CrossMark

A weighted likelihood criteria for learning importance densities in particle filtering

Muhammad Javvad ur Rehman[1,2], Sarat Chandra Dass[1*] and Vijanth Sagayan Asirvadam[3]

Abstract

Selecting an optimal importance density and ensuring optimal particle weights are central challenges in particle-based filtering. In this paper, we provide a two-step procedure to learn importance densities for particle-based filtering. The first stage importance density is constructed based on ensemble Kalman filter kernels. This is followed by learning a second stage importance density via weighted likelihood criteria. The importance density is learned by fitting Gaussian mixture models to a set of particles and weights. The weighted likelihood learning criteria ensure that the second stage importance density is closer to the true filtered density, thereby improving the particle filtering procedure. Particle weights recalculated based on the latter density are shown to mitigate particle weight degeneracy as the filtering procedure propagates in time. We illustrate the proposed methodology on 2D and 3D nonlinear dynamical systems.

Keywords: Nonlinear state-space models, Particle filter, Ensemble Kalman filter, Gaussian mixture models, Expectation-maximization (EM) algorithm

1 Introduction

For the most general forms of dynamical systems involving non-linear and non-Gaussian components, particle filters (PFs) constitute a class of methods that are able to infer the underlying filtered densities without restrictive assumptions. PFs consist of a collection of particles and weights that are updated and then propagated sequentially over time via Bayes rule. The weight and particle pairs at each time approximate the true filtered distribution in the Monte Carlo sense [36, 37]. Sequential particle filtering, thus, provides a convenient non-parametric way to approximate successive filtered distributions [34]. The nonparametric nature of PFs enables it to be applied to all state space models (linear as well as nonlinear) where the errors arise from general (i.e., non-Gaussian) distributions as well as in hierarchical models; see, for example, [7]. The exact solution using PF requires an infinite number of samples, so in practice, a large number of particles are generated. The particles are then propagated using recursive forward filtering based on procedures such as sequential importance sampling (SIS) and resampling

(SIR); see [2]. More recently, [22] reviewed resampling methods for PFs and discussed their implementation.

Other recursive forward filtering procedures, variants of the basic SIS and SIR, have also been reported in the literature. These include the auxiliary PFs (APFs), regularized PFs, the "likelihood" PF, etc; see [2] for details. These filtering procedures choose a variety of importance (i.e., proposal) densities that should ideally capture the overall form of the target density, i.e., the filtered density. Since in many situations the filtered density is not available in the closed and tractable form, choosing the importance density is not so straightforward. In [2], the authors highlight this problem, and at the same time, emphasize the importance of choosing the correct importance density to avoid particle and weight degeneracy.

Kalman-type filters and its extensions (i.e., the unscented KF (UKF), extended KF (EKF), and ensemble KF (EnKF)) use linearization techniques to arrive at filtering equations for nonlinear systems. When combined with particle-based methods, KF-type particle filtering can give rise to effective filtering methods in the presence of nonlinearity and non-Gaussianity. Used in combination with PFs, these methods construct a multitude of intermediate importance densities, via linearization, to generate particles and weights. A special case of such methods, the unscented particle filter (UPF), is discussed

*Correspondence: saratcdass@gmail.com
[1]Fundamental and Applied Sciences Department, Universiti Teknologi Petronas, Seri Iskandar, Malaysia
Full list of author information is available at the end of the article

Rehman *et al. EURASIP Journal on Advances in Signal Processing*   (2018) 2018:36

Page 2 of 19

in [27] for predicting the life of lithium ion batteries based on a localized UKF filter. Zuo et al. [47] propose a KF-type particle filtering framework in which the UKF is used during the importance sampling step. A truncated version of the UPF has been proposed by Straka et al. [40] when the distribution of measurement noise has bounded support.

Another combination involving PFs and KF-type filters is the Ensemble Kalman Particle Filters (EnKPF). The EnKPF incorporates EnKF methodology into the PF framework by combining the advantages of both and controlling the extent of the contribution of each method via a tuning parameter [12]. Localized versions of the EnKPF, localized within a grid set or using only nearby observations, for data assimilation is developed and discussed in [28, 35] for meteorological applications. An improved PF is proposed by [6] where the EnKF kernel is used to generate a multitude of importance densities at the current step for each particle obtained from the previous step. In [6], MCMC-based resampling is also performed to avoid particle impoverishment. A development of a weighted ensemble transformed Kalman filter for the non-linear image reconstruction is proposed in [5]. Proposal densities based on the EnKF filter in which the distribution is based on a sequence of previous measurements is discussed in [31]. A progressively corrected regularized particle filter is proposed in [30] to improve the nonparametric signal estimation. A recursive estimation scheme for a non-linear dynamical system is proposed in [15] where state estimation is performed based on the progressive processing. A brief survey highlighting the research gaps in state space estimation domain, giving specific attention to non-linear systems with informative observations, is reported in [25] where a modeling free solution is proposed and referred to as observation only (O_2) inference. In O_2 inference, the state estimates are directly calculated from observations [23].

The challenge of selecting an optimal importance density is closely related to the problem of weight degeneracy or particle impoverishment of the PF. Sub-optimal choices of the importance density, which deviate too far away from the targeted filtered density, give rise to importance weights that are severely skewed. Several methods are proposed in the literature to deal with weight degeneracy and particle impoverishment. In [13], an improvement of estimation accuracy is reported with the use of a smaller number of particles while maintaining particle diversity. An equivalent weights particle filter is proposed in [1] where the proposed importance density ensures that the particles end up in high probability region of posterior. In [41], particle impoverishment and sample size dependency problems are reported, and a particle swarm optimization procedure is proposed in the context of a genetic particle filter. An improved particle filter (IPF) is proposed for GPS/INS navigation system in which biases

are estimated in the first stage and then corrected for the predicted particles [43]. After this bias correction, recalculation of particle weights and resampling of particles are carried out. Another IPF was proposed in [46] based on a two-step procedure: In the first step, a standard importance density was used to simulate particles and to calculate the importance weights. In the second stage, weight optimization was performed by a pre-specified weight scaling factor, after which the particles generated in the first stage were resampled according to these new weights.

In this paper, we propose a two-step particle filtering procedure that mitigates weight degeneration. In the first step, we adopt localization to construct an importance density based on the Ensemble Kalman Filter (EnKF). This EnKF is similar to the procedure outlined in [6] but not identical to it. The second stage importance density is learned from the first stage particle and weight pairs via weighted likelihood criteria. The two-step procedure is similar to the two-step procedures of [43, 46] in that an initial pre-specified importance density is used to generate particles and weights. However, this paper is different from [43, 46] in terms of the adjustments that we perform to improve the first stage procedure. Instead of re-calibrating as in [43] or re-scaling weights as in [46], we recompute weights based on a learned importance density. We present justification so as to why the second stage weights mitigate particle impoverishment: The second stage weights are shown to be more uniformly distributed as a result of the learned importance density being close to the true but unknown filtered density, which is the target of our estimation based on the weighted likelihood criteria.

The second stage proposal density is learned from the class of Gaussian mixture models (GMMs). An expectation-maximization (EM) algorithm is developed for estimating the number of mixture components as well as GMM parameters. Note that learning importance densities based on GMMs and likelihoods have been reported in the literature as in [33]. However, the GMM model in [33] is fit to resample of particles, and thus, is subject to the variability of resampling. In our case, the weighted likelihood criteria do not depend on any resampling of particles from the set of particles and weights.

The remainder of the paper is organized as follows. Section 3.1 gives the preliminaries of particle-based filtering, while Section 3.2 presents the preliminaries of Gaussian mixture models (GMM) and the standard expectation-maximization (EM) algorithm for fitting GMMs to observed data. Section 4 presents the two-step particle filtering (TS) procedure. The first step develops the EnKF methodology in Section 4.1 for constructing an importance density. The second step where GMMs are learned via weighted likelihoods is presented

in Section 4.2. The proposed EM algorithm is adopted to weighted, rather than un-weighted (or standard), likelihoods. To validate the TS procedure, three methods are presented: the root mean square error (RMSE), the highest posterior density (HPD), and the effective sample size criteria in Section 5. Section 6 presents two examples (2D and 3D), simulated under various noise levels of the state space and observation models, to investigate the robustness of the proposed filtering procedure. Conclusions and future work are presented in Section 8.

2 Methods

The aim of our study is to select an optimal importance density in particle-based filtering. The importance density is learned by fitting Gaussian mixture models based on the maximum weighted likelihood criteria. It is shown that the resulting two-step (TS) procedure is less prone to degeneration of particles and weights. For comparing our proposed TS procedure with several other filtering procedures in the literature, we conduct simulation experiments based on dynamical systems that have been reported in the literature. Based on observations obtained from the simulation experiments, we carry out filtering steps for the selected procedures and compare their performances using several criteria such as root mean square error (RMSE), the extent of coverage by highest posterior density (HPD) sets, and values of effective sample size. All relevant statistical methodologies, such as maximum likelihood estimation, Gaussian mixture models, Bayesian HPD sets and others, as well as comparison criteria used, such as RMSE, HPD sets, and effective sample size, have been clearly described in the subsequent sections. Our study involves simulation codes developed using licensed MATLAB software; no human subjects were involved.

3 Preliminaries

State space modeling gives a unified framework for eliciting temporal dynamics of both linear and non-linear systems. State space modeling consists of two stages: (i) a model that describes underlying temporal system dynamics, called the state space model, and (ii) the measurement state model which relates the observations to the state space variables via noise factors. The discrete time stochastic system representing (i) and (ii), respectively, is given by

$$x_n = \Phi_n(x_{n-1}) + u_n, \quad \text{and} \tag{1}$$

$$y_n = \Psi_n(x_n) + v_n, \tag{2}$$

for $n = 1, 2, \cdots, T$, with T denoting the final time index and x_0 denoting the initial state vector. In (1) and (2), u_n and v_n, respectively, are the state and measurement noise random variables assumed to have known distributions f_n and g_n, respectively. We denote the state space and measurement model noise by

$$\tilde{Q}_n = var(f_n) \quad \text{and} \quad \tilde{R}_n = var(g_n), \tag{3}$$

respectively, keeping in mind that $\tilde{Q}_n$ and $\tilde{R}_n$ will be matrices (i.e., variance-covariance matrices) in the multivariate setting. The functions Φ_n and Ψ_n represent known nonlinear functions of the state space and measurement models, respectively. Given the observations $y_1, y_2, \cdots, y_T$, the aim is to estimate the underlying state vectors $x_0, x_1, \cdots, x_T$.

We introduce some notations for the subsequent presentation. The notations $n_1 : n_2$ and $a_{n_1:n_2}$ represent vectors of indices $(n_1, n_1 + 1, \cdots, n_2)$ and $(a_{n_1}, a_{n_1+1}, \cdots, a_{n_2})$ for any attribute a, respectively. The underlying state and observation vectors at time n are denoted by $x_n \in R^r$ and $y_n \in R^s$, respectively, where r and s represent the dimensions of the corresponding spaces. We also do not, at present, consider any unknown parameters in the model of (1) and (2); all quantities are assumed known except for the underlying state vectors $x_{0:T}$. The goal, therefore, is to obtain the filtered density of x_n at each n based on all observations $y_{1:n}$. A Bayesian approach provides a convenient framework for finding all filtered (target) densities [2]. In the Bayesian framework, the initial state space vector x_0 is assumed to follow a known prior density, p_0. In subsequent text, we use the notation $p(a \mid b)$, for random vectors a and b, to denote the conditional density of a given b.

3.1 Particle filters (PF)

Recursive sequential updating via Bayes rule [36, 37] is the best way to obtain all filtered densities successively. Assuming that the filtered density $p(x_{n-1} \mid y_{1:n-1})$ at step $n - 1$ is available, the n-th step filtered density is given by

$$p(x_n \mid y_{1:n}) = \frac{p(y_n \mid x_n)p(x_n \mid y_{1:n-1})}{p(y_n \mid y_{1:n-1})} \tag{4}$$

where

$$p(x_n \mid y_{1:n-1}) = \int_{R^r} p(x_n \mid x_{n-1})p(x_{n-1} \mid y_{1:n-1})dx_{n-1} \tag{5}$$

is the predictive distribution for x_n given $y_{1:n-1}$. Closed-form expressions of the filtered densities in (4) typically cannot be obtained for non-linear state space models, as is well-known and numerical techniques and approximations are, therefore, needed.

