Automatica 98 (2018) 341–349

Contents lists available at ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Brief paper

Particle Gaussian mixture filters-II ☆

Dilshad Raihan, Suman Chakravorty *

Department of Aerospace Engineering, Texas A & M University, College Station, TX, 77840, United States

ARTICLE INFO

Article history:
Received 18 January 2017
Received in revised form 23 March 2018
Accepted 4 July 2018
Available online 13 September 2018

Keywords:
Estimation algorithms
Nonlinear filters
Machine learning
Gaussian mixture models
Multimodality
Markov chain Monte Carlo
State estimation
Curse of dimensionality
Parallel algorithms

ABSTRACT

In our previous work, we proposed a particle Gaussian mixture (PGM-I) filter for nonlinear estimation. The PGM-I filter uses the transition kernel of the state Markov chain to sample from the propagated prior. It constructs a Gaussian mixture representation of the propagated prior density by clustering the samples. The measurement data are incorporated by updating individual mixture modes using the Kalman measurement update. However, the Kalman measurement update is inexact when the measurement function is nonlinear and leads to the restrictive assumption that the number of modes remains fixed during the measurement update. In this paper, we introduce an alternate PGM-II filter that employs parallelized Markov Chain Monte Carlo (MCMC) sampling to perform the measurement update. The PGM-II filter update is asymptotically exact and does not enforce any assumptions on the number of Gaussian modes. The PGM-II filter is employed in the estimation of two test case systems. The results indicate that the PGM-II filter is suitable for handling nonlinear/non-Gaussian measurement update.

1. Introduction

Nonlinear filtering is the problem of estimating the state of a stochastic nonlinear dynamical system using noisy observations. The filtered state probability density function (PDF) may assume non-Gaussian and multimodal densities in nonlinear settings. Multimodality of the state PDF can be incorporated in the estimator by employing a Gaussian mixture model (GMM) representation (Sorenson & Alspach, 1971). However, Gaussian mixture filters such as the Gaussian sum EKF/UKF tend to keep the number of mixture components fixed throughout the estimation process (Alspach & Sorenson, 1972). Adaptive entropy based Gaussian-mixture information synthesis (AEGIS) is a Gaussian mixture filtering approach capable of splitting mixture modes based on entropy considerations (DeMars, Bishop, & Jah, 2013). A Gaussian mixture 'blob' filter that limits the size of the mixture covariances using linear matrix inequality (LMI) bounds so as to limit the effects of nonlinearity has been proposed recently (Psiaki, 2016). The 'blob' filter is capable of adapting the number of mixture components used in estimation. The Particle Filters (PF) are a class of sequential Monte Carlo methods that employ a Dirac delta representation of the state PDF (Gordon, Salmond, & Smith, 1993). The PF is capable of handling non-Gaussianity and relies on importance sampling to generate a weighted set of samples, also known as particles, from the posterior PDF. However, it is not computationally feasible to use PF in the estimation of large dimensional systems due to the well known weight degeneration problem (Bengtsson, Bickel, & Li, 2008).

In our companion work (Raihan & Chakravorty, 2018), we proposed a particle Gaussian mixture filter (PGM-I) to address the general nonlinear non-Gaussian filtering problem. The PGM-I filter was designed to be a Gaussian mixture filter that is capable of handling the nonlinear uncertainty propagation without enforcing restrictive assumptions. It allows the number of mixture modes and the mixture weights to be adjusted during propagation. However, the Kalman measurement update employed in the PGM-I filter is inaccurate when the measurement functions are highly nonlinear. Additionally, under a Kalman update, the number of Gaussian components remains fixed during the measurement update. In this paper, we propose a PGM-II filter with an alternate measurement update scheme that is asymptotically exact even when the measurement models are nonlinear. The PGM-II filter does away with the Kalman measurement update used in PGM-I Filter. Instead, it uses a parallelized Markov chain Monte Carlo (MCMC) method to sample from the posterior PDF. The remainder of this article is organized as follows: Mathematical preliminaries

DOI of original article: http://dx.doi.org/10.1016/j.automatica.2018.07.023.

☆ This work was supported by the AFOSR grants FA9550-13-1-0074, FA9550-17-1-0068 under the Dynamic Data Driven Application Systems (DDDAS) program and the NSF award ECCS1637889. The material in this paper was partially presented at the 2018 International Conference on Information Fusion (FUSION 2018), July 9–13, 2018, Cambridge, UK. This paper was recommended for publication in revised form by Associate Editor Gianluigi Pillonetto under the direction of Editor Torsten Söderström.

* Corresponding author.
E-mail addresses: dilshadraihan@tamu.edu (D. Raihan), schakrav@tamu.edu (S. Chakravorty).

https://doi.org/10.1016/j.automatica.2018.07.024
0005-1098/

of Gaussian mixture filtering are introduced in Section 2. A brief overview of MCMC methods is presented in Section 3. The PGM-II filter algorithm is presented in Section 4. The PGM-II filter is employed in the estimation of two test problems in Section 5.

An extensive survey of the literature can be found in Veettil and Chakravorty (2016).

2. Gaussian mixture filtering

Let $x \in \Re^d$ be the state of a dynamical system given by

$$x_{n+1} = f(x_n, w_n), \tag{1}$$

where w_t is a noise term with known distribution. Let $z_1, z_2, \ldots, z_n$ be a sequence of measurements of the system where

$$z_n = h(x_n) + \nu_n. \tag{2}$$

The distribution of the measurement noise term ν_n is assumed to be known. Given this state space description and the initial state PDF $\pi_0(x)$, the objective of the filtering problem is to be able to determine the conditional state PDF $p_n(x|Z_n)$. Here Z_n represents the sequence of all measurements recorded until time n. The transition kernel $p_n(x|x')$ of the state Markov chain can be derived from the process model given in (1). Given the transition kernel $p_n(x|x')$ and the measurement likelihood $p_n(z_n|x)$, the filtered density of the state Markov chain can be computed using a recursive algorithm that involves two basic steps. Let π_{n-1} be the PDF of the state at time $n-1$ conditioned on Z_{n-1}. Given π_{n-1}, the prediction step evaluates the propagated prior $\pi_n^-(x)$, i.e., the PDF of the state at n conditioned on Z_{n-1}, using the law of total probability.

$$\pi_n^-(x) = \int p_n(x|x')\pi_{n-1}(x')dx', \tag{3}$$

In the measurement update step, the propagated PDF $\pi_n^-(x)$ is updated with the new measurement z_n according to the Bayes rule to obtain the posterior PDF $\pi_n(x)$.

$$\pi_n(x) = \frac{p_n(z_n|x)\pi_n^-(x)}{\int p_n(z_n|x')\pi_n^-(x')dx'}, \tag{4}$$

Let us assume that the prior PDF $\pi_{n-1}(X)$ and the propagated prior $\pi_n^-(x)$ can be approximated by a weighted sum of Gaussian PDFs.

