



DBSCAN-based particle Gaussian mixture filters

Sukkeun Kim ^{a, }, Mengwei Sun ^{a, }, Ivan Petrunin ^{a, }, Hyo-Sang Shin ^{a, b, }, *

^a Faculty of Engineering and Applied Sciences, Cranfield University, Cranfield, MK43 0AL, UK

^b Cho Chun Shik Graduate School of Mobility, Korea Advanced Institute of Science and Technology, Daejeon, 34141, Republic of Korea

ARTICLE INFO

MSC:

60J05

60J20

Keywords:

Particle filter

Particle Gaussian mixture filter

Gaussian mixture

DBSCAN

State estimation

ABSTRACT

This study addresses nonlinear and non-Gaussian state estimation problems where the particle filter (PF) exhibits the impoverishment issue. This issue arises from the discretisation of the continuous posterior distribution of the state and the use of importance sampling, where the true distribution of the state is unknown. In this study, we propose density-based spatial clustering of applications with noise (DBSCAN)-based particle Gaussian mixture (PGM) filters: the PGM-DS and PGM-DU filters, where DS indicates the PGM filter with DBSCAN and DU indicates the PGM filter with DBSCAN and the unscented transform (UT). These filters assume the posterior distribution of the state to be a Gaussian mixture model (GMM) and sample particles from this GMM. At every time step, the particles are clustered into multiple Gaussian components using DBSCAN, the components are updated with the Kalman/linear minimum mean squared error (LMMSE) update, and the GMM is reconstructed with the updated means and covariances. The proposed filters are tested in three numerical simulation scenarios and compared with other state-of-the-art nonlinear filters. The results show enhanced performance and robustness across the tested simulation scenarios, with lower computational cost compared to the other filters.

1. Introduction

The particle filter (PF), or a sequential Monte Carlo (SMC) method, is an estimator that approximates the probability density function (PDF) of the state using random discrete samples [1–3]. The characteristics of Monte Carlo (MC), the nature of Bayesian inference and the Markov property allow the PF to be one of the most commonly utilised estimators in nonlinear and non-Gaussian scenarios compared to other nonlinear filters. Because the posterior distribution of the state is typically unknown in closed form, PFs rely on samples drawn from an importance distribution to approximate it. In other words, the performance of the PF heavily depends on the choice of the importance density. Furthermore, by approximating the posterior distribution with discrete samples, the filter can suffer from additional issues such as particle degeneracy and impoverishment [4]. Particle degeneracy and impoverishment can be critical to the performance of the PF since they reduce the diversity of the random samples, which is the key strength of the PF in handling non-linearity. Particle degeneracy occurs when only a few particles have large weights, while the others have negligible weights. The most common way to address particle degeneracy is the resampling method, which samples particles in the high-weight region, as proposed in [4–7]. However, resampling algorithms can lead to the impoverishment prob-

lem, where the particles collapse into a few particles after resampling. One simple solution for this issue is to jitter the particles [4].

Multiple recent works have attempted to resolve these issues by hybridising PF with machine learning methods [8] or by incorporating kernel density approaches into PF [9]. For example, Imbiriba and Closas mitigated the particle degeneracy issue by utilising a Gaussian process (GP) to enhance the posterior distribution approximation of the PF [10], while Comandur et al. developed the Gaussian flow Gaussian PF (FGFPPF), which integrates an invertible Gaussian flow method into the GPF [11]. Kuptamete et al. mitigated the particle impoverishment issue by adopting a genetic algorithm-based particle herding scheme [12], while Kim et al. developed the shifting kernel PF (SKPF), which incorporates shifting kernel functions during resampling [13]. One of the main approaches to overcome the aforementioned issues is to incorporate Gaussian mixture model (GMM) or a weighted sum of Gaussians into Bayes filters to approximate the posterior distribution of the state. Approximating the posterior distribution of the state using GMM can improve the performance of the filter in nonlinear scenarios and can alleviate the aforementioned issues. Alspach and Sorenson proposed a Gaussian sum filter, which approximates the PDF of the state with a weighted sum of Gaussians [14]. Kotecha and Djurić proposed a Gaussian sum PF as a bank of PFs with GMM state PDF with the expectation-

* Corresponding author at: Faculty of Engineering and Applied Sciences, Cranfield University, Cranfield, MK43 0AL, UK.
E-mail address: h.shin@cranfield.ac.uk (H.-S. Shin).

maximisation (EM) algorithm [15]. Psiaki proposed a “blob” filter based on a GMM for highly nonlinear scenarios [16,17]. The name “blob” was used in contrast to the “particle” where the distribution of the state was assumed to be a GMM. One of the main differences between this approach and previous GMM filters is that the resampling step is placed between time propagation and measurement update to enforce the maximum size of the variance of each Gaussian component using a matrix inequality (MI) upper bound. Yun and Zanetti proposed fuzzy clustering methods to approximate the GMM in mixture filters, demonstrating improved performance compared to the EM algorithm and K-means clustering [18]. Raihan and Chakravorty proposed particle Gaussian mixture (PGM) filters, the PGM1 and PGM1-UT filters, using a modified K-means clustering to construct the GMM based on the clustering results [19–21]. This approach was proposed to overcome one of the major limitations in previous research which is the number of components in the GMM is fixed. However, it still faces the issue of high computational cost to ensure a minimum number of samples in each cluster for calculating the sample covariance of each component. Moreover, the modified K-means clustering still has additional issues, as it requires a predefined maximum number of clusters.

To this end, we propose PGM filters based on density-based spatial clustering of applications with noise (DBSCAN) [22,23]: the PGM-DS and PGM-DU filters, where DS indicates the PGM filter with DBSCAN and DU indicates the PGM filter with DBSCAN and unscented transform (UT). These DBSCAN-based PGM filters offer multiple advantages: a predefined number of clusters is not required, and DBSCAN ensures a minimum number of particles in each cluster by its inherent characteristics. The proposed filters are tested in three numerical simulation scenarios and compared with multiple nonlinear filters, including the unscented Kalman filter (UKF), the ensemble Kalman filter (EnKF) [24], the PF, the adaptive kernel Kalman filter (AKKF) [25] and the PGM1 and PGM1-UT filters.

1. New PGM filters with clustering based on DBSCAN are proposed.
2. The proposed filters are compared to other state-of-the-art nonlinear filters and show enhanced performance.
3. The robustness analysis confirms the robustness of the filters to measurement noise and the DBSCAN design parameters.

The rest of this paper is composed as follows: Section 2 briefly introduces the Bayes filter and GMM for the reader's comprehension. Section 3 proposes the DBSCAN-based PGM filters and Section 4 provides the result of numerical simulations of the DBSCAN-based PGM filters. Finally, we conclude this study with future work in Section 5.

2. Preliminaries

2.1. Bayes filters

Bayes filters are widely applied in many estimation problems. The state of the system of interest can be estimated using the system dynamics and the measurement model. This process can be denoted as follows, for the discrete-time case:

$$\mathbf{x}_{k|k-1} = f_k(\mathbf{x}_{k-1|k-1}, \mathbf{w}_k), \quad (1)$$

$$\mathbf{z}_k = h_k(\mathbf{x}_{k|k-1}, \mathbf{v}_k), \quad (2)$$

where the subscript $k = 1, 2, 3 \dots$ indicates the time instance, $\mathbf{x}_k$ is the state vector, f_k is the discrete system dynamics, $\mathbf{w}_k$ is the process noise, $\mathbf{z}_k$ is the measurement vector, h_k is the discrete measurement model and $\mathbf{v}_k$ is the measurement noise. The general form of the PDF of the recursive Bayes filter can be calculated as follows:

$$p(\mathbf{x}_k | \mathbf{z}_{1:k}) = \frac{p(\mathbf{z}_k | \mathbf{x}_k) p(\mathbf{x}_k | \mathbf{z}_{1:k-1})}{\int p(\mathbf{z}_k | \mathbf{x}_k) p(\mathbf{x}_k | \mathbf{z}_{1:k-1}) d\mathbf{x}_k}. \quad (3)$$

However, an analytical solution for this integration is often not available, except in some special cases such as the linear Gaussian case, as in the well-known linear Kalman filter [26]. In nonlinear and non-Gaussian cases, an approximation must be made. For example, the extended Kalman filter utilises Jacobian matrices for piecewise linearisation and UKF utilises sigma points to obtain an approximate solution. One of the most versatile methods of approximating the solution in nonlinear and non-Gaussian scenarios is PF or SMC, which approximates the posterior distribution of the state using random samples called “particles”. The PF computes the integral in Eq. (3) using sample representation or MC integration. This approximate integral can be calculated as follows:

$$p(\mathbf{x}_k | \mathbf{z}_{1:k}) \simeq \sum_{i=1}^{N_p} w_i \delta(\mathbf{x}_k - \mathbf{x}_k^i), \quad (4)$$

where N_p is the number of samples, w_i is the weight of the i^{th} sample which sums to 1, $\delta(\cdot)$ is the Dirac delta function and $\mathbf{x}_k^i$ is the i^{th} sample.