Recursive sequential particle filtering provides a convenient non-parametric way to approximate successive filtered densities [34]. A set of particles and corresponding weights, $\{x_n^i, W_n^i\}_{i=1}^M$, for $n = 1, 2, \cdots, T$ are propagated over time so that at every step n, the filtered density

Rehman *et al. EURASIP Journal on Advances in Signal Processing*   (2018) 2018:36

Page 4 of 19

$p(x_n \,|\, y_{1:n})$ is represented in the Monte Carlo sense [14] for large M as

$$p(x_n \,|\, y_{1:n}) \overset{d}{\approx} \sum_{i=1}^{M} W_n^i \, \delta_{x_n^i}(x_n), \qquad (6)$$

where $\delta_u(x)$ is the Dirac function that takes values 1 and 0 according to whether $x = u$ or otherwise. The normalized weights $W_n^i = w_n^i / \sum_{i=1}^{M} w_n^i$ are obtained from the (unnormalized) weights w_n^i which satisfy the recursive relation

$$w_n^i = w_{n-1}^i \frac{p\left(y_n \,|\, x_n^i\right) p\left(x_n^i \,|\, x_{n-1}^i\right)}{q\left(x_n^i \,|\, x_{n-1}^i, \, y_{1:n}\right)}, \qquad (7)$$

with $q(x_n^i \,|\, x_{n-1}^i, \, y_{1:n})$ being the n-th step importance density. Thus, $x_n^i \sim q(x_n^i \,|\, x_{n-1}^i, \, y_{1:n})$ for $i = 1, 2, \cdots, M$ are the M samples generated from it given the previously available particles x_{n-1}^i. The Recursive filtering performed using this sequential importance sampling (SIS) framework is described in Table 1.

PFs based on SIS suffer from weight degeneration. As n becomes large, PFs puts more weight on fewer and fewer particles, and finally, to just a singleton particle [4, 39]. As a result, PF estimates of filtering densities $p(x_n \,|\, y_{1:n})$ become increasingly unreliable as only a few particles become relevant. The source of the problem lies in the choice of the importance density q which cannot always be ideally taken to be the true filtered density (which is unknown). PF literature addressing weight degeneration discusses and develops different choices of the importance density $q(x_n \,|\, x_{n-1}, \, y_{1:n})$. Some of the notable works have been discussed in the "Introduction" section whereas others include [1, 8, 17, 21]. Many of the earlier proposed methods do not give satisfactory results whenever q deviates significantly from the ideal choice. Resampling has also been suggested as a partial solution to the weight degeneration problem. After obtaining the ensemble of particles and weights according to SIS, $\{x_n^{*i}, W_n^{*i}\}_{i=1}^{M}$, the sequential importance resampling (SIR) filter will resample particle x_n^{*i} with probability W_n^{*i}. The output of an SIR filter is the resampled particles with equal weights given

by $\{x_n^i, 1/M\}_{i=1}^{M}$. Table 2 gives the SIR procedure. For the SIR, it may happen that low weights at current step n actually correspond to high weights in step $n + 1$ in which case the resampling step loses important information and again causes weight degeneration [18]. Since most of the time high weights are statistically picked, resampling also leads to the loss of heterogeneity of the particles [34].

3.2 Gaussian mixture models (GMMs)

In this section, we present the class of Gaussian mixture densities and associated algorithms of sampling and learning (i.e., estimation of its parameters) which will be needed for the development of our two-step PF procedure. Gaussian mixture densities are a semi-parametric class of pdfs which can adequately represent any density by choosing a sufficiently large number of mixture components. The class of d-variate Gaussian mixture models (GMMs) is given by

$$f(x; \theta) = \sum_{g=1}^{G} \pi_g \phi_d \left(x; \mu_g, \Sigma_g \right) \qquad (8)$$

where G is the number of mixture components, $\pi_g, \, g = 1, 2, \cdots, G$ are non-negative mixture weights summing to 1, and ϕ_d is the pdf of a d-variate normal with mean μ_g and covariance matrix Σ_g, respectively. The parameter $(G; \theta) = (G; \pi_g, \, g = 1, 2, \cdots, G; \mu_g, \, g = 1, 2, \cdots, G; \Sigma_g, \, g = 1, 2, \cdots, G)$ represents all quantities that define the pdf of a GMM. Sampling from a GMM with known parameters $(G; \theta)$ can be easily carried out. To sample M realizations from (8), first sample the label of the mixture component $L^i \in \{1, 2, \cdots, G\}$ with probabilities $\pi_1, \pi_2, \cdots, \pi_G$, respectively, independently for $i = 1, 2, \cdots, M$. Then, conditional on $L^i = g$, sample x^i from the conditional Gaussian density $\phi_d \left(x \,|\, \mu_g, \, \Sigma_g \right)$, that is,

$$p\left(x^i \,|\, L^i = g; \theta \right) = \phi_d \left(x^i; \mu_g, \, \Sigma_g \right). \qquad (9)$$

In (9), we also make explicit the dependence of the conditional density (and, in fact, all subsequent densities) on θ. The joint density of the pair $\left(x^i, L^i \right)$ is

Table 1 Particle filtering using SIS

$\left[\{x_n^i, w_n^i\}_{i=1}^{M} \right] = \text{SIS} \left[\{x_{n-1}^i, w_{n-1}^i\}_{i=1}^{M}, \, y_n \right]$

– Initialize $x_0^i \sim p_0$ and $w_0^i = 1/M$ for $i = 1, 2, \cdots, M$.

– DO for $n = 1, 2, \cdots, T$:

– Draw $x_n^i \sim q\left(x_n \,|\, x_{n-1}^i, \, y_{1:n} \right)$, for $i = 1, 2, \cdots, M$.

– Calculate importance weights as
$$w_n^i = w_{n-1}^i \frac{p(y_n | x_n^i) \, p(x_n^i | x_{n-1}^i)}{q(x_n^i | x_{n-1}^i, y_{1:n})}$$

– Normalize : $W_n^i = w_n^i / \sum_{i=1}^{M} w_n^i$

– Propagate : $\{x_n^i, w_n^i\}_{i=1}^{M}$

Table 2 Particle filtering using SIR

$\left[\{x_n^i, 1/M\}_{i=1}^{M} \right] = \text{SIR} \left[\{x_{n-1}^i, 1/M\}_{i=1}^{M}, \, y_n \right]$

–Initialize $x_0^i \sim p_0$ and $w_0^i = 1/M$ for $i = 1, 2, \cdots, M$.

–DO for $n = 1, 2, \cdots, T$:

– Draw $x_n^{*i} \sim q(x_n \,|\, x_{n-1}^i, y_{1:n})$, for $i = 1, 2, \cdots, M$.

– Calculate importance weights as
$$w_n^{*i} = \frac{p(y_n | x_n^i) p(x_n^i | x_{n-1}^i)}{q(x_n^i | x_{n-1}^i, y_{1:n})}$$

– Normalize : $W_n^{*i} = w_n^{*i} / \sum_{i=1}^{M} w_n^{*i}$

– Resample $\{x_n^i\}_{i=1}^{M}$ from $\{x_n^{*i}\}_{i=1}^{M}$ with weights $\{ W_n^{*i} \}_{i=1}^{M}$

– Propagate : $\{x_n^i, 1/M\}_{i=1}^{M}$

Rehman *et al. EURASIP Journal on Advances in Signal Processing* (2018) 2018:36

Page 5 of 19

$$p\left(x^i, L^i; \theta\right) = \pi_{L^i} \phi_d\left(x^i; \mu_{L^i}, \Sigma_{L^i}\right), \qquad (10)$$

whereas the marginal distribution of x^i, by summing over different realizations of L^i in $\{1, 2, \cdots, G\}$ according to the probabilities $\pi_1, \pi_2, \cdots, \pi_G$, is precisely the GMM given in (8).

We assume that the number of components G is fixed and known for the subsequent discussion. When θ is unknown, a standard procedure for learning (i.e., estimating) θ based on M independent observations x^i, $i = 1, 2, \cdots, M$ from (8) is the expectation maximization (EM) algorithm [26]. We briefly describe the EM procedure as in [26] since the notations will be used to develop our EM algorithm for weighted likelihoods. The goal is to estimate θ by maximizing the (regular) likelihood

$$\mathbf{L}(\theta) \equiv \prod_{i=1}^{M} f\left(x^i; \theta\right) \qquad (11)$$

as a function of θ. This estimate of θ, known as the maximum likelihood estimate (MLE), is defined as

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}}\, \mathbf{L}(\theta) = \underset{\theta}{\operatorname{argmax}} \left(\prod_{i=1}^{M} f\left(x^i; \theta\right)\right). \qquad (12)$$

The EM algorithm is an iterative procedure that indirectly maximizes the likelihood of θ in (12) by incorporating auxiliary variables as missing observations (see [26] for details). The class label L^i is incorporated as the auxiliary variable for each observation x^i, $i = 1, 2, \cdots, M$, with the conditional distribution of L^i given x^i having the form

$$p(L^i \mid x^i; \theta) = \frac{p\left(x^i, L^i; \theta\right)}{f\left(x^i; \theta\right)} = \frac{\pi_{L^i} \phi_d\left(x^i; \mu_{L^i}, \Sigma_{L^i}\right)}{f(x^i; \theta)} \qquad (13)$$

based on (10).

The EM algorithm for finding $\hat{\theta}_{MLE}$ starts with an initial guess of θ, say $\theta^{(0)}$. At the k-th iteration, assume that $\theta^{(k)}$ has been obtained. At the $(k + 1)$-st step, the next iterate $\hat{\theta}^{(k+1)}$ is found as

$$\theta^{(k+1)} = \underset{\theta^*}{\operatorname{argmax}}\, Q\left(\theta^{(k)}, \theta^*\right) \qquad (14)$$

where $Q(\theta, \theta^*) = \sum_{i=1}^{M} \mathbf{E}\left[\log p\left(x^i, L^i; \theta^*\right)\right]$ and the expectation $\mathbf{E}$ is taken under the conditional probability distribution of L^i given x^i and θ in (13). The sequence of iterates of the likelihood $\prod_{i=1}^{M} f\left(x^i; \theta^{(k)}\right)$ for $k = 1, 2, \cdots$ can be shown to be non-decreasing and, thus, converges to a local maxima of the likelihood function in (12). Thus, starting from a initial value that is close enough to $\hat{\theta}$ guarantees that $\theta^{(k)}$ converges to the MLE $\hat{\theta}$ as $k \to \infty$. Properties of the standard EM algorithm and its application to GMMs are well known and we refer the interested reader to [26] for more details.

To select an appropriate value for G, Bayes information criteria (BIC) is used where

$$BIC = -2 \log\left(\prod_{i=1}^{M} f(x^i; \hat{\theta}_{MLE})\right) + \breve{p} \log(M), \qquad (15)$$

where $\breve{p} = (G - 1) + dG + Gd(d+1)/2$ is the number of parameters for a GMM with G mixture components. Typically, a maximum pre-specified number, G_0, is selected for the estimated number of mixture components, and the G value corresponding to the maximum BIC value in the range of $1 \leq G \leq G_0$ is selected as the estimated number of mixture components for the GMM.

4 A two-step (TS) procedure for particle filtering

We describe the proposed procedure for PF that reduces weight degeneration of particles in this section. This procedure is a recursive procedure. At the end of the $(n-1)$-th recursion, we assume that M particles, class labels (associated with GMMs which will be made clear later on), and weights, denoted by $\left\{x_{n-1}^i, L_{n-1}^i, w_{n-1}^i\right\}_{i=1}^{M}$, are available.

At the n-th stage, the procedure consists of two main steps: The first step involves constructing the importance density $q(x_n \mid x_{n-1}, y_{1:n}) \equiv q\left(x_n^i \mid x_{n-1}^i, y_{1:n}\right)$ for each i. This proposal density is selected based on the EnKF kernel separately for each i, similar to [6]; see also [10, 19] and [16]. Second, we learn the filtered density at each time step n, $p(x_n \mid y_{1:n})$ by fitting GMMs to a collection of samples and weights based on weighted likelihood criteria. The two-step procedure is outlined below and in Table 3.