$\pi_{n-1}(x) = \sum_{i=1}^{M(n-1)}\omega_i(n-1)\mathcal{G}_i(x;\mu_i(n-1), P_i(n-1))$, $\pi_n^-(x) = \sum_{i=1}^{M^-(n)}\omega_i^-(n)\mathcal{G}_i^-(x;\mu_i^-(n), P_i^-(n))$. When the GMM representations of $\pi_{n-1}(X)$ and $\pi_n^-(x)$ are substituted in (3) and (4), we get

$$\pi_n^-(x) = \sum_{i=1}^{M(n-1)} \omega_i(n-1) \underbrace{\int p_n(x|x')\mathcal{G}_i(x';\mu_i(n-1), P_i(n-1))dx'}_{\pi_{i,n}^-(x)}. \tag{5}$$

From (5), it can be seen that propagated prior can be represented as mixture model $\{(\omega_i^-(n), \pi_{i,n}^-(x))\}$, $i \in \{1, \ldots, M(n-1)\}$ where

$$\omega_i^-(n) = \omega_i(n-1), \tag{6}$$

$$\pi_{i,n}^-(x) = \int p_n(x|x')\mathcal{G}_i^-(x';\mu_i^-(n), P_i^-(n))dx'. \tag{7}$$

This mixture model has $M(n-1)$ components like the GMM of the prior PDF in (5). However, the components $\pi_{i,n}^-(x)$ are not guaranteed to be Gaussian PDFs and as demonstrated in our previous work (Raihan & Chakravorty, 2016) Gaussian mixture modes undergoing a nonlinear transformation could split to form new modes or coalesce with other modes. Hence, in general $M^-(n) \neq M(n-1)$. The PGM-I filter was proposed to incorporate this feature in Gaussian mixture filters.

Let $l_i(n)$ be the likelihood that the measurement z_n came from the ith mixture component.

$$l_i(n) \equiv \int p_n(z_n|x')\pi_{i,n}^-(x')dx'. \tag{8}$$

The expression for the posterior PDF can be rewritten as follows (Raihan & Chakravorty, 2016).

$$\pi_n(x) = \sum_{i=1}^{M^-(n)} \underbrace{\frac{w_i^-(n)l_i(n)}{\sum_j w_j^-(n)l_j(n)}}_{w_i(n)} \underbrace{\frac{p_n(z_n|x)\pi_{i,n}^-(x)}{l_i(n)}}_{\pi_{i,n}(x)}. \tag{9}$$

This shows that the posterior PDF $\pi_n(x)$ can be represented as a mixture model $\{(\omega_i(n), \pi_{i,n}(x))\}$, $i \in \{1, \ldots, M^-(n)\}$ where,

$$\omega_i(n) = \frac{w_i^-(n)l_i(n)}{\sum_j w_j^-(n)l_j(n)}, \tag{10}$$

$$\pi_{i,n}(x) = \frac{p_n(z_n|x)\pi_{i,n}^-(x)}{l_i(n)}. \tag{11}$$

This mixture model has $M^-(n)$ components and the mixands $\pi_{i,n}(x)$ are not guaranteed to be Gaussian when the measurement function is nonlinear. The PGM-I filter performs a Kalman measurement update on each component of the GMM representing the predicted prior to obtain an $M^-(n)$ component GMM representation of the posterior PDF. However, when the measurement function is highly nonlinear or ambiguous, this is not satisfactory. For example, let the propagated PDF at time n be given by a unimodal Gaussian PDF

$$\pi_n^-(x) = \mathcal{G}(X, \begin{bmatrix}0\\0\end{bmatrix}, \begin{bmatrix}1 & 0\\0 & 2\end{bmatrix}) \tag{12}$$

Assume that a measurement $z_n = 2$ is recorded where $z = x_1^2 + \tau$, such that $\tau \sim \mathcal{G}(x, 0, 2)$. Ensemble representations of $\pi_n^-(x)$ and $\pi_n(x)$ given in Fig. 1 show that the posterior PDF cannot be adequately represented with a unimodal Gaussian PDF.

3. Markov Chain Monte Carlo

The MCMC methods are a class of algorithms that are used to generate samples from probability distributions that are not amenable to direct sampling (Gilks, Richardson, & Spiegelhalter, 1996). In the present paper we consider the Metropolis Hastings (M-H) algorithm which relies on a proposal distribution to generate the samples (Hastings, 1970). Formally, let $p(x)$ be the target distribution from which the samples are to be generated. Let $Q(x^i|x^{i-1})$ be the proposal distribution. Then the MCMC algorithm proceeds as follows. Let x^{t-1} be the sampled state at $t-1$. Then generate $x^{t*} \sim Q(x|x^{t-1})$. The candidate state x^{t*} is then chosen or not based on the acceptance probability α. The acceptance probability is computed as $\alpha = min\{1, \frac{Q(x^{t-1}|x^{t*})p(x^{t*})}{Q(x^{t*}|x^{t-1})p(x^{t-1})}\}$. It can be shown that the sampling rule given above is constructed so that the target distribution $p(x)$ is the equilibrium distribution of the resulting Markov chain. This implies that the initial samples may not be distributed according to $p(x)$. As a result, all points sampled during an initial burn-in period T_{br} are discarded. In theory, the M-H algorithm is capable of generating samples from complex multi-modal distributions. However, generating a representative sample from a multimodal distribution may require a long burn-in period and a large sample size. Parallelizable MCMC algorithms that split the state space into partitions and allow asynchronous sampling from individual partition elements have been proposed recently (VanDerwerken & Schmidler, 2013). In this work, we propose a similar approach to sample from multimodal posterior PDFs.

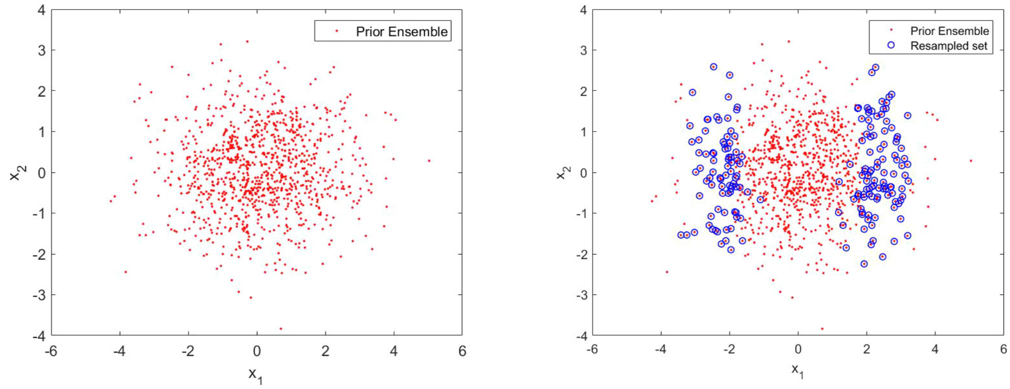


Fig. 1. Formation of multimodality during measurement update.

4. PGM-II filter

In this section we present a step by step description of the proposed PGM-II filter and an associated convergence result.

4.1. The PGM-II algorithm

The PGM-II filter relies on a Gaussian mixture representation of the state PDF. However, unlike the PGM-I filter, it is not essential for the operation of PGM-II algorithm that we obtain a functional representation of the posterior PDF.

Assumption 1. The predicted prior PDF and the filtered PDF can be accurately approximated by a GMM.

Given an ensemble of states $S_{n-1} = \{x^1_{n-1} \cdots x^N_{n-1}\}$ from the prior PDF at time $n-1$, the PGM-II Filtering algorithm is composed of the three basic steps described below.

