $m = 1, \ldots, M; n = 1, \ldots, N$. We used the bootstrap particle filter (BPF) [29] with 1000 particles and multinomial resampling as our ground truth. We compute the RMSE of the filtered estimates compared to the BPF estimates. We average the RMSE over the $N_s = 100$ simulations and report the average RMSE and the execution times in Table 1.

For $a = 0.01$ and $a = 0.1$ the EKF and GSF algorithms fail at capturing the underlying dynamics, as witnessed by the large RMSEs. Their failure is due to a rapid inflation of the covariance matrices of the underlying Gaussian distributions. This rapid inflation is driven by a large derivative of the dynamical model together with a small Kalman gain.

Moreover, we see that the AGSF algorithm does not suffer from the covariance inflation problem. This is easily understood from Eq. (13), by observing that the covariance

Table 1: For $a \in \{0.01, 0.1, 1\}$ we report the RMSE $\pm$ one standard deviation, and the running time in seconds $\pm$ one standard deviation. Lower values are better.

	$a = 0.01$		$a = 0.1$		$a = 1$	
	RMSE	time(s)	RMSE	time(s)	RMSE	time(s)
EKF	154.24±61.27	0.40±0.03	15.77±6.55	0.40±0.05	4.30±1.59	0.48±0.12
GSF	150.43±74.67	1.06±0.05	17.57±13.60	1.06±0.08	3.66±3.19	1.23±0.24
AGSF	**0.35±0.02**	3.33±0.25	**0.38±0.03**	3.32±0.26	**0.52±0.06**	3.88±0.80

$\mathbf{\Delta}_m$ multiplies the square of the derivative of the dynamical model. Since we have the freedom to choose $\mathbf{\Delta}_m$ we can use it to balance out any destabilizing inflation that the derivative of f might cause.

5. CONCLUSIONS

We have proposed a novel *augmented Gaussian sum filter* (AGSF). The AGSF uses an augmentation of Gaussian pdfs which are broken down into children components. The covariances of the children components can be tuned to achieve smaller linearization error, and allow more control of the covariance evolution, compared to standard methods. With the aid of simulation we have demonstrated how the tuning of those parameters can result in stable algorithms, when the EKF and GSF fail. The AGSF opens up many interesting possibilities for designing efficient, adaptive nonlinear filters.

References

[1] R. E. Kalman, "A new approach to linear filtering and prediction problems," 1960.

[2] S. Särkkä, *Bayesian filtering and smoothing*, Cambridge university press, 2013.

[3] G. L. Smith, S. F. Schmidt, and L. A. McGee, *Application of statistical filter theory to the optimal estimation of position and velocity on board a circumlunar vehicle*, National Aeronautics and Space Administration, 1962.

[4] E. A. Wan and R. Van Der Merwe, "The unscented kalman filter for nonlinear estimation," in *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No. 00EX373)*. Ieee, 2000, pp. 153–158.

[5] S. Julier and J. K. Uhlmann, "A general method for approximating nonlinear transformations of probability distributions," Tech. Rep., 1996.

[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] K. Ito and K. Xiong, "Gaussian filters for nonlinear filtering problems," *IEEE Transactions on Automatic Control*, vol. 45, no. 5, pp. 910–927, 2000.

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

[9] O. Straka, J. Duník, and V. Elvira, "Importance gauss-hermite gaussian filter for models with non-additive non-gaussian noises," in *2021 IEEE 24th International Conference on Information Fusion (FUSION)*. IEEE, 2021, pp. 1–7.

[10] F. B. Hildebrand, *Introduction to numerical analysis*, Courier Corporation, 1987.

[11] V. Elvira, L. Martino, and P. Closas, "Importance gaussian quadrature," *IEEE Transactions on Signal Processing*, vol. 69, pp. 474–488, 2020.

[12] J. H. Kotecha and P. M. Djuric, "Gaussian sum particle filtering," *IEEE Transactions on signal processing*, vol. 51, no. 10, pp. 2602–2612, 2003.

[13] K. Comandur, Y. Li, and S. Nannuru, "Particle flow gaussian particle filter," in *2022 25th International Conference on Information Fusion (FUSION)*. IEEE, 2022, pp. 1–6.

[14] V. Elvira, L. Martino, D. Luengo, and M. F. Bugallo, "Generalized Multiple Importance Sampling," *Statistical Science*, vol. 34, no. 1, pp. 129 – 155, 2019.

[15] P. M. Djuric, J. H. Kotecha, J. Zhang, Y. Huang, T. Ghirmai, M. F. Bugallo, and J. Miguez, "Particle filtering," *IEEE signal processing magazine*, vol. 20, no. 5, pp. 19–38, 2003.

[16] A. Doucet, A. M. Johansen, et al., "A tutorial on particle filtering and smoothing: Fifteen years later," *Handbook of nonlinear filtering*, vol. 12, no. 656-704, pp. 3, 2009.

[17] V. Elvira, L. Martino, M. F. Bugallo, and P. M. Djuric, "Elucidating the auxiliary particle filter via multiple importance sampling [lecture notes]," *IEEE Signal Processing Magazine*, vol. 36, no. 6, pp. 145–152, 2019.

[18] N. Branchini and V. Elvira, "Optimized auxiliary particle filters: adapting mixture proposals via convex optimization," in *Uncertainty in Artificial Intelligence*. PMLR, 2021, pp. 1289–1299.

[19] J. Duník, O. Straka, and M. Šimandl, "Stochastic integration filter," *IEEE Transactions on Automatic Control*, vol. 58, no. 6, pp. 1561–1566, 2013.

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

[22] B. D. Anderson and J. B. Moore, *Optimal filtering*, Courier Corporation, 2012.

[23] M. L. Psiaki, "Gaussian mixture nonlinear filtering with resampling for mixand narrowing," *IEEE Transactions on Signal Processing*, vol. 64, no. 21, pp. 5499–5512, 2016.

[24] M. L. Psiaki, J. R. Schoenberg, and I. T. Miller, "Gaussian sum reapproximation for use in a nonlinear filter," *Journal of Guidance, Control, and Dynamics*, vol. 38, no. 2, pp. 292–303, 2015.

[25] D. F. Crouse, P. Willett, K. Pattipati, and L. Svensson, "A look at gaussian mixture reduction algorithms," in *14th International Conference on Information Fusion*, 2011, pp. 1–8.

[26] P. L'Ecuyer and A. B. Owen, *Monte Carlo and Quasi-Monte Carlo Methods 2008*, Springer, 2009.

[27] C. P. Robert and G. Casella, *Monte Carlo statistical methods*, vol. 2, Springer, 1999.

[28] R. E. Caflisch, "Monte carlo and quasi-monte carlo methods," *Acta numerica*, vol. 7, pp. 1–49, 1998.

[29] N. J. Gordon, D. J. Salmond, and A. F. Smith, "Novel approach to nonlinear/non-gaussian bayesian state estimation," in *IEE proceedings F (radar and signal processing)*. IET, 1993, vol. 140, pp. 107–113.