- Initialize $x_0^i \sim p_0$ and $w_0^i = 1/M$ for $i = 1, 2, \cdots, M$.
- DO for $n = 1, 2, \cdots, T$:
- DO for each particle i, $i = 1, 2, \cdots, M$:
- STEP 1: Construct the EnKF importance density:

  1. $\left[\hat{x}_n^i, \hat{P}_n^i\right] = \text{EnKF}\left[x_{n-1}^i, L_{n-1}^i, y_n\right]$; see Section 4.1.
  2. Sample: Draw $x_n^{*i} \sim \phi_d\left(x \mid \hat{x}_n^i, \hat{P}_n^i\right)$ as in (18).
  3. Calculate weights

$$w_n^{*i} = w_{n-1}^i \frac{p\left(y_n \mid x_n^{*i}\right) p\left(x_n^{*i} \mid x_{n-1}^i\right)}{q\left(x_n^{*i} \mid \hat{x}_n^i, \hat{P}_n^i\right)}. \qquad (16)$$

- STEP 2: Learn $p(x_n \mid y_{1:n})$:

  1. Find its estimate, $\hat{f}_n(x)$, based on fitting GMMs using data $\left\{x_n^{*i}, w_n^{*i}\right\}_{i=1}^{M}$ from STEP 1; see Section 4.2.
  2. Sample: Draw $\left(x_n^j, L_n^j\right) \sim \hat{f}_n(x)$
  3. Compute weights

Rehman *et al. EURASIP Journal on Advances in Signal Processing* (2018) 2018:36

Page 6 of 19

Table 3 Two-step particle filtering (TS) algorithm

$\left[\{x_n^i,\, L_n^i,\, w_n^i\}_{i=1}^M\right] = \text{TS}\left[\{x_{n-1}^i,\, L_{n-1}^i,\, w_{n-1}^i\}_{i=1}^M,\, y_n\right]$

–Initialize $x_0^i \sim p_0$ and $w_0^i = 1/M$ for $i = 1, 2, \cdots, M$.

–DO for $n = 1, 2, \cdots, T$:

–DO for each particle i, $i = 1, 2, \cdots, M$:

– STEP 1: Construct the EnKF importance sampling density and sample

– from it:

   1. Construct: $[\hat{x}_n^i, \hat{P}_n^i] = \text{EnKF}[x_{n-1}^i, L_{n-1}^i, y_n]$.

   2. Sample: Draw $x_n^{*i} \sim \phi_d\left(x \mid \hat{x}_n^i, \hat{P}_n^i\right)$ as in (18).

   3. Calculate weights $w_n^{*i} = w_{n-1}^i \dfrac{p(y_n|x_n^{*i})p(x_n^{*i}|x_{n-1}^i)}{q\left(x_n^{*i}|\hat{x}_n^i,\hat{P}_n^i\right)}$.

– STEP 2: Learn $p(x_n \mid y_{1:n})$:

   1. Find its estimate, $\hat{f}_n(x)$, based on GMMs and data $\{x_n^{*i},\, w_n^{*i}\}_{i=1}^M$

– from STEP 1.

   2. Sample: Draw $(x_n^j, L_n^j) \sim \hat{f}_n(x)$

   3. Compute weights

$$w_n^j = \frac{1}{M}\sum_{i=1}^M \frac{w_{n-1}^i\, p\left(y_n \mid x_n^j\right) p\left(x_n^j \mid x_{n-1}^i\right)}{\hat{f}_n\left(x_n^j\right)}$$

– Propagate: $\left\{x_n^j,\, L_n^j,\, w_n^j\right\}_{j=1}^M$

$$w_n^j = \frac{1}{M}\sum_{i=1}^M \frac{w_{n-1}^i\, p\left(y_n \mid x_n^j\right)\, p\left(x_n^j \mid x_{n-1}^i\right)}{\hat{f}_n\left(x_n^j\right)}$$

(17)

– Propagate: $\left\{x_n^j,\, L_n^j,\, w_n^j\right\}_{j=1}^M$

The two steps involved are explained in greater detail in the following subsections.

4.1 Importance density based on EnKF kernel

In STEP 1, the choice of the importance density is developed by considering the Ensemble Kalman Filter (EnKF) [10, 16, 19] kernel. The key idea is to use the previous particle x_{n-1}^i at the $(n-1)$-th step to construct a separate proposal distribution for each $i = 1, 2, \cdots, M$. In this way, regions close to x_{n-1}^i can be explored leading to a choice of a localized importance sampling density that is ideal.

Fix i and recall that the particle and class label pair is x_{n-1}^i and $L_{n-1}^i = g$, say. The EnKF methodology entails the following subsequent steps:

– Sample N ensemble points $\left\{\chi_{n-1}^b\right\}_{b=1}^N$ from $\phi_d\left(x; x_{n-1}^i, \hat{\Sigma}_g\right)$, the Gaussian density with mean x_{n-1}^i and covariance matrix $\hat{\Sigma}_g$.

– Obtain N realizations of $\chi_{n|n-1}^b = \Phi_n\left(\chi_{n-1}^b\right) + u_n^b$, for $b = 1, 2, \cdots, N$, where u_n^b are samples from the distribution of errors for the state space model in (1).

– Obtain the mean and covariance matrix as

$$\hat{x}_{n|n-1}^i = \frac{1}{N}\sum_{b=1}^N \chi_{n|n-1}^b, \equiv \tilde{x} \text{ say, and}$$

$$P_{n|n-1}^i = \hat{\Sigma}_g + \frac{1}{N}\sum_{b=1}^N \left(\chi_{n|n-1}^b - \tilde{x}\right)\left(\chi_{n|n-1}^b - \tilde{x}\right)^T.$$

– Based on the measurement model (2), obtain the mean and covariance matrices for the observation process given by

$$\hat{y}_{n|n-1}^i = \frac{1}{N}\sum_{b=1}^N \Psi_n\left(\chi_{n|n-1}^{bi}\right) \equiv \tilde{y} \text{ say, and}$$

$$P_{yy}^i = \frac{1}{N}\sum_{b=0}^N \left(\Psi_n(\chi_{n|n-1}^{bi}) - \tilde{y}\right)\left(\Psi_n\left(\chi_{n|n-1}^{bi}\right) - \tilde{y}\right)^T$$

and the covariance between the state and observation processes given by

$$P_{xy}^i = \frac{1}{N}\sum_{b=0}^N \left(\chi_{n|n-1}^{bi} - \tilde{x}\right)\left(\Psi_n\left(\chi_{n|n-1}^{bi}\right) - \tilde{y}\right)^T.$$

– Apply the Kalman updating formulas

$$D_n^i = R_n + P_{yy}^i$$
$$K_n^i = P_{xy}^i \left[D_n^i\right]^{-1}$$
$$\hat{x}_{n|n}^i = \hat{x}_{n|n-1}^i + K_n^i\left(y_n - \hat{y}_{n|n-1}^i\right), \text{ and}$$
$$P_{n|n}^i = P_{n|n-1}^i - K_n^i D_n^i K_n^{i\,T}$$

where R_n is the measurement noise variance and K_n^i represents the EnKF Kalman gain matrix.

– Define

$$\hat{x}_n^i = x_{n|n}^i, \qquad \text{and} \qquad \hat{P}_n^i = P_{n|n}^i,$$

and set the EnKF importance sampling density, $q_{EnKF}\left(x \mid x_{n-1}^i, y_{1:n}\right)$, as

$$q_{EnKF}(x \mid x_{n-1}^i, y_{1:n}) = \phi_d\left(x \mid \hat{x}_n^i, \hat{P}_n^i\right), \qquad (18)$$

the Gaussian density with mean $\hat{x}_n^i$ and covariance $\hat{P}_n^i$.

We denote the procedure in STEP 1 as LLPF_{EnKF}, which can be seen to be a locally linearized PF (LLPF) based on the EnKF.

At the end of STEP 1, the resulting samples and weights $\{x_n^{*i}, w_n^{*i}\}_{i=1}^M$ (see Table 3) is an approximation to the filtered density at time step n, $p(x_n \mid y_{1:n})$, in the Monte Carlo sense of (6) for large M since we utilized the general SIS framework.

4.2 Learning GMMs via weighted likelihoods

We now describe STEP 2 of our two-step procedure outlined in Table 3. STEP 2 consist of learning a GMM from

the M particles and weights $\left\{ x_n^{*i}, w_n^{*i} \right\}_{i=1}^{M}$ obtained at the end of STEP 1. The M particle and weight pairs consists the "data" for a likelihood function from which we will estimate the parameters θ and G for the GMM defined in (8). This likelihood function is defined as

$$\mathbf{L}_w(\theta) = \prod_{i=1}^{M} f\left(x_n^{*i}; \theta\right)^{w_n^{*i}} \tag{19}$$

where $f(x; \theta)$ is the GMM defined in (8). Note that the above likelihood is a weighted version of the ordinary likelihood in (11) where each of the M terms in the weighted likelihood, $f\left(x_n^{*i}; \theta\right)$, is weighted by the corresponding weight term, w_n^{*i}, for $i = 1, 2, \cdots, M$. In view of the more general weighted likelihood formula in (19), the ordinary likelihood in (11) is a special case of (19) where the weights w_n^{*i} are constant. The weighted maximum likelihood estimator of θ, $\hat{\theta}_w$, is defined as

$$\hat{\theta}_w = \underset{\theta}{\operatorname{argmax}}\, \mathbf{L}_w(\theta) = \underset{\theta}{\operatorname{argmax}} \left(\prod_{i=1}^{M} f\left(x_n^{*i}; \theta\right)^{w_n^{*i}} \right) \tag{20}$$

and is obtained using an EM algorithm developed for the weighted likelihood; details are given in the Additional file 1: Section 2. To select an appropriate value for the number of mixture components, G, we use the BIC criteria as before as in (15) where now the likelihood $\mathbf{L}(\theta)$ is replaced by the weighted likelihood $\mathbf{L}_w(\theta)$ and the BIC is defined as $BIC = -2 \log \mathbf{L}_w\left(\hat{\theta}_w\right) + \breve{p} \log(M)$, where $\breve{p} = (G - 1) + dG + Gd(d + 1)/2$ is the number of parameters for a GMM with G mixture components. We define

$$\hat{f}_n(x_n) \equiv f\left(x_n; \hat{\theta}_w\right)$$

in STEP 2 of Table 3.

Remark 1 *The GMM class is a semi-parametric class that is flexible enough to approximate arbitrary densities by selecting a sufficiently large number of mixture components, G. The GMM class is used in our procedure to approximate the true filtered density at each time step n, $n = 1, 2, \cdots, T$, based on the weighted likelihood criteria. Additional file 1: Section 2.2 in the Appendix gives insight into how this is achieved. The relevance of the closeness of the GMM approximation to reduction in weight degeneracy of the TS procedure is also explained in detail in the Additional file 1: Section 2.2 of the Appendix.*

Remark 2 *Additional file 1: Section 2.3, we illustrate TS procedure in the case of Kalman filtering (i.e., linear systems with Gaussian noise). We show explicitly that even if the STEP 1 importance density is chosen sub-optimally, the weighted likelihood criteria in STEP 2 corrects and*

improves this sub-optimal choice and leads to a fitted GMM density that is close to the true filtered density. We demonstrate the utility of the BIC criteria which acts as a penalty term that penalizes spurious fits based on extra mixture components when they are unnecessary.

5 Monitoring weight degeneracy via RMSE, HPD sets, and N_{eff}

For a filtering procedure such as SIR, $LLPF_{EnKF}$, and TS, we evaluate its performance using the root mean square error (RMSE) criteria. The RMSE criteria for any estimator $\delta(y_{1:n})$ of x_n is defined as

$$\mathrm{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \left[\delta\left(y_{1:n}^i\right) - x_n^i \right]^2}, \tag{21}$$

based on N simulation experiments that generate the underlying state space variables $x_{0:T}^i$, and observations $y_{1:T}^i$ are given $x_{0:T}^i$, for $i = 1, 2, \cdots, N$, from (1) and (2), respectively. Note that the above RMSE is defined for every time step $n = 1, 2, \cdots, T$ for which a filtered distribution can be obtained based on the SIR, $LLPF_{EnKF}$, and TS procedures. The estimator $\delta(y_{1:n})$ will be taken to be the posterior mean of the filtered distributions $p(x_n \mid y_{1:n})$ for estimating the state space variable x_n. For a filtering procedure, let the filtered distribution at the n-th step be represented by the set of M particles x_n^i, $i = 1, 2, \cdots, M$ and weights w_n^i, $i = 1, 2, \cdots, M$. The posterior mean of the filtered distribution is calculated as

$$\delta(y_{1:n}) = \frac{\sum_{i=1}^{M} w_n^i x_n^i}{\sum_{i=1}^{M} w_n^i} \tag{22}$$

based on the M particles and weights and is taken to be the estimator of the state space variable x_n. Thus, different filtering procedures will give rise to different estimates of x_n.

We also consider another estimator in our experiments, namely, the observation-only (O_2) estimator reported in [23]. The O_2 estimator depends only on the measurement model (2) and is typically the maximum likelihood estimator.

Weight degeneracy or particle depletion over time is a common and well-known problem for PFs and any filtering procedure. In the ideal case, the importance density, $q(x_n \mid x_{n-1}, y_{1:n})$, should be identical to the filtered distribution, $p(x_n|y_{1:n})$, giving rise to equal weights. However, in most cases, a poorly chosen importance density causes weights to be starkly uneven, and with the propagation of time, weights increasingly concentrate on fewer and fewer particles.