2.2. Gaussian mixture model

The GMM, which represents a distribution as a weighted sum of multiple Gaussian components, can be used to approximate the arbitrary unknown distribution of a PF. The nature of the Gaussian distribution, which is characterised by the first two moments, enables a simple and effective calculation of the distribution of interest. The PDF of the state $\mathbf{x}_k$ can be approximated by a GMM with N_m components as follows:

$$GMM(\mathbf{x}_k) = \sum_{n=1}^{N_m} w_n \mathcal{N}(\mu_n, \sigma_n^2), \quad (5)$$

where w_n is the weight of the n^{th} Gaussian component, which sums to 1, and μ_n and σ_n^2 are the mean and variance of the n^{th} component, respectively. Here, the Gaussian component $\mathcal{N}(\mu_n, \sigma_n^2)$ is defined as follows:

$$\mathcal{N}(\mu_n, \sigma_n^2) = \frac{1}{\sigma_n \sqrt{2\pi}} e^{-\frac{1}{2} \left(\frac{\mathbf{x} - \mu_n}{\sigma_n} \right)^2}. \quad (6)$$

More details on the GMM can be found in [17,19]. A GMM can be constructed using the EM algorithm [27] with given data points, and an arbitrarily shaped PDF can be approximated using it. In this study, the GMM is utilised to approximate the unknown PDF of the state.

3. DBSCAN-based PGM filters

DBSCAN-based PGM filters utilise DBSCAN to cluster the particles, and each cluster is considered a Gaussian component of the GMM. Each component is then updated using the measurement, and the updated components are used to construct the GMM to approximate the posterior distribution of the state. At the next time step, new particles are sampled from this GMM, eliminating the resampling step commonly used in most other PFs.

3.1. DBSCAN

DBSCAN is a clustering algorithm that has three main advantages over other clustering algorithms: 1) it doesn't require a fixed number of clusters, 2) it can cluster arbitrarily shaped clusters and 3) it has the ability to handle noise. It is called density-based since clusters are defined based on the density of points within a given distance, ϵ , from core points. The two design parameters to compute the density are ϵ , which specifies the radius around a given point, and $minPts$, which is the minimum number of points required around that point to determine whether it is a core point. This ensures that clusters contain at least $minPts$ points and reduces the computational burden compared to constrained K-means clustering, which requires optimisation to guarantee a

minimum number of particles in each cluster. The parameter ϵ can be selected using a k-distance graph, whilst the parameter $minPts$ is typically chosen according to the rule of thumb $minPts \geq N + 1$. Given these design parameters, the clustering process of DBSCAN can be summarised as follows [22]:

1. Identify the core points based on ϵ and $minPts$.
2. Form connected subgraphs of core points.
3. Assign reachable points to their nearest neighbouring cluster.
4. Label the remaining points, which are neither core nor reachable, as noise.

3.2. PGM-DS and PGM-DU filters

Two variants of PGM filters are proposed here, both based on the Kalman/linear minimum mean squared error (LMMSE) update. The PGM-DS filter is a baseline method that calculates the mean and covariance of each Gaussian component directly from clustered particles and updates them using the Kalman update. The PGM-DU filter uses the UT (as in the UKF [28]), employing sigma points to calculate the mean and covariance of each component. With the LMMSE update, the updated mean $\mu_{k|k}^n$ and covariance $P_{k|k}^n$ of the n^{th} component at time step k can be calculated as follows:

$$\mu_{k|k}^n = \mu_{k|k-1}^n + K_k^n (z_k - \mathbb{E}[h(X_k^n)]), \quad (7)$$

$$P_{k|k}^n = P_{k|k-1}^n - K_k^n P_{zx,k}^n, \quad (8)$$

$$K_k^n = (P_{zx,k}^n)^T (P_{zz,k}^n)^{-1}, \quad (9)$$

where X_k^n is the set of particles clustered in the n^{th} component, K_k^n is the Kalman gain of the n^{th} component, and the cross-covariance $P_{zx,k}^n$ and innovation covariance $P_{zz,k}^n$ are calculated as follows:

$$P_{zx,k}^n = \mathbb{E}[(h(X_k^n) - \mathbb{E}[h(X_k^n)])(X_k^n - \mathbb{E}[X_k^n])^T], \quad (10)$$

$$P_{zz,k}^n = \mathbb{E}[(h(X_k^n) - \mathbb{E}[h(X_k^n)])(h(X_k^n) - \mathbb{E}[h(X_k^n)])^T]. \quad (11)$$

In the case of the PGM-DU filter, the sigma points of the n^{th} component, $\mathcal{X}_{k|k-1}^n$, are computed using the UT prior to the update step, as follows:

$$\mathcal{X}_{k|k-1}^n = [\mathbb{E}[X_k^n], \mathbb{E}[X_k^n] \pm \sqrt{(L + \lambda) P_{k|k-1}^n}], \quad (12)$$

where L is the dimension of the state, $\lambda = \alpha^2(L + \kappa) - L$ with predefined parameters α and κ , and $\sqrt{(L + \lambda) P_{k|k-1}^n}$ denotes the matrix square root, typically computed via Cholesky decomposition. The mean and covariance can be calculated using the sigma points as follows:

$$\mu_{k|k-1}^n = \sum_{i=0}^{2L} W_i^{(m)} \mathcal{X}_{i,k|k-1}^n, \quad (13)$$

$$P_{k|k-1}^n = \sum_{i=0}^{2L} W_i^{(c)} [\mathcal{X}_{i,k|k-1}^n - \mu_{k|k-1}^n] [\mathcal{X}_{i,k|k-1}^n - \mu_{k|k-1}^n]^T, \quad (14)$$

where $W_i^{(m)}$ and $W_i^{(c)}$ are the mean and covariance weights of each sigma point, defined as follows:

$$W_0^{(m)} = \frac{\lambda}{L + \lambda}, \quad (15)$$

$$W_0^{(c)} = \frac{\lambda}{L + \lambda} + (1 - \alpha^2 + \beta), \quad (16)$$

$$W_i^{(m)} = W_i^{(c)} = \frac{1}{2(L + \lambda)}, \quad i = 1, \dots, 2L. \quad (17)$$

The sigma points of n^{th} component in the measurement space can be calculated by $\mathcal{Z}_{i,k|k-1}^n = h(\mathcal{X}_{i,k|k-1}^n)$ and the mean can be calculated as follows:

Algorithm 1 PGM-DS and PGM-DU filters.