(1) **Prediction**: During the prediction step, the PGM-II filter generates an ensemble S^-_n from the predicted prior $\pi^-_n(x)$ using samples drawn from the prior PDF $\pi_{n-1}(x)$, i.e., the ensemble S_{n-1}, and the Markov transition kernel $p_n(x'|x)$. A pictorial representation of the prediction is given as the first step in Fig. 2.

(2) **Clustering** : A functional representation of $\pi^-_n(x)$ in the form of a GMM is recovered from the ensemble S^-_n using a clustering scheme $\mathcal{C}$ (Richard, Duda, Hart, & Stork, 2000). The output of the clustering scheme is composed of the mixture weights $\omega^-_i(n)$, means $\mu^-_i(n)$ and covariances $P^-_i(n)$. The ellipsoids obtained at the end of clustering step in Fig. 2 represent the Gaussian mixture components. In particular, $\pi^-_n(x) = \sum_{i=1}^{M^-(n)} \omega^-_i(n)\mathcal{G}_i(x; \mu^-_i(n), P^-_i(n))$.

(3) **Measurement update**: The PGM-II filter relies on a parallelized MCMC method to perform the measurement update. The parallelized MCMC update is broken down into the following four steps.

   (a) Sample from the ith posterior mixture component $\pi_{i,n}(x)$ from (11) using MCMC to obtain the ith posterior component ensemble A_i.
   (b) Cluster the ith posterior component sample A_i to obtain a functional representation for the component pdf $\pi_{i,n}(x)$.
   (c) Evaluate the ith posterior mixture component weight $w_i(n)$ from (10).
   (d) Sample from the mixture model $\{w_i, \pi_{i,n}(x)\}$ to obtain a full posterior ensemble S_n.

The four step update process is described in more detail below.

Let $p_n(z_n|x)$ be the measurement likelihood. Then the posterior distribution is proportional to the product of the predicted prior and the likelihood $p_n(z_n|x)$, i.e., $\pi_n(x) \propto p_n(z_n|x)\pi^-_n(x)$. We rewrite the posterior PDF in its mixture form as obtained in (9):

$$\pi_n(x) = \sum_{i=1}^{M^-(n)} w_i(n)\pi_{i,n}(x). \qquad (13)$$

Furthermore, from (7), $\pi_{i,n}(x) \propto p_n(z_n|x)\pi^-_{i,n}(x)$. In step 3a of the measurement update, the PGM-II filter generates ensembles A_i from the mixture components $\pi_{i,n}(x), i \in \{1, 2, \ldots, M^-(n)\}$ using MCMC since $p_n(z_n|x)$ is given and $\pi^-_{i,n}(x)$ is known from the clustering step. From a computational standpoint, it is much more appealing to perform MCMC sampling on the individual mixture components $\pi_{i,n}(x)$ as opposed to the full posterior PDF $\pi_n(x)$. This completes step 3a. Due to the random walk behavior of MCMC, consecutive samples from A_i will be correlated. To remove correlations, we propose clustering the samples and obtaining a functional representation for the underlying component pdf $\pi_{i,n}(x)$. Notice that the mixture representation of $\pi_{i,n}(x)$ will be parameterized by expectations of various functions of the component random variable. The ergodicity of the chain will ensure that sample averages computed from MCMC samples during clustering will converge to these expectations, in spite of the correlations. Once a mixture representation for $\pi_{i,n}(x)$ is constructed, we can obtain independent samples from it by direct sampling. The clustering of A_i to obtain functional representation of $\pi_{i,n}(x)$ completes the step 3b of measurement update. Notice that in (13), each component PDF $\pi_{i,n}(x)$ has a mixing probability $w_i(n)$ associated with it. Step 3c consists of obtaining these weights. However, to compute the mixture weights, we need to evaluate the modal likelihoods $l_i(n)$, given by the integral in (8). Evaluating this integral is non trivial when the measurement function is nonlinear. So an approximation is used in the computation of $w_i(n)$. The calculation of approximate modal likelihoods is discussed in detail in Section 4.2. From the component pdfs $\pi_{i,n}(x)$ in step 3b and the weights $w_i(n)$ in step 3c, we can obtain a mixture representation of the posterior pdf as given in (13). Given the mixture representation, a sample X from the full posterior PDF $\pi_n(x)$ can be obtained via the two step approach given below.

(i) Choose a component by sampling k from $\{1, 2, \ldots, M^-(n)\}$ with probability $w_k(n)$.

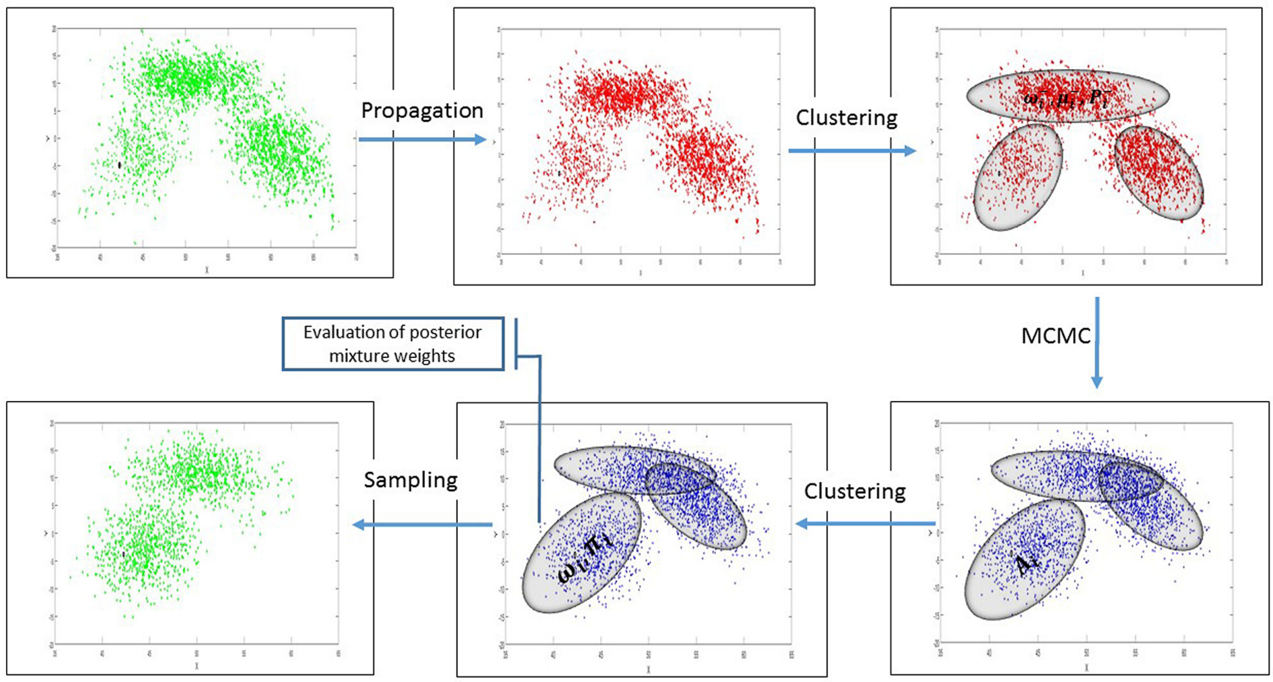


Fig. 2. PGM-II filter—prediction and update.