Assessment of particle degeneracy can be carried out using the RMSE criteria. Particle degeneration affects the quality of the filtering, which in turn, affects the quality

of posterior mean calculated from the filtered distribution. When particle degeneracy is present, the posterior mean estimates will deviate far away from the true value of x_n. As a result, the RMSE values will be large. Typically, as n increases from 1 to T, the filtering performance deteriorates further and the RMSEs will show an increasing trend. This situation can be verified by observing that the mean and standard deviations of the RMSE (over $n = 1, 2, \cdots, T$) will both be large. We provide numerical examples in Section 6 based on simulation experiments.

Another assessment of weight degeneracy is based on the HPD sets which is described in Additional file 1: Section 1 for this article. Using the HPD sets, we show that the true state space vector x_n lies inside its 95% HPD set for each n. This coverage demonstrates that the filtering procedure under consideration does not suffer from particle weight depletion during the propagation of particles and weights; for if there was weight depletion at any stage, the HPD sets constructed thereafter would not cover the true value of x_n with high probability. The coverage probabilities of the HPD sets is then obtained by repeating the simulation experiments and checking whether x_n belongs inside its 95% HPD set or not for each simulated data. These coverage probabilities are reported in Section 6.

We also calculate the effective sample size [2]

$$N_{eff,n} = \left[\sum_{i=1}^{M} \left(W_n^i \right)^2 \right]^{-1} \qquad (23)$$

as a measure that reflects the extent of uniformity of the normalized weights W_n^i, computed from w_n^i during the n-th step of the filtering procedure. The quantity $N_{eff,n}$ satisfies $1 \leq N_{eff,n} \leq M$ with lower bound 1 indicating extreme weight degeneration: all probability is concentrated on one particle only with $W_n^i = 1$ for that particle. The upper bound M indicates that the weights are all equal to $1/M$, the ideal case. A filtering procedure that outputs particles from the true filtered density at each time step will have a constant value of $N_{eff} = M$ in the ideal case. Deviations from this ideal case indicate the extent of weight/particle degeneration of the filtering procedure.

6 Experimental results

Two test examples, one 2D and one 3D example used in [33], are given in this section to illustrate the performance of the TS procedure. This section compares the performance of three estimators: the O_2, SIR, and TS, based on their RMSE values. We also study the robustness of the TS procedure under various noise levels.

6.1 Example 1

The ordinary differential equation (ODE) model in [33] given by

$$\dot{x}_1 = -x_1/2, \qquad \dot{x}_2 = \sin(x_2) \qquad (24)$$

is considered, where $x = (x_1, x_2) \in R^2$ is the state space variable. The state space model corresponding to the time-discretized version of (24) is

$$x_{n,1} = x_{n-1,1} - \Delta t (x_{n-1,1}/2) + u_{n,1}, \qquad (25)$$

$$x_{n,2} = x_{n-1,2} - \Delta t \sin (x_{n-1,2}) + u_{n,2} \qquad (26)$$

where $u_n = (u_{n,1}, u_{n,2})^T \sim \mathcal{N}(\mathbf{0}, \tilde{Q}_n)$,

$$\tilde{Q}_n \equiv Q_n \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} \qquad (27)$$

with $Q_n \equiv (\Delta t) q_n$. Equations (25) and (26) are a special case of the state space model in (1). The measurement model considered is linear:

$$y_n = x_n + v_n \qquad (28)$$

where $v_n \sim \mathcal{N}\left(0, \tilde{R}_n\right)$ with $\tilde{R}_n \equiv R_n I_{2\times 2}$. The numbers R_n and Q_n govern the extent of noise in the measurement and state space models, respectively. We note that the covariance matrix of the measurement model is given by $\tilde{R}_n \equiv R_n I_{2\times 2}$ whereas covariance matrix of the state space model is given by $\tilde{Q}_n$ which is related to Q_n (and q_n) as in (27). In our experiments, constant values of $Q_n = Q$ and $R_n = R$ are considered.

The prior on x_0 is taken as

$$p_0(x_0) = \phi_2 \left(x; \begin{bmatrix} -12 \\ 0 \end{bmatrix}, Q \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} \right). \qquad (29)$$

We set $\Delta t = 0.02$ and $T = 10$ as the final time point. We carried out $N = 100$ simulation experiments based on various specifications of R and Q. These specifications are reported in Tables 4 and 5. The x-trajectories were generated from the prior distribution (29), followed by the state space transition kernel given by Eqs. (25) and (26). Given $x_{0:T}$, the observations $y_{1:T}$ were generated from the measurement model (28). For the TS procedure, the EnKF kernel based importance distribution is constructed using 50 ensemble particles. The GMMs are fitted using the maximum setting of $G_0 = 10$ components.

To illustrate how the trajectories of the true state space variables look like in Example 1, we provide trajectory plots corresponding to $R = 0.1$ and $Q = 0.2$ in Fig. 1. Figure 1 gives the trajectory plots of $x_{n,1}$ and $x_{n,2}$ (true values) together with their estimates based on the TS, SIR, and O_2 procedures. The average of RMSE over $N = 100$ experiments for the three procedures is shown in Fig. 2 for the noise level specifications $R = 0.1$ and $Q = 0.2$. The RMSE of all R and Q combinations are reported in Tables 4 and 5.

Rehman *et al. EURASIP Journal on Advances in Signal Processing* (2018) 2018:36

Page 9 of 19

Table 4 Simulation results for Example 1: Mean and standard deviations of the RMSE for three different estimators of $x_{n,1}$, namely, O_2, the posterior mean based on SIR and the posterior mean based on TS

Exp. no.	(Q, R)	$\frac{R}{Q}$	RMSE for $x_{n,1}$					
			O_2		SIR		TS	
			Mean	SD	Mean	SD	Mean	SD
1	(0.1, 0.1)	1.0	0.0977	0.0093	0.1260	0.0728	0.0696	0.0238
2	(0.2, 0.1)	0.5	0.1048	0.0128	0.2225	0.1305	0.0665	0.0075
3	(0.4, 0.1)	0.25	0.0930	0.0126	0.4081	0.2352	0.0815	0.0212
4	(0.8, 0.1)	0.125	0.0985	0.0123	0.6030	0.3530	0.0872	0.0327
5	(1.0, 0.1)	0.100	0.0974	0.0127	0.6848	0.5169	0.1190	0.0538
6	(0.1, 1.0)	10.0	1.0290	0.1465	0.4606	0.2287	0.3164	0.1152
7	(0.4, 1.0)	2.5	0.9933	0.1684	0.6965	0.3935	0.4716	0.1602

We summarize our findings based on Tables 4 and 5 for Example 1. When R is small (i.e., the value 0.1 in Experiments 1–5 in Tables 4 and 5) and Q is small (for example, Exp. 1 and 2 in Tables 4 and 5), the filtering procedures give better performance compared to O_2. In this situation, the state space model is informative and filtering adds value to the final goal of estimating x_n. The RMSE values of the O_2 estimator are slightly higher compared to SIR and TS, and the RMSE of TS is the lowest (best), e.g., see Exp. 1 and 2 in Tables 4 and 5. However, when Q increases, deterioration of filtering due to particle degeneracy is apparent, as evidenced by the larger RMSE (mean as well as standard deviation). The increase in RMSE is more pronounced for the SIR filter compared to TS, which indicates that TS produces robust estimates of x_n compared to SIR even when Q becomes larger. The performance of the O_2 estimator is not affected by the increase in Q and remains robust throughout. When R is small, the RMSE of the O_2 estimator is only slightly larger compared to the optimal RMSE of the posterior mean obtained by TS.

When R is large (i.e., the value 1 in Experiments 6 and 7 in Tables 4 and 5) and Q is small, the filtering procedures give significantly better performance compared to O_2. In this situation, the state space model is more informative
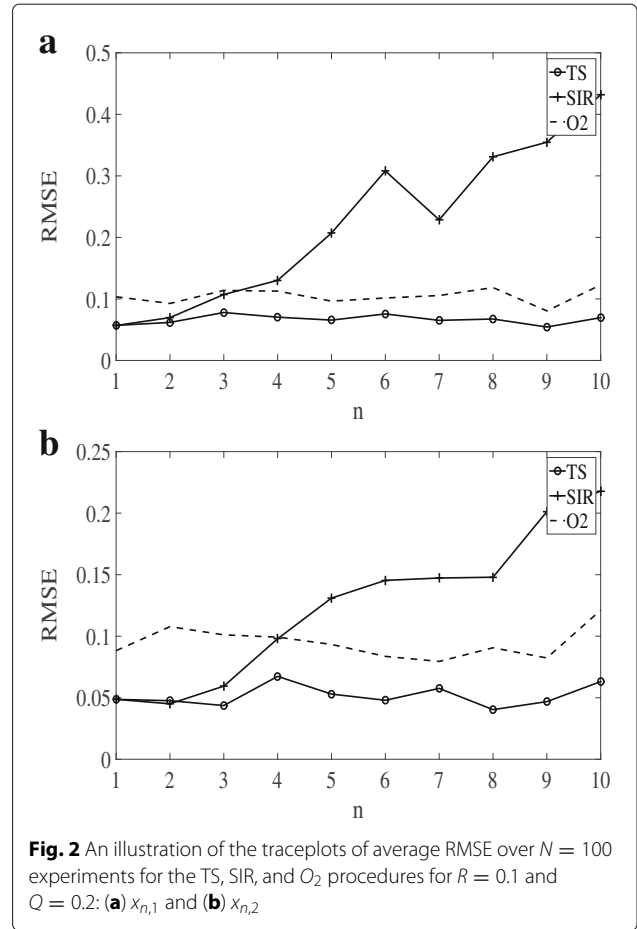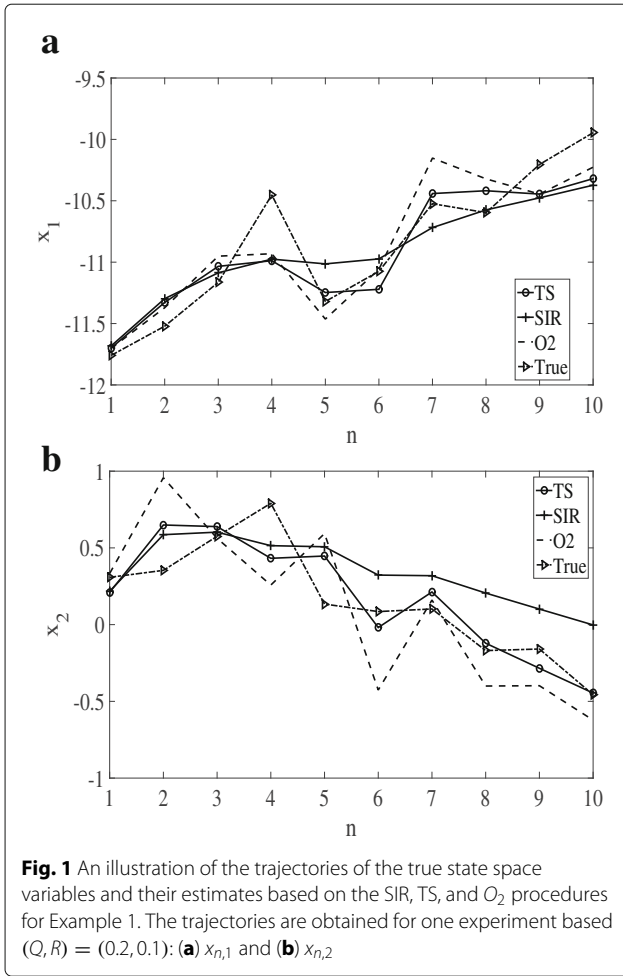
compared to the measurement model, and filtering adds significant value to the estimation of x_n. The RMSE values of the O_2 estimator are much higher compared to SIR and TS. However, as in the previous case, deterioration of the filtering due to particle degeneracy is apparent as Q increases and it is more pronounced for the SIR filter. TS produces robust estimates for x_n when Q becomes larger compared to SIR. The performance of the O_2 estimator is not affected by the increase in Q and remains robust throughout, but the RMSE of O_2 is significantly higher compared to TS when R is large.

Contour plots are obtained for the 95% HPD sets corresponding to $x_{n,1}$ and $x_{n,2}$; these are given in Fig. 3 for $(Q, R) = (0.2, 0.1) = 0.2$, for time steps $n = 5$, $n = 7$, and $n = 10$, and for three randomly selected experiments 25, 49, and 99 out of the $N = 100$. It can be observed from Fig. 3 that the true simulated values of $x_{n,1}$ and $x_{n,2}$ belong to their respective HPD sets in all the panels.