Given: $f_k, h_k, x_0, P_0, Q, R, z_k, N_p, \epsilon, minPts, \alpha, \beta, L, \lambda$

- 1: **for** k in K **do**
- 2: Draw samples from given initial x_0 and P_0
- 3: Time propagation with given system dynamics f_k
- 4: Clustering using DBSCAN with ϵ and $minPts$
- 5: **if** Distance between clusters $\leq$ threshold **then**
- 6: Merge clusters
- 7: **end if**
- 8: **if** PGM-DS **then**
- 9: Update with Eqs. (7)–(11)
- 10: **else**
- 11: Calculate sigma points with Eqs. (12)–(18)
- 12: Update with Eqs. (7)–(9) and Eqs. (19)–(20)
- 13: **end if**
- 14: Construct a GMM with updated mean and covariance with Eq. (5)
- 15: Sample N_p new particles from the GMM for the next time step
- 16: **end for**

$$z_k^n = \sum_{i=0}^{2L} W_i^{(m)} \mathcal{Z}_{i,k|k-1}^n. \quad (18)$$

The update can be performed using Eqs. (7) and (8), where the cross-covariance $P_{zx,k}^n$ and the innovation covariance $P_{zz,k}^n$, computed using the sigma points, are as follows:

$$P_{zx,k}^n = \sum_{i=0}^{2L} W_i^{(c)} [\mathcal{Z}_{i,k|k-1}^n - z_k^n] [\mathcal{X}_{i,k|k-1}^n - \mu_{k|k-1}^n]^T, \quad (19)$$

$$P_{zz,k}^n = \sum_{i=0}^{2L} W_i^{(c)} [\mathcal{Z}_{i,k|k-1}^n - z_k^n] [\mathcal{Z}_{i,k|k-1}^n - z_k^n]^T. \quad (20)$$

The PGM filters have a straightforward structure – the posterior distribution of the state is approximated by a GMM and the GMM is reconstructed using the updated mean and covariance of each Gaussian component. Time propagation is performed directly using the system dynamics, as shown in Eq. (1), similarly to the PF. After the time propagation, DBSCAN clusters the propagated particles into multiple components with associated weights. It is worth noting that DBSCAN performs clustering based on density, meaning that a predefined number of clusters is not required. In other words, the number of components in the GMM is automatically determined by DBSCAN based on particle density. In addition, the design parameter ϵ in DBSCAN is chosen empirically, while the other parameter, $minPts$, is set to $N + 1$ in this implementation. At this stage, if components are sufficiently close to one another, they may be merged into a single cluster. For the update step, the PGM-DS filter calculates the mean and covariance directly from clustered particles, whilst the PGM-DU filter uses the sigma points to compute the mean and covariance. Each component with mean and covariance is then updated using the Kalman update, as shown in Eqs. (7) and (8). For the next time step, particles are sampled from the posterior distribution of the state, which is reconstructed as a GMM from the previous time step. The proposed PGM-DS and PGM-DU filters are summarised in Algorithm 1.

4. Numerical simulation

The performance of the proposed PGM-DS and PGM-DU filters is examined and compared to other filters in the following three numerical simulation scenarios:

1. The first scenario is a 1-dimensional nonlinear dynamics, chosen to illustrate filter performance in a low-dimensional but highly nonlinear setting.
2. The second scenario is a 2-dimensional partially observable pendulum model, which captures both nonlinearity and ambiguity due to partial observability.

3. The last scenario is a 40-dimensional data assimilation problem of atmospheric dynamics, where the main challenges are high dimensionality and intermittent observations. The PF exhibited an impoverishment issue after the measurement update.

The proposed filters are compared to UKF, EnKF, PF, AKKF and PGM1 filters. The EnKF is a sample-based Bayes filter in which the samples (ensemble members) are shifted rather than weighted, incorporating with a Kalman update. It is more efficient than the standard PF, particularly in high-dimensional applications such as the tested Lorenz 96 model, if the filter is applicable in the given scenario. It is worth noting, however, that the performance of the EnKF can be limited in some nonlinear and non-Gaussian settings, as is discussed in [29]. The EnKF used in this study is implemented based on [24]. The AKKF is another sample-based Bayes filter that uses a kernel method, where particle updates are performed in kernel space via mapping functions. It demonstrates improved performance with a smaller number of particles compared to the standard PF, but the computational cost can increase exponentially in high-dimensional scenarios due to the kernel mapping for each particle. The implementation is based on [25], using a quadratic kernel, a kernel normalisation parameter $\lambda_{\tilde{K}} = 10^{-3}$ and a polynomial scale factor $\alpha_K = 10^3$. Each scenario was evaluated using 100 MC simulations, run in parallel. It is worth noting that none of the filters were parallelised in the simulation implementation, although parallelisation is feasible since it was not the focus of the study.

4.1. Performance evaluation metrics

For the performance evaluation, three metrics are considered: 1) the time-averaged root mean square error (ARMSE), 2) the percentage of cases exceeding the 99.99% χ^2 bound and 3) the averaged elapsed time. The MC-averaged RMSE over a total of N_{MC} runs at time step k can be calculated as follows:

$$\text{RMSE}_{MC}(k) = \sqrt{\frac{1}{N_{MC}} \sum_{mc=1}^{N_{MC}} (\mathbf{x}_k^{mc} - \hat{\mathbf{x}}_k^{mc})^2}, \quad (21)$$

where $\mathbf{x}_k^{mc}$ is the true state and $\hat{\mathbf{x}}_k^{mc}$ is the estimated state at time step k of the mc^{th} MC run. The ARMSE can be calculated as follows:

$$\text{ARMSE} = \frac{1}{K} \sum_{k=1}^K \text{RMSE}_{MC}(k). \quad (22)$$

A smaller ARMSE value indicates that the estimated state is closer to the true state. In other words, the filter provides more accurate state estimation across the MC runs and time steps. The percentage of cases exceeding the 99.99% χ^2 bound is calculated by counting the number of cases where the χ^2 value exceeds the 99.99% threshold at each time step and dividing it by the total number of MC runs. The χ^2 at time step k of the mc^{th} MC run can be calculated as follows:

$$\chi_{mc}^2(k) = (\mathbf{x}_k^{mc} - \hat{\mathbf{x}}_k^{mc})^T (\mathbf{P}_k^{mc})^{-1} (\mathbf{x}_k^{mc} - \hat{\mathbf{x}}_k^{mc}). \quad (23)$$

This chi-squared value indicates how well the model fits the data. In other words, a higher percentage of cases exceeding the threshold suggests that the filter's computed covariance does not accurately reflect the true estimation error [30]. The final metric, averaged elapsed time, is defined as the total computation time for all MC runs divided by the number of MC runs.

4.2. Scenario 1: 1-dimensional nonlinear dynamics

The first scenario is a 1-dimensional nonlinear dynamics [31], which is also considered in [16,19,20,25]. The system dynamics and measurement model for this problem are defined as follows:

$$x_k = 0.5x_{k-1} + \frac{25x_{k-1}}{1+x_{k-1}^2} + 8\cos(1.2(k-1)) + w_k, \quad (24)$$

Table 1

UT parameters for the 1-dimensional nonlinear dynamics.

Parameters	α	β	γ
Value	1.3	1.5	0.2

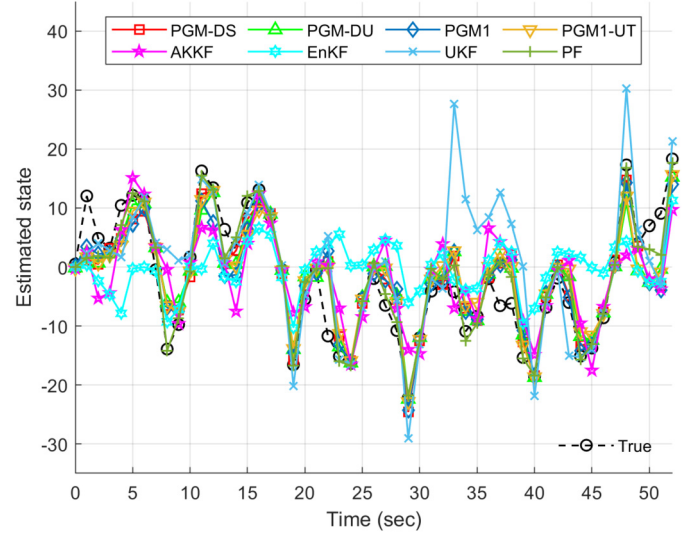


Fig. 1. One example result of the 1-dimensional nonlinear dynamics.