(ii) Draw a sample X from the component PDF $\pi_{k,n}(x)$.

This is the sampling process given in step 3d which completes the measurement update step. Algorithm 1 gives a pseudo code description of the PGM-II filter.

4.2. Calculation of likelihoods

As mentioned previously, the PGM-II filter generates an ensemble A_i from the posterior mixture component $\pi_{i,n}(x)$ using MCMC sampling. Let $\eta(x)$ be a proper PDF.

Algorithm 1 PGM-II Algorithm

Given $\pi_0(x) = \sum_{i=1}^{M(0)} \omega_i(0)\mathcal{G}_i(x; \mu_i(0), P_i(0))$, transition density kernel $p_n(x'|x)$, n = 1.

(1) Sample N particles $X^{(i)}$ from π_{n-1} and the transition kernel $p_n(x'|x)$ as follows:

  (a) Sample $X^{(i)}$ from $\pi_{n-1}(.)$.

  (b) Sample $X'^{(i)}$ from $p_n(.|X^{(i)})$.

(2) Use a Clustering Algorithm $\mathcal{C}$ to cluster the set of particles $\{X'^{(i)}\}$ into $M^-(n)$ Gaussian clusters with weights, mean and covariance given by $\{w_i^-(n), \mu_i^-(n), P_i^-(n)\}$.

(3) Use MCMC to sample from the component posteriors $\pi_{i,n}(x)$ to generate the ensembles A_i

(4) Compute the mixture weights $w_i(n)$ by evaluating the sequence of modal likelihoods $l_i(n)$ using (8) and (10)

(5) Sample N particles from the weighted collection of ensembles $\{(w_i(n), A_{n,i})\}$

(6) n = n+1, go to Step 1.

Then, from (11), we have $\int_{\mathcal{R}^n} \frac{\eta(x)\pi_{i,n}(x)}{p_n(z_n|x)\mathcal{G}_i^-(x;\mu_i^-(n),P_i^-(n))} dx = \frac{1}{l_i(n)}$. Since A_i are samples from $\pi_{i,n}(x)$, an importance sampling approximation to the above integral can be arrived as follows:

$$\frac{1}{l_i(n)} \approx \sum_{j=1}^{N_{i,n}} \frac{\eta(x_j)}{p_n(z_n|x_j)\mathcal{G}_i^-(x_j; \mu_i^-(n), P_i^-(n))}. \tag{14}$$

Hence an estimate of $l_i(n)$ can be computed by evaluating the sum given in (14) using the MCMC samples and taking the reciprocal (Gelfand & Dey, 1994).

To sample from the posterior PDF $\pi_n(x)$ using MCMC, we need a function that is at least proportional to it. Given a GMM representing the predicted prior PDF, we have $\pi_{i,n}(x) \propto \mathcal{G}_i^-(x; \mu_i^-(n), P_i^-(n))$ $p_n(z_n|x)$. The PGM-II filter obtains a GMM representation of the predicted prior by clustering the predicted ensemble. In the present work, we have used an approach that relies on k-means clustering algorithm to obtain the GMM parameters. The simple k-means clustering algorithm requires the number of mixture components to be input externally. To overcome this limitation, we have developed a clustering scheme which determines the optimal number of clusters given an upper bound on this number (Veettil & Chakravorty, 2016). In present work we have used Gaussian proposals of the form $X^t \sim X^{t-1} + \mathcal{G}(0, K_p\Sigma)$, where K_p is a positive constant. The covariance Σ can be chosen as the component covariance of the predicted prior $P_i^-(n)$.

4.3. Analysis of the PGM-II algorithm

In the following, we prove that the PGM-II filter density converges in probability to the true filter density under certain assumptions. We showed in Raihan & Chakravorty (2018) that under the condition of exponential forgetting of initial conditions, the true filter density can be approximated arbitrarily well with arbitrarily high confidence given that the sampling error in each step is small. We establish a similar result in the following. Define: $P(\hat{\pi}_{n-1}) \equiv \hat{\pi}_n^- = \sum_{i=1}^{M^-(n)} \hat{\omega}_i^-(n)\mathcal{G}_i(x; \hat{\mu}_i^-(n), \hat{P}_i^-(n))$, $\hat{P}(\hat{\pi}_{n-1}) \equiv$

$\hat{\hat{\pi}}_n^- = \sum_{i=1}^{M^-(n)} \hat{\hat{\omega}}_i^-(n)\mathcal{G}(x; \hat{\hat{\mu}}_i^-(n), \hat{\hat{P}}_i^-(n))$, $F_{z_n}(\hat{\pi}_{n-1}) = \sum_{i=1}^{M(n)} \hat{\omega}_i(n)\mathcal{G}(x; \hat{\mu}_i(n), \hat{P}_i(n))$, $\hat{F}_{z_n}(\hat{\pi}_{n-1}) = \sum_{i=1}^{M(n)} \hat{\hat{\omega}}_i(n)\mathcal{G}(x; \hat{\hat{\mu}}_i(n), \hat{\hat{P}}_i(n))$.

The above results represent the true and the approximate PGM predicted and filtered densities at time n given the approximate density $\hat{\pi}_{n-1}$ at time $n-1$. We have the following result:

Lemma 1. *Given the GMM representation of the prior pdf above, and a perfect Clustering algorithm $\mathcal{C}$, given any $\epsilon' > 0$, and $\delta' > 0$, there exists an $N_{\epsilon',\delta'}(n) < \infty$ such that: if the number of samples used to approximate the predicted pdf at time n is greater than $N_{\epsilon',\delta'}(n)$ then:*

$$Prob(|\hat{\hat{\omega}}_i^-(n) - \hat{\omega}_i^-(n)| > \epsilon') < \delta', \tag{15}$$

$$Prob(|\hat{\hat{\mu}}_i^{j-}(n) - \hat{\mu}_i^{j-}(n)| > \epsilon') < \delta', \tag{16}$$

$$Prob(|\hat{\hat{P}}_i^{jk-}(n) - \hat{P}_i^{jk-}(n)| > \epsilon') < \delta', \tag{17}$$

for all i, j, k, where $\hat{\mu}_i^{j-}$ represents the jth element of the mean vector $\hat{\mu}_i^-$ and $\hat{P}_i^{jk-}$ represents the (j, k)th *element of the covariance matrix $\hat{P}_i^-$.*

It must be noted that perfect clustering algorithm is an idealized assumption. In practice, cluster assignment, moment calculation, etc. are prone to errors.

Lemma 2. *Let $|\hat{\hat{\omega}}_i^-(n) - \hat{\omega}_i^-(n)| < \epsilon'$, $|\hat{\hat{\mu}}_i^{j-}(n) - \hat{\mu}_i^{j-}(n)| < \epsilon'$, and $|\hat{\hat{P}}_i^{jk-}(n) - \hat{P}_i^{jk-}(n)| < \epsilon$ for all i, j, k. Then, given that the state of the system $x \in \Re^d$, there exists $C^-(n) < \infty$ such that $\|\hat{\hat{\pi}}_n^- - \hat{\pi}_n^-\| < C^-(n)d\epsilon'$.*

Lemmas 1 and 2 are proved in Raihan & Chakravorty (2018).