Next, we calculated the average coverage probabilities of the HPD sets which are given in Table 6. The coverage probabilities are averaged over $N = 100$ experiments, reported for each $n = 1, 2, \cdots, 10$ and for $(Q, R) = (0.2, 0.1)$. The high values of coverage probabilities show that the filtering performance of the TS algorithm is very

Table 5 Simulation results for Example 1: Mean and standard deviations of the RMSE for three different estimators of $x_{n,2}$, namely, O_2, the posterior mean based on SIR and the posterior mean based on TS

Exp. no.	(Q, R)	R/Q	RMSE for $x_{n,2}$					
			O_2		SIR		TS	
			Mean	SD	Mean	SD	Mean	SD
1	(0.1, 0.1)	1.0	0.0986	0.0088	0.0646	0.0315	0.0395	0.0089
2	(0.2, 0.1)	0.5	0.0947	0.0129	0.1242	0.0607	0.0516	0.0086
3	(0.4, 0.1)	0.25	0.1069	0.0168	0.1781	0.0750	0.0689	0.0107
4	(0.8, 0.1)	0.125	0.1009	0.0138	0.4562	0.3604	0.0819	0.0182
5	(1.0, 0.1)	0.100	0.1030	0.0114	0.5090	0.3914	0.1136	0.0500
6	(0.1, 1.0)	10.0	0.9943	0.0934	0.2452	0.1371	0.1920	0.0753
7	(0.4, 1.0)	2.5	0.9926	0.1125	0.4424	0.1824	0.2905	0.0888

Fig. 1 An illustration of the trajectories of the true state space variables and their estimates based on the SIR, TS, and O_2 procedures for Example 1. The trajectories are obtained for one experiment based on $(Q, R) = (0.2, 0.1)$: (**a**) $x_{n,1}$ and (**b**) $x_{n,2}$

Fig. 2 An illustration of the traceplots of average RMSE over $N = 100$ experiments for the TS, SIR, and O_2 procedures for $R = 0.1$ and $Q = 0.2$: (**a**) $x_{n,1}$ and (**b**) $x_{n,2}$

effective; coverage deterioration is not seen with the propagation of time. In the same way, we obtained coverage probabilities for all pairs of (Q, R) considered earlier. We found that coverage probabilities were similar to the ones reported in Table 6, thus demonstrating the robustness of the TS procedure under various noise conditions.

Figure 4 shows the quality of the GMM approximation to frequency histograms for a selected experiment; the frequency histograms are generated from resamples of $\left\{ x_n^i, w_n^i \right\}_{i=1}^M$ which are representative of the true filtered density. However, the resamples are subject to sampling variability. Note that the GMM curve fit (represented by the solid line) is a good fit to all the frequency histograms and does not suffer from resampling variability. The goodness of fit of the GMM and absence of resampling variability is the reason why weights of the TS procedure are more uniform and less prone to degeneracy.

To study the advantage of the ML estimation scheme, we report N_{eff} for the $LLPF_{EnKF}$ and the TS procedures in Fig. 5 for four different combinations of R and Q. The
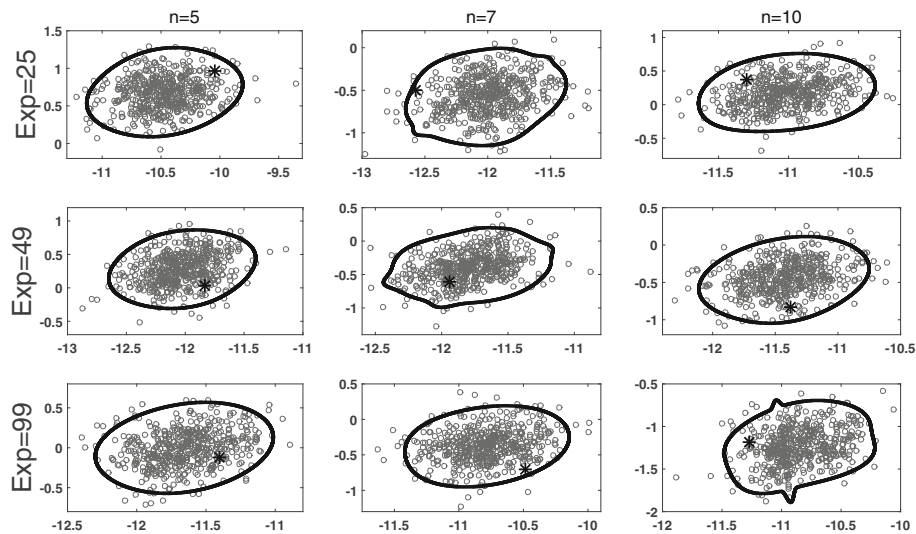
$LLPF_{EnKF}$ procedure is the first stage of the TS procedure without the ML estimation scheme. We note that the decrease in N_{eff} is much lesser for the *TS* procedure, indicating more uniform weights for all the noise levels considered. Thus, the TS procedure is robust in the sense that its weights are more uniform (hence, less skewed) compared to $LLPF_{EnKF}$ under various noise conditions.

6.2 Example 2
The well known Lorenz 63 model is proposed as an application for the TS methodology. The Lorenz model exhibits strong non-linearity and chaos [9] and is considered as a benchmark example in data assimilation problems for testing the effectiveness of filtering methodology [11]. The Lorenz 63 model is a 3D model describing atmospheric convection based on the following ODEs: $\dot{x}_1 = \alpha(-x_1 + x_2)$, $\dot{x}_2 = \beta x_1 - x_2 - x_1 x_3$, and $\dot{x}_3 = -\gamma x_3 + x_1 x_2$, where $\alpha = 10$, $\beta = 28$, and $\gamma = 8/3$. The dynamical system corresponding to the time-discretized version of the continuous Lorentz 63 model is

$$x_{n,1} = x_{n-1,1} + \Delta t\, \alpha(-x_{n-1,1} + x_{n-1,2}) + u_{n,1} \qquad (30)$$

$$x_{n,2} = x_{n-1,2} + \Delta t\left[\beta x_{n-1,1} - x_{n-1,2} - x_{n-1,1} x_{n-1,3}\right] + u_{n,2} \quad (31)$$

Rehman *et al. EURASIP Journal on Advances in Signal Processing* (2018) 2018:36

Page 11 of 19

Fig. 3 95% HPD contour sets based on the fitted GMM, for the experiments $E = 25, 49$, and 99. The true simulated value of $x_n \equiv (x_{n,1}, x_{n,2})$ is marked by '×' in each panel

$$x_{n,3} = x_{n-1,3} + \Delta t \left[-\gamma x_{n-1,3} + x_{n-1,1} x_{n-1,2} \right] + u_{n,3} \quad (32)$$

where $u_n = (u_{n,1}, u_{n,2}, u_{n,3})^T \sim \mathcal{N}(\mathbf{0}, \tilde{Q}_n)$ with $\tilde{Q}_n \equiv Q_n I_{3\times3} \equiv (\Delta t) q_n I_{3\times3}$. The measurement model considered is linear:

$$y_n = x_n + v_n \quad (33)$$

where $v_n \sim \mathcal{N}(0, \tilde{R}_n)$ with $\tilde{R}_n \equiv R_n I_{3\times3}$. As in Example 1, we consider constant values $R_n = R$ and $Q_n = Q$ to govern the extent of noise in the measurement and state space models, respectively.

The prior on x_0 is taken as normal

$$p_0(x_0) = \phi_2 \left(x; \begin{bmatrix} -0.2 \\ -0.2 \\ 8 \end{bmatrix}, Q \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \right). \quad (34)$$

As in Example 1, we carried out $N = 100$ simulation experiments based on various specifications of R and Q. These specifications are reported in Tables 7, 8, and 9. The x-trajectories were generated from the prior distribution (34) followed by the state space transition kernel given by

Eqs. (30)–(32). Given $x_{0:T}$, the observations $y_{1:T}$ were generated from the measurement model (33). We set $\Delta t = 0.02$ and $T = 10$ as the final time point. As before for the TS procedure, the EnKF kernel-based importance distribution is constructed using 50 ensemble particles. The GMMs are fitted using the maximum setting of $G_0 = 10$ components.
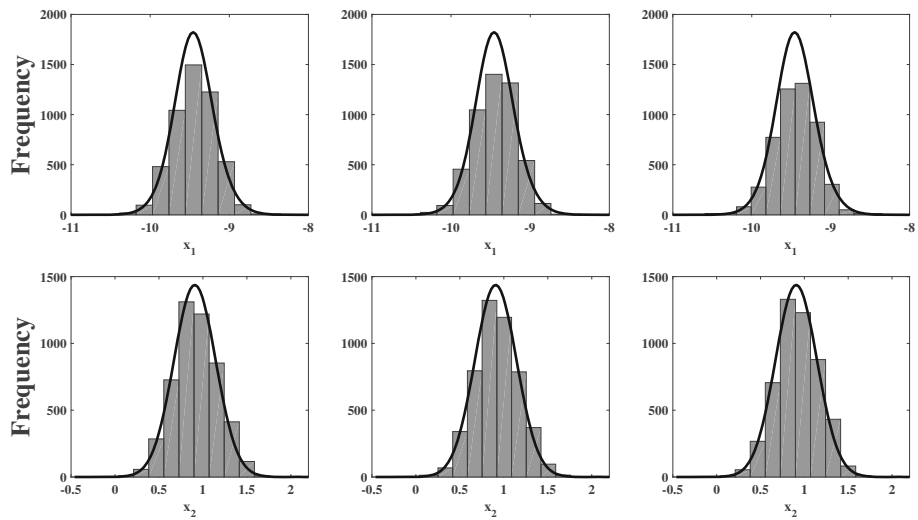
Figure 6 gives the average RMSE plots over $N = 100$ experiments for comparing the TS, SIR, and O_2 procedures based on $(Q, R) = (0.5, 1)$. The RMSE corresponding to all (Q, R) specifications are reported in Tables 7, 8, and 9.

We obtain similar findings for Example 2. When R is small (i.e., the value 0.1 in Experiments 1–5 in Tables 7, 8, and 9) and Q is small, the filtering procedures that yield the posterior mean as the estimate of x_n give better performance compared to O_2. The RMSE values of the O_2 estimator are slightly higher compared to SIR and TS (e.g., Exp. 1 and 2 in all the tables). However, when Q increases, deterioration of filtering due to particle degeneracy is apparent, as evidenced by the larger RMSE (mean as well

Table 6 Average coverage probabilities (in %) of the 95% HPD confidence set for $(x_{n,1}, x_{n,2})$ based on the TS procedure

Time step n	Coverage (%) by 95% HPD sets	Time step n	Coverage (%) by 95% HPD sets
1	97.6	6	92.6
2	94.6	7	93.4
3	95.6	8	94.4
4	95.2	9	95.6
5	94.4	10	93.8

Coverage probabilities are shown for each n in Example 1 corresponding to $(Q, R) = (0.2, 0.1)$

Rehman *et al. EURASIP Journal on Advances in Signal Processing* (2018) 2018:36

Page 12 of 19

Fig. 4 Univariate frequency histograms of $x_{n,1}$ (top row) and $x_{n,2}$ (bottom row) in Example 1 in the case of filtering at $n = T$. The non-normality (asymmetries) of the frequency histograms is well approximated by the fitted GMM (solid line). The histograms are constructed using resamples from $\{x_T^i, w_T^i\}_{i=1}^M$

as standard deviation). The increase in RMSE (both mean and standard deviation) is more pronounced for the SIR filter compared to TS, which indicates that TS produces robust estimates of x_n even when Q becomes larger.