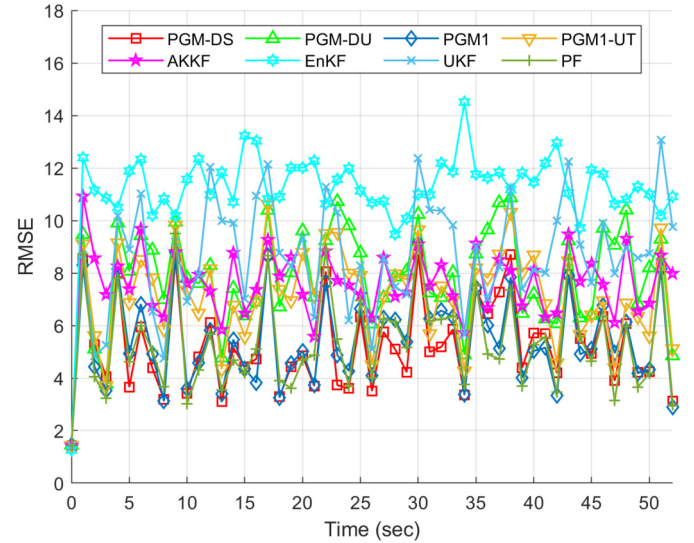


Fig. 2. RMSE of the 1-dimensional nonlinear dynamics.

$$z_k = \frac{x_k^2}{20} + v_k, \quad (25)$$

where w_k and v_k are zero-mean Gaussian process and measurement noises, respectively. The initial state and noise covariances are set as in previous studies [19], $x_0 = 0.0$, $P_0 = 2$, $Q = 10$ and $R = 1$ over 52 time steps with $\Delta t = 1$. It is worth noting that this problem is partially observable due to the presence of x_k^2 in the measurement model. A total of 50 particles is used for all particle-based filters, including PGM-DS and PGM-DU filters. The DBSCAN design parameters are set to $\epsilon = 10$ and $\min Pts = 2$, and the maximum number of clusters for the PGM1 and PGM1-UT filters is set to three. The parameters for the PGM1-DU filter, the PGM1-UT filter and the UKF are shown in Table 1.

The results from a total of 100 MC runs of the 1-dimensional nonlinear dynamics are shown in Figs. 1, 2 and 3 and Table 2.

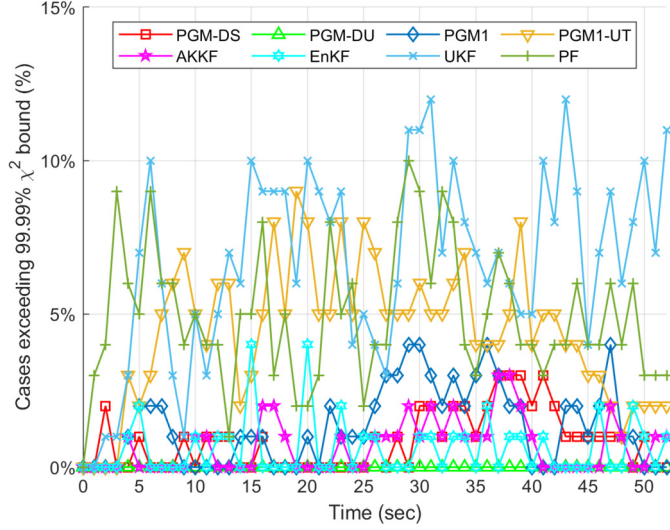


Fig. 3. The percentage of cases exceeding the 99.99% χ^2 bound of the 1-dimensional nonlinear dynamics.

Table 2
Results of the 1-dimensional nonlinear dynamics.

Filters	ARMSE	% of cases exceeding 99.99% χ^2 bound	Average elapsed time per run (s)
PGM-DS	5.3291	0.85%	0.1040
PGM-DU	7.8349	0.00%	0.0985
PGM1	5.4168	1.32%	66.0475
PGM1-UT	7.1555	4.40%	65.9947
AKKF	7.5849	0.70%	0.1011
EnKF	11.2110	0.62%	0.0591
UKF	8.6486	6.70%	0.0059
PF	5.3718	4.92%	0.0197

The PGM-DS filter demonstrated the best performance in terms of ARMSE, while the PGM-DU filter the best performance with the lowest percentage of cases exceeding the 99.99% χ^2 bound, compared to all other tested filters. Compared to the PGM1 and PGM1-UT filters, the PGM-DS and PGM-DU filters showed enhanced performance in either ARMSE or the percentage of cases exceeding the 99.99% χ^2 bound, while also reducing the average elapsed time by approximately 99.84%.

4.3. Scenario 2: simple pendulum system

The second scenario is a simple pendulum system, in which a rod rotates with a given initial angle and angular velocity. The system dynamics and measurement model for this system are defined as follows:

$$\theta_k = \theta_{k-1} + \Delta t \dot{\theta}_{k-1} + \Delta t^2 \ddot{\theta}_{k-1} + w_k, \quad (26)$$

$$z_k = L |\sin \theta_k| + v_k, \quad (27)$$

where θ_k , $\dot{\theta}_k$ and $\ddot{\theta}_k$ are the angle, angular velocity and angular acceleration between the rotating rod and the ground, respectively. Δt denotes the time step, L is the length of the rod, set to $L = 3$. The terms w_k and v_k are zero-mean Gaussian process and measurement noises with covariance Q and R , respectively. The initial state and noise covariances are set to $x_0 = [0.4\pi, 0.0]^T$, $P_0 = 0.4I_{2 \times 2}$, $Q = 10$ and $R = 0.02$ for 200 time steps with $\Delta t = 0.05$. The initial state estimate for the filters is set to $x_0 = [0.0, 0.0]^T$. It is worth noting that this problem is also partially observable due to the presence of $|\sin \theta_k|$ in the measurement model, where only the magnitude of the projected rod length is observed, and the sign is unknown. This leads to a bimodal posterior distribution, making the problem non-trivial, especially under the tested initial condition $\dot{\theta}_0 = 0$, in which two mathematically valid solutions may exist. Furthermore, it is assumed that the initial state estimate provided for the filters

Table 3
UT parameters for the simple pendulum system.

Parameters	α	β	κ	γ
Value	1.0	2.0	0.0	$\alpha^2(n + \kappa) - n$

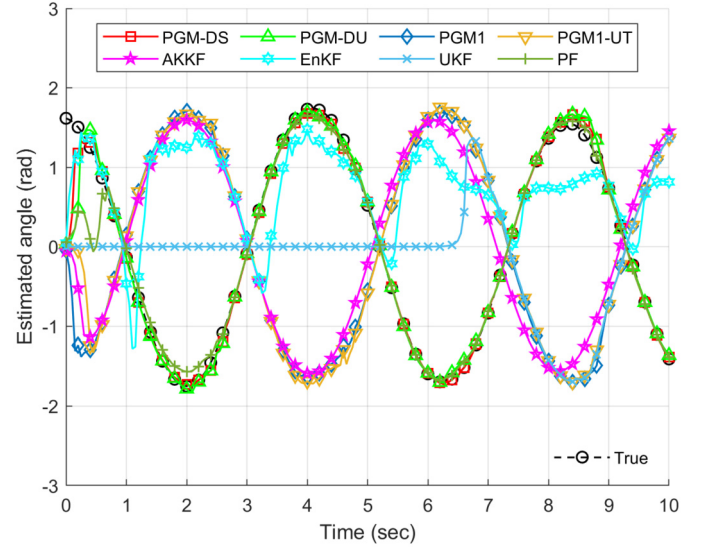


Fig. 4. One example result of the simple pendulum system.