Lemma 3. *Let, $\|\hat{\hat{\pi}}_n^- - \hat{\pi}_n^-\| < \epsilon^-$, then given the posterior $\hat{\hat{\pi}}_n^* = F_{z_n}(\hat{\pi}_{n-1})$, there exists $k(n) < \infty$ s.t: $\|\hat{\hat{\pi}}_n^* - \hat{\pi}_n\| < k(n)\epsilon^-$.*

Proof. Let $K_1 = \int p_n(z_n|x')\hat{\pi}_n^-(x')dx'$, $K_2 = \int p_n(z_n|x')\hat{\hat{\pi}}_n^-(x')dx'$. Then

$$\|\hat{\hat{\pi}}_n^* - \hat{\pi}_n\| = \int |(\frac{\hat{\pi}_n^-(x')}{K_1} - \frac{\hat{\hat{\pi}}_n^-(x')}{K_2})p_n(z_n|x')|dx' \tag{18}$$

$$= \int |(\frac{\hat{\pi}_n^-(x') - \hat{\hat{\pi}}_n^-(x') + \hat{\hat{\pi}}_n^-(x')}{K_1} - \frac{\hat{\hat{\pi}}_n^-(x')}{K_2})p_n(z_n|x')|dx'$$

$$\le \int |\frac{K_2 - K_1}{K_1K_2}|\hat{\hat{\pi}}_n^-(x')p_n(z_n|x')dx' + \int |\hat{\pi}_n^-(x') - \hat{\hat{\pi}}_n^-(x')|\frac{p_n(z_n|x')}{K_1}dx' \tag{19}$$

$$= |\frac{K_2 - K_1}{K_1}| + \max_{x'} \frac{p_n(z_n|x')\delta^-}{K_1}$$

$$= \frac{|\int(\hat{\pi}_n^-(x') - \hat{\hat{\pi}}_n^-(x'))p_n(z_n|x')dx'|}{K_1} + \max_{x'} \frac{p_n(z_n|x')\epsilon^-}{K_1}$$

$$\le 2\max_{x'} \frac{p_n(z_n|x')\epsilon^-}{K_1} \tag{20}$$

Choosing $k(n) = 2\max_{x'} \frac{p_n(z_n|x')}{K_1}$ completes the proof.

Let $\hat{\hat{\pi}}_n^*$ be the exact posterior evaluated from the propagated PDF $\hat{\hat{\pi}}_n^-$. The filtered PDF $\hat{\hat{\pi}}_n$ is a GMM representation of $\hat{\hat{\pi}}_n^*$. By Lemma 1, there exists an upper bound on the number of samples $N^*_{\epsilon',\delta'}$ such that the mixture parameters of $\hat{\hat{\pi}}_n^*$ are estimated with an accuracy of ϵ' with a confidence $1-\delta'$ if the MCMC draws these many samples. Let the number of particles used in PGM-II filter be $N = \max(N_{\epsilon',\delta'}, N^*_{\epsilon',\delta'})$. Therefore: $\|\hat{\pi}_n - \hat{\hat{\pi}}_n\| = \|\hat{\pi}_n - \hat{\hat{\pi}}_n^* + \hat{\hat{\pi}}_n^* - \hat{\hat{\pi}}_n\|, \le \|\hat{\pi}_n - \hat{\hat{\pi}}_n^*\| + \|\hat{\hat{\pi}}_n^* - \hat{\hat{\pi}}_n\|$.

From Lemmas 1–3, we have $Prob(\|\hat{\pi}_n - \hat{\hat{\pi}}_n^*\| > k(n)C^-(n)d\epsilon') < \delta'$, from Lemmas 1 and 2, we also have $Prob(\|\hat{\hat{\pi}}_n^* - \hat{\hat{\pi}}_n\| > C(n)d\epsilon') < \delta'$. Clearly, $Prob\Big(\|\hat{\pi}_n - \hat{\hat{\pi}}_n\| > \big(k(n)C^-(n) + C(n)\big)d\epsilon'\Big) < 2\delta'$. Hence, by choosing ϵ' such that $\epsilon = \big(k(n)C^-(n) + C(n)\big)d\epsilon'$, and δ' such that $\delta = 2\delta'$, and $N = \max(N_{\epsilon',\delta'}, N^*_{\epsilon',\delta'})$, we get $Prob(\|\hat{\pi}_n - \hat{\hat{\pi}}_n\| > \epsilon) < \delta$.

This proves that if the number of samples used to approximate the predicted and posterior GMM parameters are more than N, then the sampling error stays within the desired bounds with confidence $1-\delta$. Assuming that the underlying Markov chain has the exponential forgetting property, this suffices to show the convergence in probability of the PGM-II density to the true filter density identical to Lemma 2 in Raihan & Chakravorty (2018).

5. Numerical examples

In this section, we employ the PGM-II Filter in the estimation of two test case systems to study the filtering performance. The results are compared with that of other nonlinear filters such as UKF, PF, PGM-I filter and Blob filter. A basic sequential importance resampling (SIR) implementation of the PF is considered. The PGM-I variant which uses the unscented transform to perform the measurement update, i.e., the PGM-I(UT) filter, is used in this comparison study (Raihan & Chakravorty, 2016). The estimation results are compared for their accuracy, consistency and informativeness. The accuracy of estimates is evaluated in terms of a Monte Carlo averaged root mean squared error ($E_{rms}(t)$). The value of $E_{rms}(t)$ is computed as $E_{rms}(t) = \sqrt{\frac{1}{N_{Mo}}\sum_{j=1}^{N_{Mo}} \|x_{j,t} - \mu_{j,t}\|_2^2}$, where $x_{j,t}$ and $\mu_{j,t}$ represent the actual and estimated states at the time instant t during the jth Monte Carlo run. Also evaluated is the time averaged error ($\overline{E_{rms}}$) given by $\overline{E_{rms}} = \frac{1}{T}\sum_{t=1}^{T} E_{rms}(t)$.

The NEES test is employed to evaluate the consistency of the filtered PDF. The NEES test statistic ($\beta_{j,t}$) for a unimodal Gaussian PDF is given by $\beta_{j,t} = (x_{j,t} - \mu_{j,t})^T \mathbf{P}_{\mathbf{j,t}}^{-1}(\mathbf{x}_{\mathbf{j,t}} - \mu_{\mathbf{j,t}})$, where $P_{j,t}$ represents the covariance of the filtered PDF at time t during jth Monte Carlo run. The Monte Carlo averaged NEES test (β_t) is computed from this expression as $\beta_t = \frac{1}{N_{Mo}}\sum_{j=1}^{N_{Mo}} \beta_{j,t}$. When $x \in \mathbb{R}^n$ is distributed normally, the statistic given by $N_{Mo}\beta_t$ is distributed according to a χ^2 distribution with nN_{Mo} degrees of freedom.

The informativeness of estimates is compared in terms of the volume of $2-\sigma$ uncertainty region (Raihan & Chakravorty, 2016). When the state PDF is represented by a GMM, this volume can be computed as the sum of the $2-$*sigma* volumes of individual mixture modes, i.e., $V\sigma_2 = \sum_{i=1}^{L} |2\Sigma_i|$. Note that when the mixture modes overlap, measuring their total volume as given above will lead to underestimating the informativeness of the estimate.