When R is large (i.e., the value 1 in Experiments 6–9 in Tables 7, 8, and 9) and Q is small, the filtering procedures give superior performance compared to O_2. In

this situation, the state space model is more informative compared to the measurement model, and filtering adds significant value to the estimation of x_n. The RMSE values of the O_2 estimator are much higher compared to SIR and TS. However, as in the previous case, deterioration of the filtering due to particle degeneracy is apparent as Q increases and it is more pronounced for the SIR filter. TS
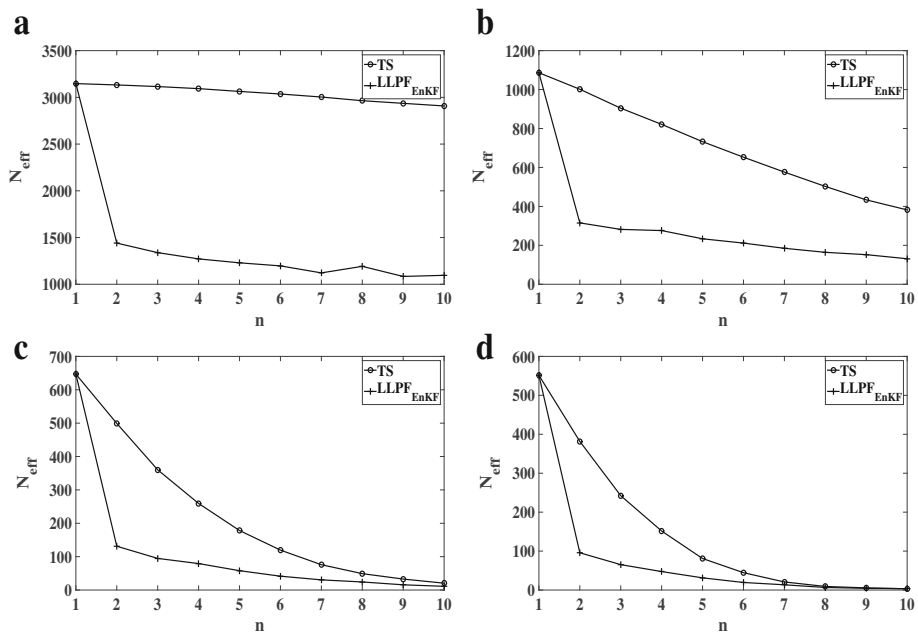
Fig. 5 Traceplots of average $N_{eff,n}$ of the TS and $LLPF_{EnKF}$ procedures based on $N = 100$ simulation experiments for Example 1 corresponding to four different (Q, R) specifications: **a** $(Q, R) = (0.4, 1)$, **b** $(Q, R) = (0.4, 0.1)$, **c** $(Q, R) = (0.8, 0.1)$, and **d** $(Q, R) = (1, 0.1)$

Rehman *et al. EURASIP Journal on Advances in Signal Processing* (2018) 2018:36

Page 13 of 19

Table 7 Simulation results for Example 2: Mean and standard deviations of the RMSE for three different estimators of $x_{n,1}$, namely, O_2, the posterior mean based on SIR and the posterior mean based on TS

Exp. no.	(Q, R)	R/Q	RMSE for $x_{n,1}$					
			O_2		SIR		TS	
			Mean	SD	Mean	SD	Mean	SD
1	$(0.05, 0.1)$	2.0	0.0975	0.0092	0.0731	0.0247	0.0430	0.0046
2	$(0.1, 0.1)$	1.0	0.0975	0.0092	0.1316	0.0562	0.0600	0.0054
3	$(0.2, 0.1)$	0.5	0.0975	0.0092	0.2421	0.1286	0.1057	0.0354
4	$(0.5, 0.1)$	0.2	0.0975	0.0092	0.5258	0.3407	0.2066	0.0958
5	$(1.0, 0.1)$	0.1	0.0995	0.0118	1.0438	0.6207	0.2796	0.1236
6	$(0.1, 1.0)$	10.0	1.0819	0.1884	0.1991	0.0474	0.1832	0.0432
7	$(0.2, 1.0)$	5.0	1.0171	0.1087	0.3268	0.0854	0.2598	0.0376
8	$(0.5, 1.0)$	2.0	1.0537	0.1222	1.2481	0.5182	0.5583	0.1881
9	$(1.0, 1.0)$	1.0	1.0000	0.2104	1.3098	0.4975	0.5926	0.1279

produces robust estimates for x_n when Q becomes larger compared to SIR. The performance of the O_2 estimator is not affected by the increase in Q and remains robust throughout, but the RMSE of O_2 is significantly higher compared to TS when R is large making it a sub-optimal estimator in this case.

We do not provide visual plots of the HPD sets for $x_n = (x_{n,1}, x_{n,2}, x_{n,3})$ since they are 3D sets. Nevertheless, the methodology outlined in the Additional file 1: Section 1 of the paper is able to compute the thresholds $\tilde{\kappa}_\alpha$ and coverage probabilities of HPD sets in 3D and, in fact, for any dimension.

The average coverage probabilities of the HPD sets over $N = 100$ experiments are given in Table 10 for each $n = 1, 2, \cdots, 10$, and for $(Q, R) = (0.5, 1)$. The high values of coverage probabilities show that the filtering performance of the TS algorithm is very effective; coverage deterioration is not seen with the propagation of time. In the same way, we obtained coverage probabilities for all

pairs of (Q, R) considered earlier. We found that coverage probabilities were similar to the ones reported in Table 10, thus demonstrating the robustness of the TS procedure under various noise conditions.

Figure 7 shows the quality of the GMM approximation to frequency histograms for a selected experiment; the frequency histograms generated from resamples of $\{x_n^i, w_n^i\}_{i=1}^M$ are representative of the true filtered density. However, the resamples are subject to sampling variability. Note that the GMM curve fit (represented by the solid line) is a good fit to all the frequency histograms and does not suffer from resampling variability. The goodness of fit of the GMM and the absence of resampling variability are the reasons why weights of the TS procedure are more uniform and less prone to degeneracy.

We also report N_{eff} for the $LLPF_{EnKF}$ and the TS procedures in Fig. 8 for four different combinations of R and Q. We note that the decrease in N_{eff} is lesser for the proposed TS procedure indicating more

Table 8 Simulation results for Example 2: Mean and standard deviations of the RMSE for three different estimators of $x_{n,2}$, namely, O_2, the posterior mean based on SIR and the posterior mean based on TS

Exp. no.	(Q, R)	R/Q	RMSE for $x_{n,2}$					
			O_2		SIR		TS	
			Mean	SD	Mean	SD	Mean	SD
1	$(0.05, 0.1)$	2.0	0.0988	0.0189	0.0745	0.0284	0.0500	0.0050
2	$(0.1, 0.1)$	1.0	0.0988	0.0189	0.1288	0.0659	0.0654	0.0088
3	$(0.2, 0.1)$	0.5	0.0988	0.0189	0.2423	0.1643	0.1146	0.0499
4	$(0.5, 0.1)$	0.2	0.0988	0.0189	0.6131	0.4831	0.2293	0.1181
5	$(1.0, 0.1)$	0.1	0.1007	0.0150	1.5960	1.4733	0.2881	0.1205
6	$(0.1, 1.0)$	10.0	0.9569	0.1735	0.3094	0.1273	0.3027	0.1210
7	$(0.2, 1.0)$	5.0	0.9910	0.1608	0.3606	0.0862	0.3440	0.0568
8	$(0.5, 1.0)$	2.0	1.0473	0.1379	1.4103	0.7227	0.7169	0.1229
9	$(1.0, 1.0)$	1.0	0.9946	0.1457	1.2677	0.6478	0.6411	0.1119

Rehman *et al. EURASIP Journal on Advances in Signal Processing* (2018) 2018:36

Page 14 of 19

Table 9 Simulation results for Example 2: Mean and standard deviations of the RMSE for three different estimators of $x_{n,3}$, namely, O_2, the posterior mean based on SIR and the posterior mean based on TS

Exp. no.	(Q, R)	R/Q	RMSE for $x_{n,3}$					
			O_2		SIR		TS	
			Mean	SD	Mean	SD	Mean	SD
1	$(0.05, 0.1)$	2.0	0.1007	0.0119	0.1190	0.0406	0.0567	0.0132
2	$(0.1, 0.1)$	1.0	0.1007	0.0119	0.1800	0.0944	0.0696	0.0103
3	$(0.2, 0.1)$	0.5	0.1007	0.0119	0.2976	0.1850	0.1137	0.0456
4	$(0.5, 0.1)$	0.2	0.1007	0.0119	0.6206	0.4280	0.2268	0.1089
5	$(1.0, 0.1)$	0.1	0.0944	0.0145	1.0841	0.7334	0.2652	0.0931
6	$(0.1, 1.0)$	10.0	0.9370	0.1374	0.2932	0.0616	0.2062	0.0188
7	$(0.2, 1.0)$	5.0	0.9515	0.2547	0.5539	0.1642	0.3280	0.0606
8	$(0.5, 1.0)$	2.0	1.0107	0.1934	1.5100	0.8464	0.6677	0.0613
9	$(1.0, 1.0)$	1.0	0.9635	0.0891	1.3989	0.7751	0.5752	0.0739

uniform weights for all the noise levels considered as in Example 1.

The distribution of weights prior to ML estimation is transformed into a new set of weights based on the GMM fitted density in the second stage of the TS procedure. The

Fig. 6 An illustration of the traceplots of average RMSE over $N = 100$ experiments for the TS, SIR, O_2 procedures based on $(Q, R) = (0.5, 1)$: **a** $x_{n,1}$, **b** $x_{n,2}$, and **c** $x_{n,3}$

ML estimation scheme ensures that the (second stage) weights are more uniformly distributed and that the (second stage) particles and weights provide a good approximation to the filtered density. We have provided some insight into why the ML estimation scheme is able to do so in Additional file 1: Section 2.2. By making the weights more uniform, the ML estimation scheme ensures that weight degeneracy is mitigated.
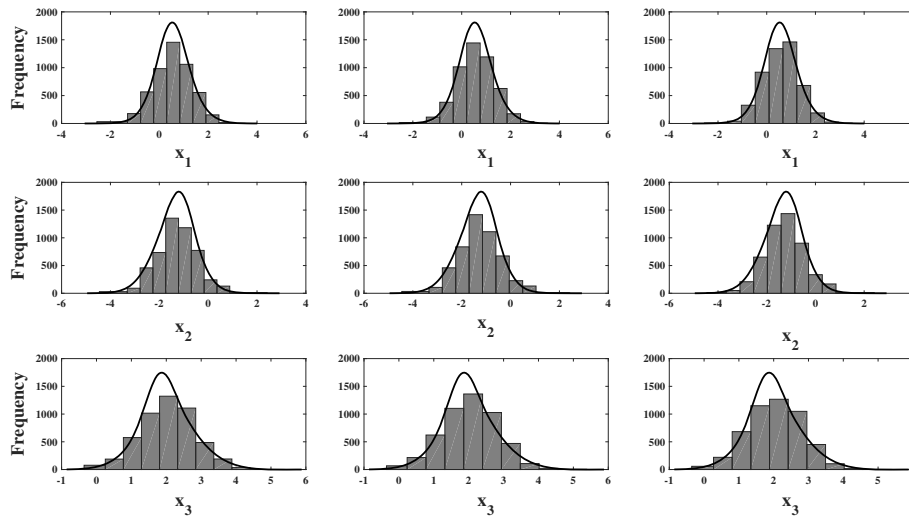
Based on the N_{eff} plots for Examples 1 and 2, as given by Figs. 5 and 8, we note that N_{eff} decreases at a much slower rate for TS compared to $LLPF_{EnKF}$ for all the noise specifications considered. Thus, the ML estimation scheme yields weights that are more uniform and less prone to degeneracy under all noise levels, thus making the TS procedure more robust.