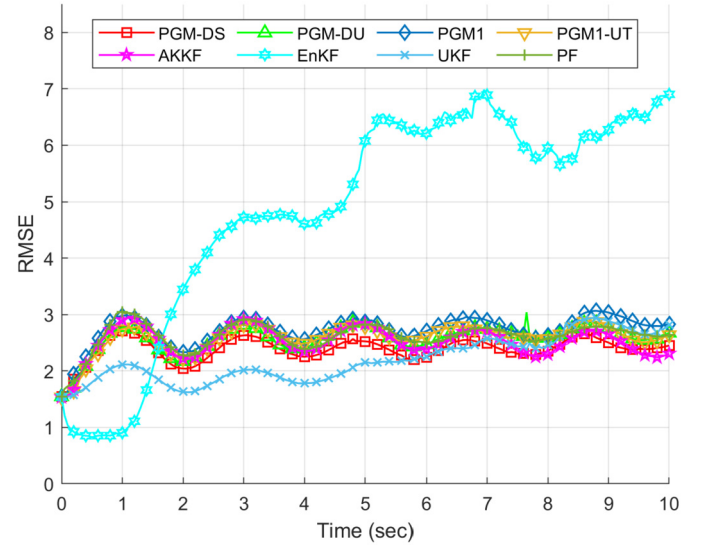


Fig. 5. RMSE of the simple pendulum system.

differs from the true initial condition of the system dynamics. A total of 100 particles is used for all particle-based filters, including the PGM-DS and PGM-DU filters. The DBSCAN design parameters are set to $\epsilon = 5$ and $minPts = 3$, and the maximum number of clusters for the PGM1 and PGM1-UT filters is set to two. The parameters for the PGM-DU filter, the PGM1-UT filter and the UKF are shown in Table 3.

The results from a total of 100 MC runs of the simple pendulum system are shown in Figs. 4, 5 and 6 and Table 4.

As aforementioned, this system has two mathematically valid solutions due to the partial observability in the measurement model. However, since ARMSE is computed using the mean of the estimated state, it may not align with either mode of the posterior distribution. This can result in misleading interpretations of estimation accuracy. For these reasons, ARMSE is not an appropriate performance metric for this scenario. Furthermore, since the system dynamics of the pendulum are

Table 4
Results of the simple pendulum system.

Filters	ARMSE	% of cases exceeding 99.99% χ^2 bound	Correct guess	Average elapsed time per run (s)
PGM-DS	2.3880	49.63%	49	1.2204
PGM-DU	2.5697	47.11%	53	1.0566
PGM1	2.7229	56.19%	45	139.8337
PGM1-UT	2.5836	46.54%	53	139.6743
AKKF	2.5224	64.17%	34	4.1178
EnKF	4.8542	44.27%	0	1.3329
UKF	2.1949	21.29%	0	0.0475
PF	2.6180	66.18%	52	0.2957

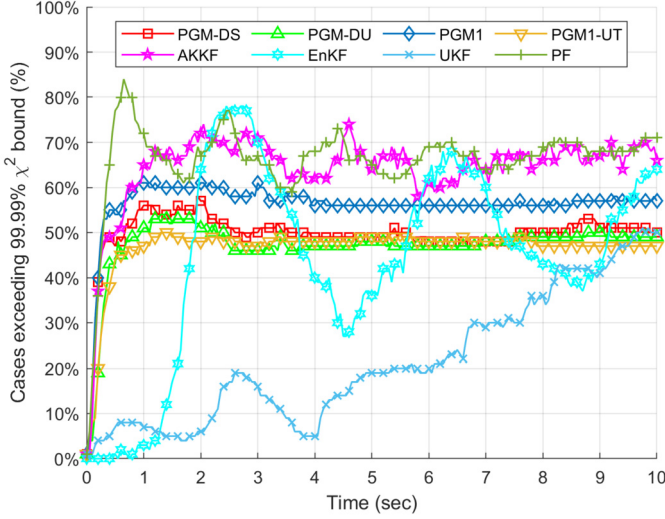


Fig. 6. The percentage of cases exceeding the 99.99% χ^2 bound of the simple pendulum system.

periodic, poor estimation or even complete estimation failure can still produce low ARMSE. For example, although the UKF failed to estimate the state, as shown in Fig. 4, it still produced a lower ARMSE compared to other filters that successfully estimated the state. In terms of the percentage of cases exceeding the 99.99% χ^2 bound, the PGM-DS and PGM-DU filters outperformed both the AKKF and PF. Also, PGM-DS and PGM-DU filters demonstrated unbiased estimation, as indicated by the “Correct guess” metric in Table 4, whereas the AKKF produced biased estimates and the UKF failed to estimate. Compared to the PGM1 and PGM1-UT filters, the PGM-DS and PGM-DU filters showed enhanced performance while also reducing the average elapsed time by approximately 99.13%.

4.4. Scenario 3: Lorenz 96 model

The final scenario is the Lorenz 96 model [32], which describes a simplified model of atmospheric dynamics. The Lorenz 96 model assumes N_o equally spaced observation points along the latitude of Earth and is formulated as a cyclical system. In other words, $x_0 = x_{N_o-1}$ and $x_{N_o} = x_1$. The system dynamics model for this problem is defined as follows:

$$\dot{x}_i = x_{i-1} (x_{i+1} - x_{i-2}) - x_i + F, \quad (28)$$

where x_i is the state at the i^{th} observation point, and F is a constant external forcing term. In order to test the Lorenz 96 model using discrete-time filters, the system is discretised using a fourth-order Runge-Kutta (RK4) ODE solver [33]. The discretised form of the Lorenz 96 model is given as follows:

$$\mathbf{x}_k = \mathbf{x}_{k-1} + \frac{\Delta t}{6} (k_1 + 2k_2 + 2k_3 + k_4) + \mathbf{w}_k, \quad (29)$$

Table 5
UT parameters for the Lorenz 96 model.

Parameters	α	β	κ	γ
Value	0.001	2.0	0.0	$\alpha^2 (n + \kappa) - n$

Table 6
Results of Lorenz 96 model.

Filters	ARMSE	% of cases exceeding 99.99% χ^2 bound	Average elapsed time per run (s)
PGM-DS	18.4441	45.02%	81.9433
PGM-DU	18.4680	41.51%	80.9594
PGM1	18.4313	44.90%	365.1138
PGM1-UT	18.4697	41.73%	355.8726
AKKF	18.7708	46.39%	1.4065e+03
EnKF	18.3014	50.69%	11.7241
PF	23.0626	41.96%	11.7051

where $\mathbf{x}_k = [x_k^1, x_k^2, \text{isthestateand} \dots, x_k^{N_o}]^T$ and $\mathbf{w}_k$ is the zero-mean Gaussian process noise with covariance $\mathbf{Q}$ at time step k . Also, Δt is the time step, and k_1 through k_4 are defined as follows:

$$k_1 = \mathbf{f}(\mathbf{x}_{k-1}), \quad (30)$$

$$k_2 = \mathbf{f}\left(\mathbf{x}_{k-1} + \frac{\Delta t}{2} k_1\right), \quad (31)$$

$$k_3 = \mathbf{f}\left(\mathbf{x}_{k-1} + \frac{\Delta t}{2} k_2\right), \quad (32)$$

$$k_4 = \mathbf{f}(\mathbf{x}_{k-1} + \Delta t k_3), \quad (33)$$

where $\mathbf{f}(\cdot)$ represents the system dynamics. The measurement model for the Lorenz 96 model is defined as follows:

$$\mathbf{z}_k = \mathbf{H} \mathbf{x}_k + \mathbf{v}_k, \quad \mathbf{H}_{i,j} = \begin{cases} 1, & \text{if } j = 2i - 1. \\ 0, & \text{otherwise,} \end{cases} \quad (34)$$

where $\mathbf{v}_k$ is the zero-mean Gaussian measurement noise with covariance $\mathbf{R}$ at time step k . In this simulation, 40 observation points are considered (i.e., $i = 1, 2, \dots, 40$), with a constant external forcing term set to $F = 8$. The initial state and noise covariances are as in previous studies [19], $\mathbf{x}_0 = F[1 \dots 1 \dots 1]^T$, $\mathbf{P}_0 = 10^{-3} \mathbf{I}_{40 \times 40}$, $\mathbf{Q} = 10^{-2}$ and $\mathbf{R} = 10^{-2}$ for 200 time steps with $\Delta t = 0.05$. In addition, the measurement update is performed every 20 time steps, corresponding to a 1 Hz update rate. It is worth noting that this problem is high-dimensional (40-dimensional) and partially observable, as the update is performed intermittently and measurement is available only from certain stations (if $j = 2i - 1$). A total of 2,000 particles are used for all particle-based filters, including the PGM-DS and PGM-DU filters. A higher number of particles is used in this scenario compared to the previous ones due to the increased state dimensionality. The DBSCAN design parameters are set to $\epsilon = 100$ and $\text{minPts} = 41$, and the maximum number of clusters for the PGM1 and PGM1-UT filters is set to two. The parameters for the PGM-DU filter, the PGM1-UT filter and the UKF are shown in Table 5.