5.1. Example 1

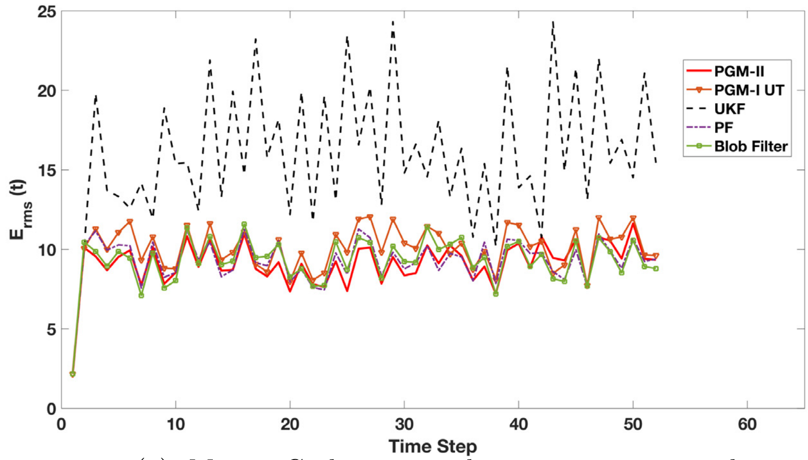
In test case 1, we consider a variant of the well known one dimensional estimation problem (Gordon et al., 1993)

$$x_k = \frac{x_{k-1}}{2} + \frac{25x_{k-1}}{1 + x_{k-1}^2} + 8\cos[1.2(k-1)] + w_{k-1} \tag{21}$$

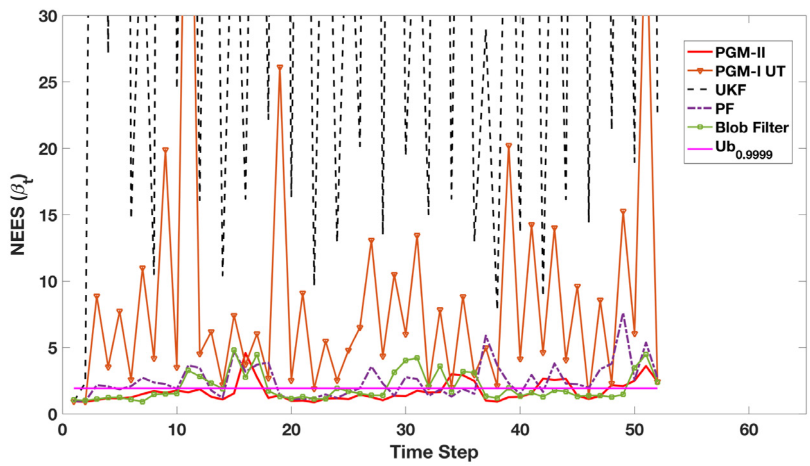
We define a multimodal measurement function

$$z_k = 4\sin(8x_k) + \nu_k. \tag{22}$$

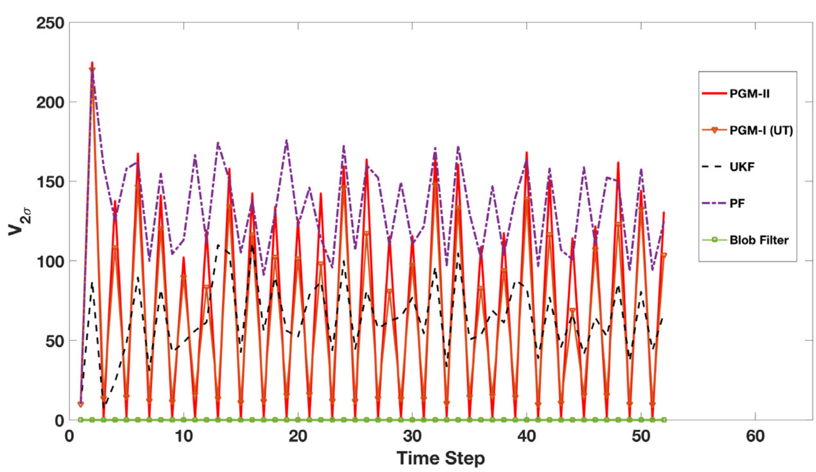
The process and measurement noises are assumed to be independent zero mean Gaussian random variables with covariances $Q = 6, R = 0.1$, respectively. Measurements are recorded at every other instant. The estimation is performed for a duration of 50 time steps and repeated over 50 Monte Carlo runs. The PF is implemented as an SIR with 80 particles. The values of the parameters α, β, χ used



(a) Monte Carlo averaged root mean squared error ($E_{rms}(t)$)



(b) Monte Carlo averaged NEES test statistic(β_t)



(c) Average volume of $2-\sigma$ ellipse ($V_{2\sigma}(t)$)

Fig. 3. Results: Test case 1.

in the UKF are 1.3, 1.5 and 0.2. The PGM-II filter and the PGM-I filter are employed with 80 particles and a maximum number of 6 mixture components. For the blob filter, 80 Gaussians with a maximum covariance of 10^{-4} was used in the estimation process.

The Monte Carlo averaged RMSE results (E_{rms}) are plotted in Fig. 3a. The PGM-II filter, blob filter and the PF are seen to outperform the UKF by a large margin. The tracking performance of PGM-II filter is also found to be somewhat better than that of PGM-I filter.

Table 1
Case 1: Results.

	$RMSE_{pos}$	%cases above 99.99%U_b	$V_{2\sigma}$
PGM-II	9.1066	25	71.8883
PGM-I(UT)	10.0047	92.31	64.3722
UKF	15.9752	98.08	63.4955
PF	9.2925	59.62	131.5503
Blob filter	9.2737	34.62	0.0137

Table 2
Case 2: Terminal results.

	$RMSE_{pos}$	%cases above 99.99%U_b	$\hat{l}og(V_{2\sigma})$
PGM-II	2.8257	22	−21.8973
PGM-I(UT)	4.5577	68	−46.4367
UKF	9.0014	70	−45.2474
PF	9.2239	100	−359.4890
Blob filter	0.6999	4	−48.2278

The time averaged tracking error $\overline{E_{rms}}$ given in Table 2 underlines this observation. The results of NEES test plotted in Fig. 3b show that the UKF estimates overstep the 99.99% upper bound $Ub_{0.9999}$ during the entire duration of the simulation after $t = 1$. The PGM-II filter and the blob filter are seen to offer more consistent estimates that lie within the 99.99% upper bound. The total fraction of the simulated time ($\beta_{c\%}$) during which each filter offered consistent estimates according to the $Ub_{0.9999}$ can be computed. The values of $\beta_{c\%}$ for all five filters are also listed in Table 1. The NEES results indicate that the PGM-II filter outperforms the blob filter, PF, PGM-I filter and the UKF. Finally, the Monte Carlo averaged $2-\sigma$ volumes for each of the five filters are plotted in Fig. 3c. The time averaged values of the $2-\sigma$ volumes are listed in Table 1. The blob filter is seen to have the smallest time averaged $2-\sigma$ volumes.