The details of computational times for the three estimators are as follows: The average processing time of the TS procedure is 8–12 s per time step, whereas the average processing times of SIR and O_2 are 0.0019 and 1.42×10^{-7} s, respectively. TS has the largest computational time but gives superior performance: TS has lower RMSE values compared to O_2 for small R and significantly lower
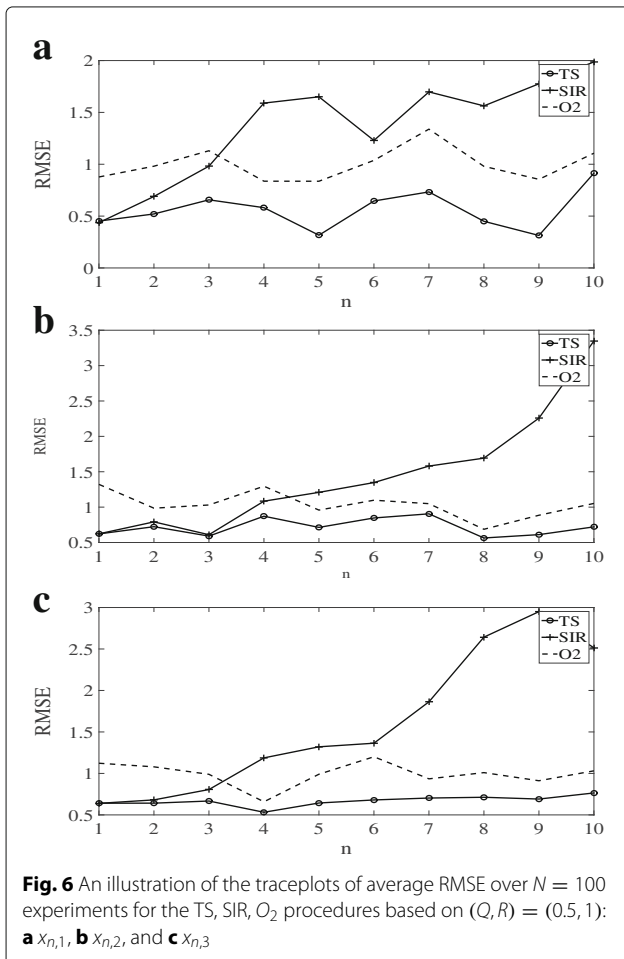
Table 10 Coverage probabilities (expressed as percentage) of the 95% HPD confidence set based on the filtered densities at each time point n in Example 2 (the Lorentz 63 model) with $Q = 0.5$ and $R = 1$

Time step	Coverage (%)	Time step	Coverage (%)
n	by 95% HPD sets	n	by 95% HPD sets
1	96.6	6	93.4
2	95.8	7	90.8
3	93.4	8	90.4
4	95.4	9	93.6
5	94.0	10	94.6

Rehman *et al. EURASIP Journal on Advances in Signal Processing* (2018) 2018:36

Page 15 of 19

Fig. 7 Univariate frequency histograms of $x_{n,1}$ (top row) and $x_{n,2}$ (bottom row) for the Lorentz 63 model in the case of filtering at $n = T$. The non-normality (asymmetries) of the frequency histograms is well approximated by the fitted GMM (solid line). The histograms are constructed using resamples from $\{x_T^i, w_T^i\}_{i=1}^M$

RMSE compared to O_2 for large R when particle degeneracy is negligible. TS remains more robust compared to SIR when filtering is subject to various noise conditions. Computational times are based on running the experiments on a DELL Precision T1700 workstation with Xeon E3-1226 v3 processor with processing speed 3.3 GHz and 16 GB of RAM. Codes were developed in MATLAB 2015a.

7 Discussion
To summarize, the TS procedure produces robust estimates of the underlying state space variable under various noise conditions. The value of filtering and in particular TS is best realized when the noise of the state space model is small at every fixed level of the measurement noise. When the measurement noise level is small, TS procedures are slightly better estimates compared to O_2,
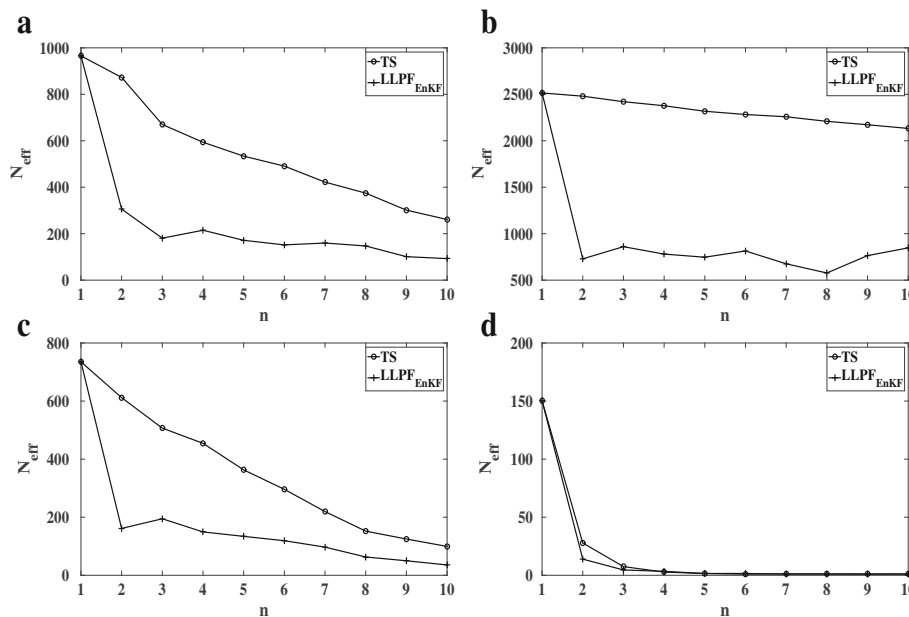
Fig. 8 Traceplots of average $N_{eff,n}$ of the TS and $LLPF_{EnKF}$ procedures based on $N = 100$ simulation experiments for Example 2 corresponding to four different (Q, R) specifications: **a** $(Q, R) = (1, 1)$, **b** $(Q, R) = (0.2, 1)$, **c** $(Q, R) = (0.1, 0.1)$, and **d** $(Q, R) = (0.05, 0.1)$

Rehman *et al. EURASIP Journal on Advances in Signal Processing* (2018) 2018:36

Page 16 of 19

whose performance is of comparable level. But when R is large, TS gives significantly better performance compared to O_2. The best performance of the TS procedure is observed when R is large and Q is small. For these values of R and Q, the effect of particle degeneracy is the smallest. For higher levels of Q, particle degeneracy becomes more pronounced and the posterior mean of the filtered density becomes a poorer estimate of x_n. Nevertheless, the TS procedure is less affected by particle degeneracy due to the implementation of ML estimation in the second stage. The smaller RMSE values of the TS shows that the ML scheme works to reduce particle degeneracy. The distribution of weights prior to the ML estimation is made to be more uniform by transforming to a new set of weights obtained by fitting the GMM.

The performance of the O_2 estimator is not affected by the increase in Q and remains robust throughout. When R is small, the RMSE of the O_2 estimator is only slightly larger compared to the optimal RMSE of the posterior mean obtained by TS, but when R is large, the O_2 estimator has significantly larger RMSE compared to the posterior mean obtained using the TS procedure.

The "value" of filtering can be seen in the situations when R/Q becomes larger for every fixed value of R. This is consistent with the findings of [23] where filtering was deemed effective when

$$r \equiv \delta_P^2/\delta_0^2 \qquad (35)$$

was small and

$$p \equiv \frac{m_P - m_0}{\delta_0^2} \qquad (36)$$

was close to zero; see Eqs. (16) and (17) of [23]. In the above, δ_P^2 and δ_0^2, respectively, denote the variances of the state space and measurement models, whereas m_P and m_0, respectively, denote the means of the state space and measurement models. In their paper [23], the authors show that filtering is deemed effective when PoFB (probability of filter benefit) is above 0.5. Based on the funnel-shaped region of Fig. 3 in their paper, they show that when r and p are small, the PoFB, indeed, lies above 0.5. This is because the PoFB lies along the upper boundary of the funnel-shaped diagram which is significantly above the constant line of 0.5.

To study the impact of the above results in our context, we note that the observation model is always unbiased in our experiments, that is, $m_0^i = x_n^i$ where i is the index of the i-th experiment, $i = 1, 2, \cdots, N$, and x_n^i is the true simulated but unknown value of the state space variable at time step n. Note that Q and R are proxies of δ_P^2 and δ_0^2, so that R/Q largely corresponds to the case where r is small.

For the i-th experiment,

$$p^i = \frac{m_P^i - x_n^i}{\delta_0^2}$$

where $m_P^i = \Phi_n(x_{n-1}^i)$ (see (1)). It follows that ave (p^i) over all experiments is

$$\text{ave}(p^i) = \text{ave}\left(\frac{m_P^i - x_n^i}{\delta_0}\right) = \frac{\text{ave}(m_P^i - x_n^i)}{\delta_0} = 0.$$

Furthermore, the variance of p^i,

$$\text{var}(p^i) = \text{var}\left(\frac{m_P^i - x_n^i}{\delta_0}\right) = \frac{\text{var}(m_P^i - x_n^i)}{\delta_0^2} = \frac{\delta_P^2}{\delta_0^2} = r.$$

Thus, when R/Q is large, r is small, and hence, var (p^i) is also small. It follows that the values of p^i will be highly concentrated around its mean value $p = 0$ which forms the upper boundary of the funnel-shaped region of PoFB values in Fig. 3 of [23]. Thus, the PoFB values from the experiments will almost all be significantly higher than 0.5, indicating that filtering is very effective in almost all of the N experiments. When R/Q is small, r is large, and hence, $var(p^i)$ is large but the mean value remains the same at $p = 0$. The PoFB now can vary from experiment to experiment and can be significantly below 0.5 indicating that the filtering is not so effective. This phenomenon is exemplified by the SIR, but our TS procedure still remains robust even in such situations.

8 Conclusions

We propose and develop a two-step (TS) particle filtering procedure to select an optimal importance density and reduce weight degeneracy in particle filtering. The TS procedure is based on fitting Gaussian mixture models to a set of particles and weights via weighted likelihood. It is shown that particle weights from the TS procedure do not deteriorate significantly over time compared to other PF methods considered. In the subsequent paragraphs, we provide theoretical comparisons between the TS and other filtering procedures (and their extensions) to gauge the scope of the TS procedure in a variety of applications.

Pure filtering refers to the situation where parameters of the dynamical system are completely fixed and known. Currently, the TS procedure can be applied to pure filtering problems only. In future work, we plan to extend the TS procedure to perform parameter inference in dynamical systems. For parameter inference, there is always an associated pure filtering procedure in addition to estimating the unknown parameters; see, for example, [20]. We will investigate properties of the parameter inference algorithm when the TS procedure is used as the underlying filtering algorithm.

Rao-Blackwellized filters are filters based on conditionally Gaussian models [29]. The TS procedure currently

Rehman *et al. EURASIP Journal on Advances in Signal Processing* (2018) 2018:36

Page 17 of 19

does not incorporate any such conditioning but can be developed to be a part of such filters where it can be used for the non-linear and non-Gaussian components. The TS procedure is robust, that is, it is not severely affected by significant measurement noise levels. Thus, the TS procedure can be compared with other procedures in terms of the extent of their robustness for filtering. In [20], the unscented Kalman filter (UKF) is shown to be robust for online model assessment and parameter estimation. Thus, a comparison can be made when the UKF is replaced by the TS procedure for online model assessment and parameter inference; see also [3, 38, 45].

Spatial applications involve state space vectors that are high dimensional and where a priori dependencies exist between vector components that are spatially close to each other. The TS methodology can be extended to such high-dimensional situations by eliciting the class of GMM models (in STEP 2 of the TS procedure) to capture these inherent spatial dependencies. Thus, the TS approach can be seen as a promising approach for higher dimensional problems involving spatial as well as more general dependencies.

In tracking applications, the probability hypothesis density (PHD) filter is used to track multiple objects [42]. In [32], for example, the PHD filter is proposed to track the branches from centerlines of neurons. Methodology for finding the PHD filter based on the TS procedure can also be developed based on a modified weighted likelihood criteria for random sets.

If the measurement model is highly complex and non-linear, the likelihood component in filtering is not available in closed (i.e., tractable) form. The evaluation of the likelihood becomes comparatively difficult. In such situations, the method proposed in [24] can be used to calculate the likelihood numerically. The weights calculated in STEP 2 of the TS procedure involve likelihood computations, and when the likelihood is not available in closed form, the TS procedure can benefit from the numerical methods reported in [24] for evaluating the likelihood. In a similar manner, different approaches to filtering (see [44]) can be used in combination with the TS procedure for improved filtering results.

An important contribution of estimation via weighted likelihood, as proposed in this paper, is that the unknown filtered density $f_n(x)$ can now be obtained based on statistical density estimation techniques. Incorporation of estimation techniques in filtering opens up the possibility of incorporating many other statistical methods of modeling and inference into the area of filtering and, later on, for the development of tracking, monitoring, and early warning applications. Statistical models will be used to elicit a priori structures of the state space vector in dynamical systems, while the estimation scheme will select the best

statistical model that adheres to this structure based on particles and weights. The spatial and tracking scenarios mentioned earlier serve as illustrations of potential applications of this general statistical framework to filtering.

Additional file

Additional file 1: Supplementary Material: A weighted likelihood criteria for learning importance densities in particle filtering. (PDF 183 kb)

Abbreviations
APF: Auxiliary particle filter; BIC: Bayes information criteria; EnKF: Ensemble Kalman filter; EKF: Extended Kalman filter; EnKPF: Ensemble Kalman particle filter; EM: Expectation maximization; GMM: Gaussian mixture models; HPD: Highest posterior density; IPF: Improved particle filter; KF: Kalman filter; LLPF: Linear localized particle filter; MCMC: Markov chain Monte Carlo; PF: Particle filter; SIR: Sequential importance resampling; SIS: Sequential importance sampling; TSPF: Two-step particle filter; UKF: Unscented Kalman filter; UPF: Unscented particle filter

Availability of data and materials
Supplementary data for detailed calculations are available in separate file.