The results from a total of 100 MC runs of the Lorenz 96 model are shown in Figs. 7, 8 and 9 and Table 6.

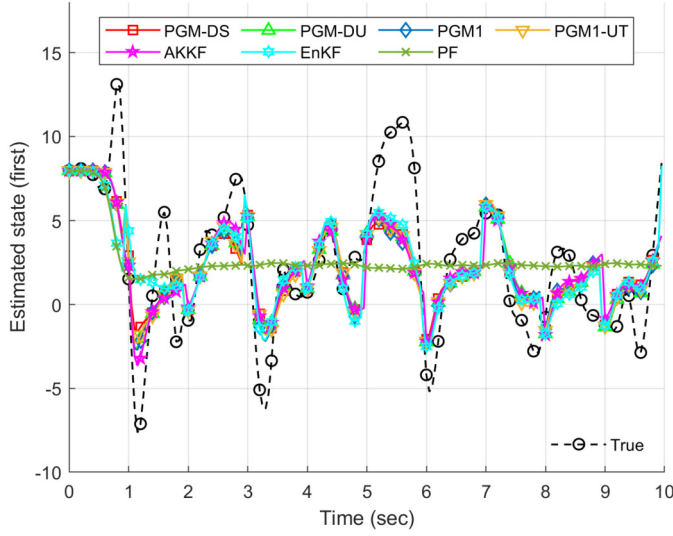


Fig. 7. One example result of Lorenz 96 model.

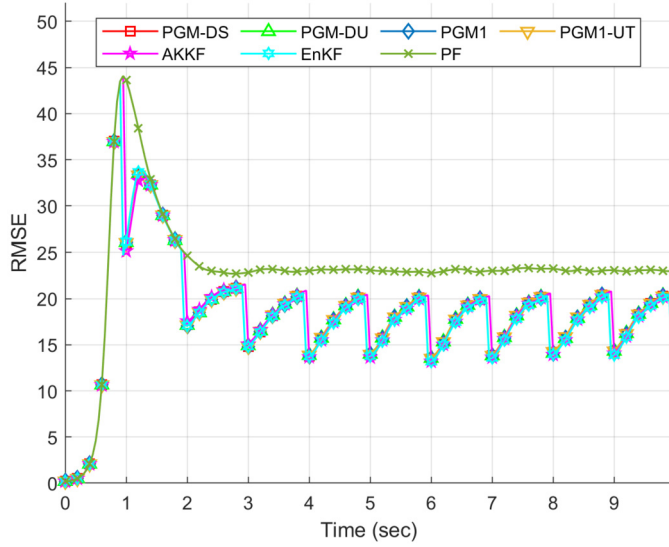
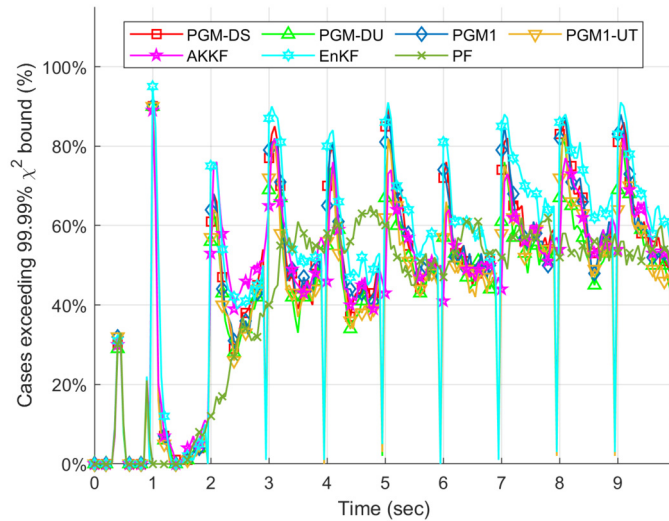
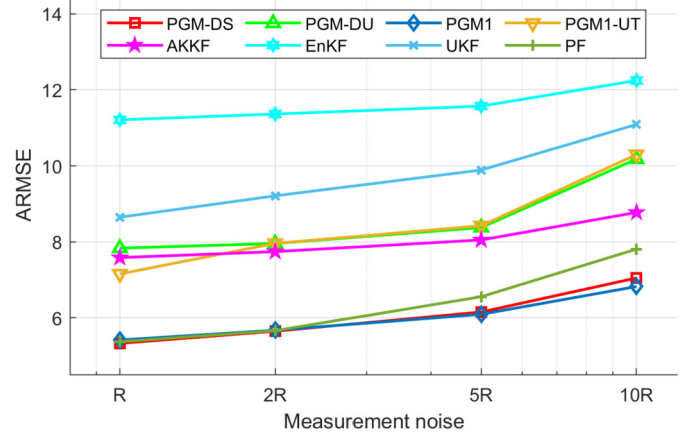


Fig. 8. RMSE of Lorenz 96 model.

Fig. 9. The percentage of cases exceeding the 99.99% χ^2 bound of Lorenz 96 model.Fig. 10. ARMSE of the tested filters under increasing measurement noise, R , $2R$, $5R$ and $10R$, in the 1-dimensional nonlinear dynamics.

The EnKF is one of the most widely used filtering techniques for high-dimensional data assimilation problems such as the Lorenz 96 model [24], and serves as a strong baseline for performance comparison. All tested filters, including the PGM-DS and PGM-DU filters, demonstrated performance comparable to the EnKF while outperforming it in terms of the percentage of cases exceeding the 99.99% χ^2 bound. It is also noteworthy that the PF failed to estimate the state after the first measurement update due to the particle impoverishment problem. Compared to the PGM1 and PGM1-UT filters, as well as the AKKF, the PGM-DS and PGM-DU filters achieved improved or comparable performance in ARMSE or the χ^2 metric while reducing the average elapsed time by approximately 77.40% compared to the PGM1 and PGM1-UT filters, and by 94.21% compared to the AKKF.

5. Robustness analysis

To further assess the robustness and properties of the proposed filters, a robustness analysis is conducted using the 1-D nonlinear dynamics. First, the noise robustness of the proposed filters is tested by increasing the measurement noise. Second, the sensitivity of the proposed filters to the DBSCAN design parameters is evaluated.

5.1. Noise robustness analysis

Measurement noise can significantly impact the performance of state estimation, and it is important to evaluate the robustness of the proposed filters to measurement noise. The proposed filters, along with other nonlinear filters, were tested using the 1-dimensional nonlinear dynamics under increased measurement noise, as shown in Fig. 10.

In this analysis, four measurement noise levels, R , $2R$, $5R$ and $10R$, were considered to test the filters under baseline noise and under moderate, high and extreme increases in measurement noise, respectively. The figure indicates that the proposed filters exhibit a small increase in ARMSE at the small increment level, $2R$, and a slightly higher increase in ARMSE for the $5R$ and $10R$ cases. In contrast, most of the other filters display a consistent increase in ARMSE as the measurement noise increases. In particular, the UKF and PF exhibit steeper increases in ARMSE at $5R$ and $10R$. These results demonstrate that the proposed filters are robust to moderate increases in measurement noise. However, the PGM-DU filter, which incorporates the UT, demonstrates a relatively rapid increase in ARMSE as the noise level increases.