5.2. Example 2

In this example, we evaluate the performance of PGM-II filter in the so called "Blind tricyclist" problem proposed in Psiaki (2013). As the name suggests, the Blind tricyclist problem involves the estimation of the state of a blind tricyclist steering across an amusement park. The blind tricyclist is given the speed and steering angle time histories as inputs so that he can navigate across the park. However, his initial position coordinates (X_1, X_2) and heading angle (X_3) are unknown to him. To assist the navigation, measurements are recorded, but only intermittently and they consist of the relative bearing angle between the tricyclists heading and the location of two friends who are riding merry-go-rounds. The blind tricyclist can distinguish between the measurements coming from the two friends. However he only knows the centers and radii of the merry-go-rounds with certainty. The initial rotation angles (X_4, X_5) and the fixed rotation rates (X_6, X_7) of the two merry-go-rounds are unknown. The objective of the blind tricyclist problem is to estimate the quantities $X_1, \dots, X_7$ at all times. Hence it is a seven dimensional nonlinear estimation problem that involves both static and dynamic parameters. Note that estimation errors for the static parameters, and consequently the blind tricyclist problem, do not satisfy the exponential forgetting criterion employed in PGM-I filter. The equations governing the evolution of the state variables can be found in Psiaki (2013). The relative bearing angle measurement between the blind tricyclist and the first merry-go-round at the instant k is given by

$$\psi_{1,k} = atan2(y_1 + \rho_1 \sin(X_4) - X_2 - b_r \sin(X_3)) \\ (x_1 + \rho_1 \cos(X_4) - X_1 - b_r \cos(X_3)) - X_3 + \nu_k. \tag{23}$$

Here, (x_1, y_1) represents the center of the first merry-go-round, ρ_1 represents its radius and b_r represents the distance between the point below the blind tricyclists head and the midpoint of the two rear wheels. The noise parameters used in simulating the blind tricyclist problem are given in Psiaki (2013). In the simulation of blind tricyclist problem, the PGM-I and PGM-II filters are implemented with 8000 particles where 10,000 particles were used in the SIR type PF implementation. The values of the parameters α, β, χ used in the UKF implementation are 0.01, 2, 0, respectively. The blob filter was implemented using 7000 Gaussian pdfs with the LMI upper bound on the mixture covariances chosen from (Psiaki, 2016). For the MCMC step, the length of the burn-in time is set to be 800. The propagated prior and the final posterior are represented using 8000 samples. During the measurement update, the 8000 samples get divided among the clusters, i.e. if there are 2 clusters, each cluster gets to sample $q*(8000/2)$ points using MCMC where q is a positive integer. The standard practice is to choose a large enough q, pick every qth point from the ensemble obtained from MCMC and discard the rest. Instead we cluster these samples and get functional representations of the component posteriors $\pi_{i,n}(x)$ using $q = 3$. After computing the posterior weights $w_i(n)$, we sample a set of 8000 particles from $\{w_i(n), \pi_{i,n}(x)\}$. The sampling covariance was chosen as $0.05 \times P_i^-(n)$ where $P_i^-(n)$ represents the ith propagated prior covariance given by the clustering algorithm. The maximum number of mixture components used during the clustering step for PGM-II filter is set to be three. Keeping the number of clusters small helps to keep the computational cost low. Additionally, it helps to keep the approximate propagated pdf more diffuse which improves the diversity of hypotheses that are sampled. However, in order for the filter to not assign disproportionate confidence in any single mode, the diagonal elements of the clustered prior covariance matrices are never allowed to fall below a certain lower bound. This helps to prevent the loss of diversity. It also makes the estimates less accurate. When the diagonal elements do fall below this threshold, they are updated artificially. The lower bounds used on the diagonal elements of the prior covariance are summed up in the vector V_{lb} below.

$$V_{lb} = \begin{bmatrix} 2.8 & 2.8 \, 4e-3 & 9.9e-2 & 9.9e-2 \, .2e-4 & 2e-4 \end{bmatrix}^T \tag{24}$$

The PGM-I filter is also implemented with a maximum number of 3 Gaussian components.

The accuracy and informativeness of the estimation results are analyzed using RMSE and $V_{\sigma 2}$ as in test case 1. However, the NEES test is performed as described in Psiaki (2013), i.e, by computing the fraction of the total number of Monte Carlo runs that produced NEES test statistic that falls within the 99.99% upper bound of a seven dimensional chi squared random variable. This upper bound is computed to be equal to $U_b = 29.8775$. The results obtained from 50 Monte Carlo runs of the Blind tricyclist problem are plotted in Fig. 4. The results show that by the end of the estimation process, the blob filter offers the most accurate and consistent estimates followed by the PGM-II filter and the PGM-I filter. The terminal RMSE position error, terminal % of cases where the NEES results are above 99.99% and the time averaged 2 sigma ellipse volume $V_{\sigma 2}$ are provided in Table 2. The value of $V_{\sigma 2}$ for the PF is seen to be smallest. However, this result must be analyzed in conjunction with the fact that the PF results are almost always inconsistent.

The results of the blind tricyclist estimation problem indicates certain important limitations associated with the implementation of PGM-II filter. In theory, the MCMC based measurement update is capable of sampling from any posterior probability distribution. It is also well suited for sampling in large dimensions in comparison to other approaches such as the importance sampling. However, when the target distribution is highly non-Gaussian, as in the blind tricyclist problem, the Markov chain can be slowly mixing. This can diminish the ability of the MCMC based approaches to sufficiently