Authors' contributions
SCD and VSA conceived of the presented ideas. MJR developed the theory, extended the ideas, and performed the computations. All authors verified the analytical methods and results. SCD and VSA supervised the work and its findings. All authors discussed the results, provided critical feedback, and helped shape the final manuscript. All authors read and approved the final manuscript.

Authors' information
M. Javvad ur Rehman received the master's degree from Quaid-e-Azam University, Islamabad, Pakistan, in 2010. He is currently pursuing his Ph.D. with the Universiti Teknologi PETRONAS, Malaysia. He is serving as a lecturer in the Engineering Department, National University of Modern Languages, Islamabad, Pakistan. His research interests include signal processing, parameter estimation, Bayesian computation, filtering, and smoothing. Sarat C. Dass is currently an associate professor in the Department of Fundamental and Applied Sciences, Universiti Teknologi PETRONAS. Previously he was in the Department of Statistics and Probability at Michigan State University, USA, conducting teaching and research in Statistics. He received the M.Sc. and Ph.D. degrees in Statistics from Purdue University in 1995 and 1998, respectively. His research interests include pattern recognition, image processing, Bayesian computation, filtering and smoothing, spatio-temporal analysis, and biometric authentication. His statistical modeling techniques and methodologies have been useful in analyzing different aspects of variability in the areas of biometric authentication and neuroscience to ease interpretation of complex data. He has active collaborations in various areas of engineering, computer science, and health. Vijanth S. Asirvadam studied at the University of Putra, Malaysia, in a Bachelor of Science (Hon) degree program with major in Statistics. He graduated in 1997 before leaving for Queen's University in Belfast to do his Masters' degree. He received his Master's of Science degree in Engineering Computation with Distinction. He later joined the Intelligent Systems and Control Research Group at Queen's University, Belfast, in 1999, where he completed his Ph.D. degree with research in the topic of On-line

Rehman *et al. EURASIP Journal on Advances in Signal Processing* (2018) 2018:36

Page 18 of 19

and Constructive Neural Learning Methods. He took his previous employment as a system engineer (1999) and later as a lecturer at Multimedia University, Malaysia, between 2003 and 2005. He was also a senior lecturer at the Faculty of Engineering and Computer Technology at AIMST University from 2005 to 2006. Since 2006, he served as a senior lecturer and later as an associate professor (2011 onwards) in the Department of Electrical and Electronics Engineering, Universiti Teknologi PETRONAS (UTP). His research interests include linear and non-linear system identification, unconstrained optimization, and model validation. On the application side, his main research interests are on computing techniques in signal, image, and video processing. Dr. Vijanth is a member of the Institute of Electrical and Electronics Engineering (IEEE).

Competing interests
The authors declare that they have no competing interests.

Publisher's Note
Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Author details
[1]Fundamental and Applied Sciences Department, Universiti Teknologi Petronas, Seri Iskandar, Malaysia. [2]Faculty of Engineering and Computer Science, National University of Modern Languages, Islamabad, Pakistan. [3]Department of Electrical and Electronic Engineering, Universiti Teknologi Petronas, Seri Iskandar, Malaysia.

References
1. M Ades, PJ Van Leeuwen, An exploration of the equivalent weights particle filter. Quarteraly J. R. Meteorol. Soc. **139**(672), 820–840 (2013)
2. MS Arulampalam, S Maskell, N Gordon, T Clapp, A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. Signal Proc. IEEE Trans. **50**(2), 174–188 (2002)
3. SE Azam, E Chatzi, C Papadimitriou, A Smyth, Experimental validation of the Kalman-type filters for online and real-time state and input estimation. J. Vib. Control. **23**(15), 2494–2519 (2017)
4. T Bengtsson, P Bickel, B Li. Curse-of-dimensionality revisited: collapse of the particle filter in very large scale systems (Institute of Mathematical Statistics, Ohio, 2008), pp. 316–334. https://doi.org/10.1214/193940307000000518
5. S Beyou, A Cuzol, S Subrahmanyam Gorthi, E Mémin, Weighted ensemble transform Kalman filter for image assimilation. Tellus A: Dynamic Meteorology. **65**(1), 18,803 (2013)
6. H Bi, J Ma, F Wang, An improved particle filter algorithm based on ensemble Kalman filter and Markov chain Monte Carlo method. IEEE J Sel. Top. Appl. Earth Obs. Remote Sens. **8**(2), 447–459 (2015)
7. M Briers, A Doucet, S Maskell, Smoothing algorithms for state–space models. Ann. Inst. Stat. Math. **62**(1), 61–89 (2010)
8. J Cornuet, MARIN JM, A Mira, CP Robert, Adaptive multiple importance sampling. Scand. J. Stat. **39**(4), 798–812 (2012)
9. L Dovera, E Della Rossa, Multimodal ensemble Kalman filtering using Gaussian mixture models. Comput. Geosci. **15**(2), 307–323
10. G Evensen, Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. J. Geophys. Res. Oceans (1978–2012). **99**(C5), 10,143–10,162 (1994)
11. G Evensen, PJ Van Leeuwen, An ensemble Kalman smoother for nonlinear dynamics. Mon. Weather Rev. **128**(6), 1852–1867 (2000)
12. M Frei, HR Künsch, Bridging the ensemble Kalman and particle filters. Biometrika. **100**(4), 781–800 (2013)
13. X Fu, Y Jia, An improvement on resampling algorithm of particle filters. IEEE Trans. Sign. Process. **58**(10), 5414–5420 (2010)
14. NJ Gordon, DJ Salmond, AFM Smith, Novel approach to nonlinear/non-Gaussian Bayesian state estimation. IEE Proc. F - Radar Sign. Process. **140**(2), 107–113 (1993). https://doi.org/10.1049/ip-f-2.1993.0015
15. UD Hanebeck, K Briechle, A Rauh, Progressive Bayes: a new framework for nonlinear state estimation. Proc SPIE. **5099**, 256–67 (2003)
16. I Hoteit, DT Pham, M Gharamti, X Luo, Mitigating observation perturbation sampling errors in the stochastic EnKF. Mon. Weather Rev. **143**(7), 2918–2936 (2015)
17. Y Huang, PM Djuric, A hybrid importance function for particle filtering. IEEE Sign. Process. Lett. **11**(3), 404–406 (2004). https://doi.10.1109/LSP.2003.821715
18. N Kantas, A Doucet, SS Singh, JM Maciejowski, An overview of sequential Monte Carlo methods for parameter estimation in general state-space models. IFAC Proc. **42**(10), 774–785 (2009)
19. M Katzfuss, JR Stroud, CK Wikle, Understanding the ensemble Kalman filter. Am. Stat. **70**(4), 350–357 (2016)
20. T Kontoroupi, AW Smyth, Online bayesian model assessment using nonlinear filters. Struct. Control. Health Monit. **24**(3), e1880 (2017). https://onlinelibrary.wiley.com/doi/abs/10.1002/stc.1880
21. PJ van Leeuwen, Nonlinear data assimilation in geosciences: an extremely efficient particle filter. Q. J. R. Meteorol. Soc. **136**(653), 1991–1999 (2010)
22. T Li, M Bolic, PM Djuric, Resampling methods for particle filtering: classification, implementation, and strategies. IEEE Signal Proc. Mag. **32**(3), 70–86 (2015). https://doi.org/10.1109/MSP.2014.2330626
23. T Li, JM Corchado, J Bajo, S Sun, JF Paz, Effectiveness of Bayesian filters: an information fusion perspective. Inf. Sci. **329**, 670–689 (2016)
24. T Li, S Sun, JM Corchado, TP Sattar, S Si, Numerical fitting-based likelihood calculation to speed up the particle filter. Int. J. Adapt. Control. Signal Proc. **30**(11), 1583–1602 (2016)
25. T Li, J Su, W Liu, JM Corchado, Approximate Gaussian conjugacy: parametric recursive filtering under nonlinearity, multimodality, uncertainty, and constraint, and beyond. Front. Inf. Technol. Electron Eng. **18**(12), 1913–1939 (2017). https://doi.org/10.1631/FITEE.1700379
26. G McLachlan, T Krishnan, *Basic Theory of the EM Algorithm*. (Wiley-Blackwell, 2007). https://doi.org/10.1002/9780470191613.ch3
27. Q Miao, L Xie, H Cui, W Liang, Pecht M, Remaining useful life prediction of lithium-ion battery with unscented particle filter technique. Microelectron. Reliab. **53**(6), 805–810 (2013)
28. M Morzfeld, D Hodyss, C Snyder, What the collapse of the ensemble Kalman filter tells us about particle filters. Tellus A Dyn. Meteorol. Oceanogr. **69**(1), 1283,809 (2017)
29. A Olivier, AW Smyth, Particle filtering and marginalization for parameter identification in structural systems. Struct. Control. Health Monit. **24**(3), e1874 (2017). https://onlinelibrary.wiley.com/doi/abs/10.1002/stc.1874
30. N Oudjane, C Musso, in *Proceedings of the Third International Conference on Information Fusion, vol. 2*. Progressive correction for regularized particle filters, (2000), pp. THB2/10–THB2/17. https://doi.org/10.1109/IFIC.2000.859873
31. N Papadakis, Mémin É, A Cuzol, N Gengembre, Data assimilation with the weighted ensemble kalman filter. Tellus A. **62**(5), 673–697 (2010)
32. M Radojević, E Meijering, Automated neuron tracing using probability hypothesis density filtering. Bioinformatics. **33**(7), 1073–1080 (2017)
33. D Raihan, Chakravorty S, in *2016 19th International Conference on Information Fusion (FUSION)*. Particle Gaussian mixture (PGM) filters (IEEE, 2016), pp. 1369–1376
34. B Ristic, S Arulampalam, NJ Gordon, *Beyond the Kalman filter: particle filters for tracking applications. Vol. 3*. (Artech house, London, 2004), pp. 1077–2626
35. S Robert, HR Künsch, Localizing the ensemble Kalman particle filter. Tellus A Dyn. Meteorol. Oceanogr. **69**(1), 1282,016 (2017)
36. S Sarkka, A Nummenmaa, Recursive noise adaptive Kalman filtering by variational Bayesian approximations. IEEE Trans. Autom. Control. **54**(3), 596–600 (2009)
37. Särkkä S, J Hartikainen, IS Mbalawata, H Haario, Posterior inference on parameters of stochastic differential equations via non-linear Gaussian filtering and adaptive MCMC. Stat. Comput. **25**(2), 427–437 (2015). https://doi.org/10.1007/s11222-013-9441-1
38. A Smyth, M Wu, Multi-rate kalman filtering for the data fusion of displacement and acceleration response measurements in dynamic system monitoring. Mech. Syst. Signal Process. **21**(2), 706–723 (2007)
39. C Snyder, T Bengtsson, M Morzfeld, Performance bounds for particle filters using the optimal proposal. Mon Weather Rev. **143**(11), 4750–4761 (2015)
40. O Straka, J Duník, M Šimandl, in *Proceedings of the 2011 American Control Conference*. Truncated unscented particle filter (IEEE, 2011), pp. 1825–1830. https://doi.org/10.1109/ACC.2011.5991296

41.　G Tong, Z Fang, X Xu, in *2006 IEEE International Conference on Evolutionary Computation*. A particle swarm optimized particle filter for nonlinear system state estimation (IEEE, 2006), pp. 438–442. https://doi.org/10.1109/CEC.2006.1688342

42.　BN Vo, WK Ma, The Gaussian mixture probability hypothesis density filter. IEEE Trans. Sign. Process. **54**(11), 4091–4104 (2006)

43.　X Wang, W Ni, An improved particle filter and its application to an INS/GPS integrated navigation system in a serious noisy scenario. Meas. Sci. Technol. **27**(9), 095,005 (2016)

44.　X Wang, T Li, S Sun, JM Corchado, A survey of recent advances in particle filters and remaining challenges for multitarget tracking. Sensors. **17**(12), 2707 (2017)

45.　C Zhang, R Zhi, T Li, J Corchado, in *2016 Sensor Signal Processing for Defence (SSPD)*. Adaptive m-estimation for robust cubature Kalman filtering (IEEE, 2016), pp. 1–5. https://doi.org/10.1109/SSPD.2016.7590586

46.　J Zhu, X Wang, Q Fang, in *2013 International Conference on Information Science and Cloud Computing Companion*. The improved particle filter algorithm based on weight optimization (IEEE, 2013), pp. 351–356. https://doi.org/10.1109/ISCC-C.2013.140

47.　J Zuo, Y Jia, Q Gao, Simplified unscented particle filter for nonlinear/non-gaussian Bayesian estimation. J. Syst. Eng. Electron. **24**(3), 537–544 (2013). https://doi.org/10.1109/JSEE.2013.00062