5.2. Parameter sensitivity analysis

The selection of the two DBSCAN design parameters ϵ and $minPts$ can affect the performance of clustering which can lead to a difference

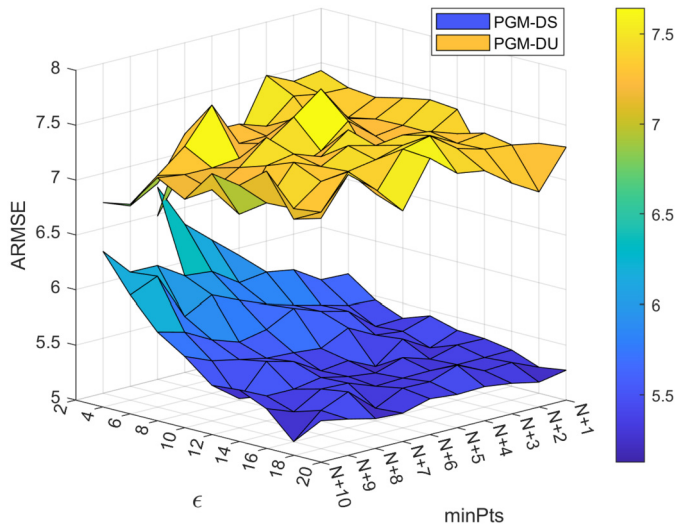


Fig. 11. Sensitivity of the PGM-DS and PGM-DU filters to the DBSCAN design parameters, ϵ and $\minPts$, in the 1-dimensional nonlinear dynamics. The surface plots show the ARMSE values for different parameter combinations (rotated for better visibility).

in the performance of the filtering with GMM construction. The grid search sensitivity analysis was conducted by varying the DBSCAN design parameters across predefined ranges to explore their joint impact on the filter performance, using the 1-dimensional nonlinear dynamics, as shown in Fig. 11.

In this analysis, ten ϵ values 2, 4, 6, ..., 20 and ten $\minPts$ values $N+1$, $N+2$, ..., $N+10$ were considered. The figure indicates that the ARMSE of the PGM-DS filter increases as $\minPts$ increases and ϵ decreases. In contrast, the PGM-DU filter is relatively robust to variations in both parameters, although its overall error remained larger than that of the PGM-DS filter. Furthermore, the proposed filters failed to cluster when all particles were considered as noise, leading to estimation failure when $\epsilon = 2$ and $\minPts \geq N+8$, as shown on the left side of Fig. 11.

6. Discussion

We proposed two new filters: the PGM-DS and PGM-DU filters. Throughout all three numerical scenarios, both filters outperformed the PGM1 and PGM1-UT filters, as well as other tested filters, while reducing the computational cost. The reduction in computation was possible due to the characteristic of DBSCAN, where the minimum number of particles to calculate the sample covariance of the Gaussian component was ensured with the DBSCAN design parameter $\minPts$, whilst the PGM1 and PGM1-UT filters with modified K-means need an optimisation step to ensure that. In addition to the comparison with other PGM filters, the proposed filters also showed improved performance compared to the PF, although they required higher computation. It should be emphasised, however, that the primary purpose of the computation comparison was to highlight that the computational load of the PGM-DS and PGM-DU filters is significantly lower than that of other PGM filters, a finding that holds true. Moreover, whilst some filters showed good performance in specific scenarios, the PGM-DS and PGM-DU filters consistently performed well across all tested cases. In particular, the AKKF exhibited degraded performance in the first scenario, the EnKF was unable to provide an accurate estimate in the first two scenarios, consistent with its known limitations in some nonlinear settings, and both the UKF and PF failed to estimate the state properly in the last scenario. One thing that should be noted here is that the PGM-DU filter showed a smaller percentage of cases exceeding the 99.99% compared to the PGM-DS filter, and the PGM-DS filter showed a smaller ARMSE compared to the PGM-DU filter. As we already mentioned in Section 4.1, this indicates that

the PGM-DS filter provides more accurate estimates of the state, but the filter's computed covariance does not indicate the true accuracy well. On the other hand, the PGM-DU filter suggests less accurate estimates of the state, but it tends to predict the better-matched uncertainty to the actual one.

From the robustness analysis, it is shown that the proposed filters are robust to moderate increases in measurement noise and to variations in the DBSCAN design parameters within certain regions in parameter space. However, the PGM-DU filter demonstrated a relatively rapid increase in ARMSE as the noise level became larger. Moreover, the proposed filters failed to cluster and estimate the state when $\epsilon = 2$ and $\minPts \geq N+8$, which represents a limitation of the filters. The rationale behind this failure is the noise-handling behaviour of DBSCAN, which labels samples as noise if they are not reachable from core points based on the parameters ϵ and $\minPts$. In the extreme case, all particles can be labelled as noise and, consequently, no samples are left to construct the GMM, causing the filter to fail in estimation. This can also limit performance if the filter parameters are not properly tuned, as the observation that more samples are considered as noise indicates fewer particles are available to approximate the density, leading to a poorer GMM approximation. Therefore, future research exploring other variants of DBSCAN may help mitigate this issue.

In addition to exploring alternative clustering strategies, future work will also consider other filters. One such method is the feedback particle filter (FPF), a popular variant of PF that provides a resampling-free solution to nonlinear problems and addresses the particle degeneracy issue [34]. The FPF has also shown promising results in high-dimensional settings [35], and a detailed comparison with the proposed filters is planned as part of future research.

7. Conclusion

In this study, DBSCAN-based PGM filters, the PGM-DS and PGM-DU filters, are proposed. These new filters based on DBSCAN demonstrated enhanced performance, which does not require a predefined number of Gaussian components and eliminates the need for optimisation to ensure the minimum number of particles in a cluster. Throughout three numerical simulation scenarios, the PGM-DS and PGM-DU filters demonstrated improved estimation performance compared to the PGM1 and PGM1-UT filters, with much less computational cost. Furthermore, it is shown that the proposed filters perform well across all three tested numerical simulations, whereas other nonlinear filters, such as UKF, EnKF or PF, failed to estimate the state in certain scenarios. The robustness analysis demonstrated that the proposed filters are robust to moderate increases in measurement noise compared to other tested filters, and to variations in the DBSCAN design parameters within certain regions of the parameter space, though performance may degrade when parameters are poorly tuned.

Several directions remain for future research. First, exploring alternative types of DBSCAN to mitigate the limitations caused by parameter selection is one direction of future work. Moreover, although the present study focused on comparisons with UKF, EnKF, PF, AKKF and PGM1 filters, a thorough comparison with the FPF is planned for future work. Such a comparison would provide a more comprehensive evaluation of the relative strengths and limitations of the proposed filters.

CRediT authorship contribution statement

Sukkeun Kim: Conceptualization, Data curation, Investigation, Methodology, Visualization, Writing – original draft. **Mengwei Sun:** Supervision, Writing – review & editing. **Ivan Petrunin:** Data curation, Supervision, Writing – review & editing. **Hyo-Sang Shin:** Conceptualization, Funding acquisition, Project administration, Supervision, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This research was supported by the Inha University funded project Precision Guidance and Navigation for Deep Space Exploration (project number P20962).

Appendix A. Supplementary material

Supplementary material related to this article can be found online at <https://doi.org/10.1016/j.dsp.2025.105546>.

Data availability

No data was used for the research described in the article.