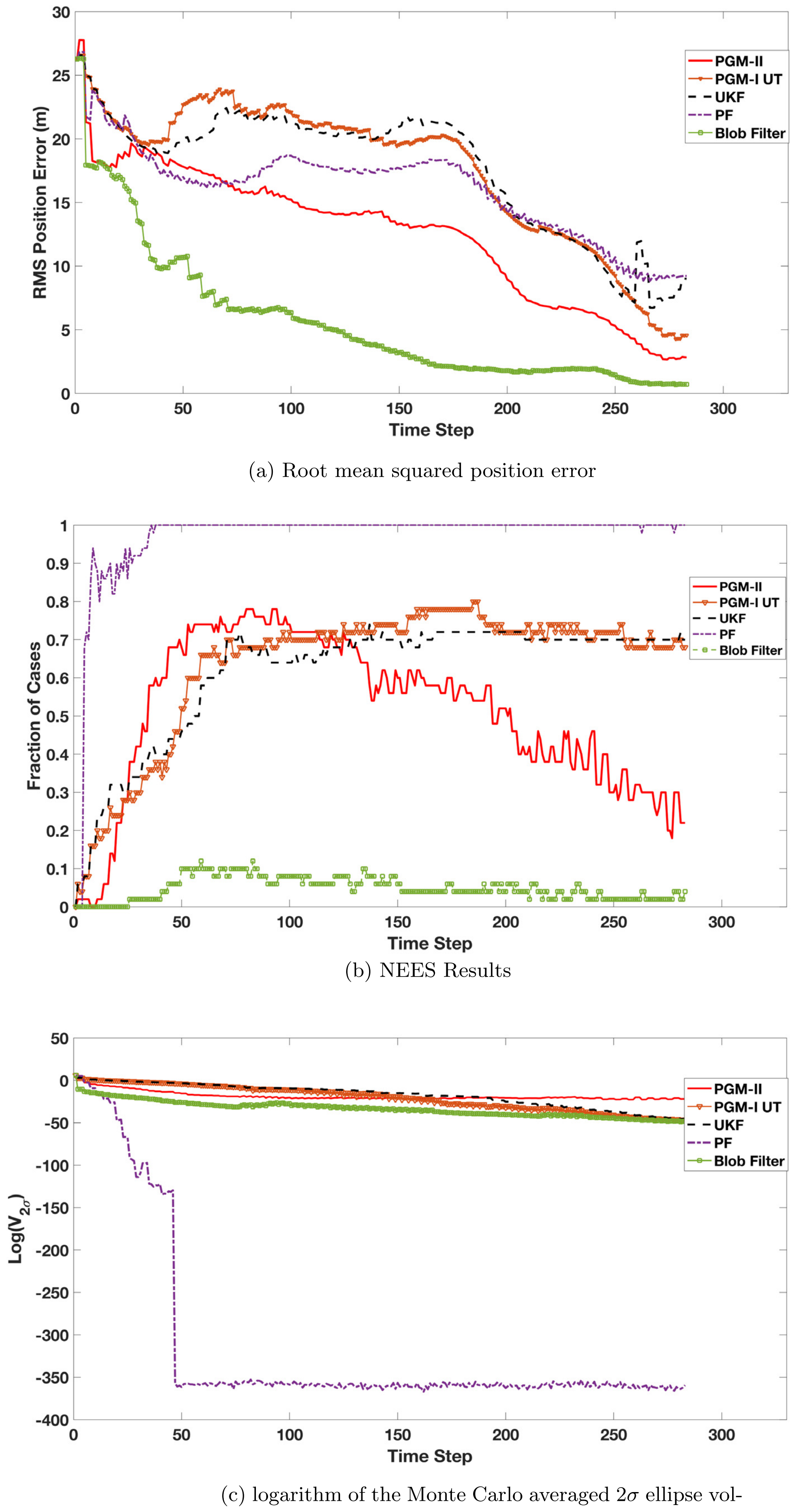


Fig. 4. Results: Test case 2.

explore the state space in a reasonable amount of time. The parallelized approach presented in this work was meant to alleviate this problem. The results indicate that this aspect of the problem requires further study. The bigger picture is that clustering might not be the best way to approximate non-Gaussian unimodal pdfs, and thus, more research is required into exploring "curse of dimensionality free density estimation schemes" for such pdfs. It must be observed that while the LMI based 'blob' filtering approach has several advantages over the conventional Gaussian sum filters (Alspach & Sorenson, 1972), the number of Gaussians used may still need to be increased exponentially with the dimension of the state space in order to cover the volume of a single Gaussian during reapproximation.

6. Conclusions

A novel particle Gaussian mixture filter that does not use a Kalman type linearized measurement update is presented in this paper. The proposed approach, termed the PGM-II filter, uses the transition kernels of the underlying Markov chain to generate samples during the propagation step. The samples are then clustered to recover a GMM representation of the propagated prior PDF. The measurement update is performed with the help of a parallelized MCMC based sampling algorithm. As a result, the PGM-II measurement update step is asymptotically exact and does not enforce restrictive assumptions on the number of mixture components. The PGM-II filter is employed in the estimation of two test cases to evaluate the estimation performance. The PGM-II filter is seen to outperform the PGM-I filter, the PF, and the UKF in both test cases. The blob filter is seen to offer superior performance in the blind tricyclist problem. It is demonstrated that the PGM-II filter is capable of handling the nonlinear/non-Gaussian measurement update. Strategies for improving the performance of the MCMC method in sampling extremely multimodal target densities need to be studied further.

References

Alspach, D. L., & Sorenson, H. W. (1972). Nonlinear Bayesian estimation using Gaussian sum approximations. *IEEE Transactions on Automatic Control*, *17*(4), 439–448.

Bengtsson, T., Bickel, P., & Li, B. (2008). Curse-of-dimensionality revisited: Collapse of particle filter in very large scale systems. In *Probability and statistics: Essays in honor of David A. Freedman, Vol. 2* (pp. 316–334).

DeMars, K. J., Bishop, R. H., & Jah, M. K. (2013). Entropy-based approach for uncertainty propagation of nonlinear dynamical systems. *Journal of Guidance, Control, and Dynamics*, *36*(4), 1047–1056.

Gelfand, A., & Dey, D. K. (1994). Bayesian model choice: Asymptotics and exact calculations. *Journal of the Royal Statistical Society. Series B. Statistical Methodology*, *56*, 501–514.

Gilks, W. R., Richardson, S., & Spiegelhalter, D. J. (1996). *Markov Chain Monte Carlo in practice*. London: Chapman and Hall.

Gordon, N., Salmond, D., & Smith, A. F. M. (1993). A novel approach to nonlinear/non-Gaussian bayesian state estimation. *IEEE Proceedings F, Radar and Signal Processing*, *140*(2), 107–113.

Hastings, W. K. (1970). Monte Carlo samping methods using Markov chains and their applications. *Biometrika*, *57*, 97–109.

Psiaki, Mark L. (2013). The blind tricyclist problem and a comparative study of nonlinear filters. *IEEE Control Systems Magazine*, *33*(3), 40–54.

Psiaki, Mark L. (2016). Gaussian mixture nonlinear filtering with resampling for mixand narrowing. *IEEE Transactions on Signal Processing*, *64*(21), 1047–1056.

Raihan, D., & Chakravorty, S. (2016). Particle gaussian mixture (pgm) filters. In *Proceedings of the 2016 19th international conference on information fusion* (pp. 1369–1376).

Raihan, D., & Chakravorty, S. (2018). Particle Gaussian mixture filters-I. *Automatica*. https://doi.org/10.1016/j.automatica.2018.07.023.

Richard , Duda, O., Hart, P. E., & Stork, D. G. (2000). *Pattern classification* (2nd ed.). New York: Wiley-Interscience.

Sorenson, H. W., & Alspach, D. L. (1971). Recursive Bayesian estimation using Gaussian sums. *Automatica*, *7*(4), 465–479.

VanDerwerken, D. N., & Schmidler, S. C. (2013). Parallel Markov Chain Monte Carlo. ArXiv e-prints, December. preprint, https://arxiv.org/abs/1312.7479.

Veettil, Dilshad Raihan Akkam, & Chakravorty, Suman (2016). Particle gaussian mixture (pgm) filters. Technical report, preprint, http://arxiv.org/abs/1603.04510.

Dilshad Raihan A.V. received his B.Tech in Mechanical Engineering in 2013 from the Indian Institute of Technology, Madras. He is now a Ph.D. student in the department of Aerospace Engineering at Texas A&M University, College Station. His main research interests are in Bayesian estimation and machine learning.

Suman Chakravorty obtained his B.Tech in Mechanical Engineering in 1997 from the Indian Institute of Technology, Madras and his Doctoral degree in Aerospace Engineering from the University of Michigan, Ann Arbor in 2004. From August 2004–August 2010, he was an Assistant Professor with the Aerospace Engineering Department at Texas A&M University, College Station and since August 2010, he has been an Associate Professor in the department. His broad research interests lie in the estimation and control of stochastic dynamical systems with application to autonomous, distributed robotic mapping and planning and situational awareness problems. He is a member of AIAA, ASME and IEEE. He is an Associate Editor for the ASME Journal on Dynamical Systems, Measurement and Control and the IEEE Robotics and Automation Letters.