References

- [1] N.J. Gordon, D.J. Salmond, A.F.M. Smith, Novel approach to nonlinear/non-Gaussian Bayesian state estimation, *IEE Proc., F, Radar Signal Process.* 140 (2) (1993) 107, <https://doi.org/10.1049/ip-f-2.1993.0015>.
- [2] G. Kitagawa, A Monte Carlo filtering and smoothing method for non-Gaussian nonlinear state space models, in: *Proceedings of the 2nd US-Japan Joint Seminar on Statistical Time Series Analysis 2*, 1993, pp. 110–131.
- [3] J. Candy, Bootstrap particle filtering, *IEEE Signal Process. Mag.* 24 (4) (2007) 73–85, <https://doi.org/10.1109/msp.2007.4286566>.
- [4] T. Li, S. Sun, T.P. Sattar, J.M. Corchado, Fight sample degeneracy and impoverishment in particle filters: a review of intelligent approaches, *Expert Syst. Appl.* 41 (8) (2014) 3944–3954, <https://doi.org/10.1016/j.eswa.2013.12.031>, arXiv:1308.2443.
- [5] R. Douc, O. Cappé, E. Moulines, Comparison of resampling schemes for particle filtering, in: *ISPA 2005, Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis*, 2005, pp. 64–69.
- [6] T. Li, M. Bolic, P.M. Djuric, Resampling methods for particle filtering: classification, implementation, and strategies, *IEEE Signal Process. Mag.* 32 (3) (2015) 70–86, <https://doi.org/10.1109/msp.2014.2330626>.
- [7] C. Kuptamete, N. Aunsri, A review of resampling techniques in particle filtering framework, *Measurement* 193 (2022) 110836, <https://doi.org/10.1016/j.measurement.2022.110836>.
- [8] S. Kim, I. Petrunin, H.-S. Shin, A review of Bayes filters with machine learning techniques and their applications, *Inf. Fusion* 114 (2025) 102707, <https://doi.org/10.1016/j.inffus.2024.102707>.
- [9] S.L. Kham, K. Chamnongthai, N. Aunsri, A review of efficient techniques and applications of kernel density in particle filtering framework, *Inf. Fusion* 122 (2025) 103169, <https://doi.org/10.1016/j.inffus.2025.103169>.
- [10] T. Imbiriba, P. Closas, Enhancing particle filtering using Gaussian processes, in: *2020 IEEE 23rd International Conference on Information Fusion (FUSION)*, IEEE, 2020, pp. 1–7.
- [11] K. Comandur, Y. Li, S. Nannuru, Particle flow Gaussian particle filter, in: *2022 25th International Conference on Information Fusion (FUSION)*, IEEE, 2022, pp. 1–6.
- [12] C. Kuptamete, Z.-H. Michalopoulou, N. Aunsri, Adaptive genetic algorithm-based particle herding scheme for mitigating particle impoverishment, *Measurement* 214 (2023) 112785.
- [13] S. Kim, H.J. Park, J.-H. Choi, D. Kwon, A novel prognostics approach using shifting kernel particle filter of li-ion batteries under state changes, *IEEE Trans. Ind. Electron.* 68 (4) (2021) 3485–3493, <https://doi.org/10.1109/TIE.2020.2978688>.
- [14] D. Aspath, H. Sorenson, Nonlinear Bayesian estimation using Gaussian sum approximations, *IEEE Trans. Autom. Control* 17 (4) (1972) 439–448, <https://doi.org/10.1109/tac.1972.1100034>.
- [15] J.H. Kotecha, P.M. Djurić, Gaussian sum particle filtering, *IEEE Trans. Signal Process.* 51 (10) (2003) 2602, <https://doi.org/10.1109/tsp.2003.816754>.
- [16] M.L. Psiaki, Gaussian mixture nonlinear filtering with resampling for mixand narrowing, *IEEE Trans. Signal Process.* 64 (21) (2015) 5499–5512, <https://doi.org/10.1109/tsp.2016.2595503>.
- [17] M.L. Psiaki, J.R. Schoenberg, I.T. Miller, Gaussian sum reapproximation for use in a nonlinear filter, *J. Guid. Control Dyn.* 38 (2) (2015) 292–303, <https://doi.org/10.2514/1.g000541>.
- [18] S. Yun, R. Zanetti, Clustering methods for particle filters with Gaussian mixture models, *IEEE Trans. Aerosp. Electron. Syst.* 58 (2) (2022) 1109–1118, <https://doi.org/10.1109/taes.2021.3117655>.
- [19] D. Raihan, S. Chakravorty, Particle Gaussian mixture filters-I, *Automatica* 98 (2018) 331–340, <https://doi.org/10.1016/j.automatica.2018.07.023>.
- [20] D. Raihan, S. Chakravorty, Particle Gaussian mixture filters-II, *Automatica* 98 (2018) 341–349, <https://doi.org/10.1016/j.automatica.2018.07.024>.
- [21] D. Raihan, S. Chakravorty, Particle Gaussian mixture filters: application and performance evaluation, in: *2019 22th International Conference on Information Fusion (FUSION)*, 2019, pp. 1–8.
- [22] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, A density-based algorithm for discovering clusters in large spatial databases with noise, in: *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, KDD'96*, AAAI Press, 1996, pp. 226–231.
- [23] N. Murugesan, I. Cho, C. Tortora, *Data Analysis and Rationality in a Complex World, Studies in Classification, Data Analysis, and Knowledge Organization*, 2021, pp. 175–185.
- [24] G. Evensen, Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics, *J. Geophys. Res., Oceans* 99 (C5) (1994) 10143–10162, <https://doi.org/10.1029/94JC00572>, <https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/94JC00572>.
- [25] M. Sun, M.E. Davies, I.K. Proudler, J.R. Hopgood, Adaptive kernel Kalman filter, *IEEE Trans. Signal Process.* 71 (2023) 713–726, <https://doi.org/10.1109/tsp.2023.3250829>, arXiv:2203.08300.
- [26] R.E. Kalman, A new approach to linear filtering and prediction problems, *J. Basic Eng.* 82 (1) (1960) 35–45, <https://doi.org/10.1115/1.3662552>.
- [27] R.A. Redner, H.F. Walker, Mixture densities, maximum likelihood and the EM algorithm, *SIAM Rev.* 26 (2) (1984) 195–239, <https://doi.org/10.1137/1026034>.
- [28] S. Julier, J. Uhlmann, Unscented filtering and nonlinear estimation, *Proc. IEEE* 92 (3) (2004) 401–422, <https://doi.org/10.1109/JPROC.2003.823141>.
- [29] I. Grooms, A comparison of nonlinear extensions to the ensemble Kalman filter: Gaussian anamorphosis and two-step ensemble filters, *Comput. Geosci.* 26 (3) (2022) 633–650.
- [30] M. Psiaki, The blind tricyclist problem and a comparative study of nonlinear filters: a challenging benchmark for evaluating nonlinear estimation methods, *IEEE Control Syst.* 33 (3) (2013) 40–54, <https://doi.org/10.1109/mcs.2013.2249422>.
- [31] N.J. Gordon, D.J. Salmond, A.F. Smith, *Novel Approach to Nonlinear/Non-gaussian Bayesian State Estimation*, IEE Proceedings F (Radar and Signal Processing), vol. 140, IET, 1993, pp. 107–113.
- [32] E. Lorenz, *Predictability: a problem partly solved*, Ph.D. thesis, Shinfield Park, Reading, 1995.
- [33] E. Süli, D.F. Mayers, *An Introduction to Numerical Analysis*, Cambridge University Press, 2003.
- [34] T. Yang, P.G. Mehta, S.P. Meyn, Feedback particle filter, *IEEE Trans. Autom. Control* 58 (10) (2013) 2465–2480, <https://doi.org/10.1109/TAC.2013.2258825>.
- [35] S.C. Surace, A. Kutschireiter, J.-P. Pfister, How to avoid the curse of dimensionality: scalability of particle filters with and without importance weights, *SIAM Rev.* 61 (1) (2019) 79–91.



DSPACE

<https://dspace.org/>

DBSCAN-based particle Gaussian mixture filters

Kim, Sukkeun; Sun, Mengwei; Petrunin, Ivan; Shin, Hyo-Sang

2026-01-01

Elsevier

<https://doi.org/10.1016/j.dsp.2025.105546>

Kim S, Sun M, Petrunin I, Shin H-S. (2026) DBSCAN-based particle Gaussian mixture filters. Digital Signal Processing, Volume 168, January 2026, Article number 105546