Memorandum 2009 (May 2013). ISSN 1874-4850. Available from: http://www.math.utwente.nl/publications
Department of Applied Mathematics, University of Twente, Enschede, The Netherlands

The Unscented Rao-Blackwellized Marginal Particle Filter applied to the joint state and parameter estimation problem

Edson Hiroshi Aoki

April 25, 2013

1 Problem mathematical formulation

Let us consider a time-varying system described by k, θ, S_k, Z_k, where k is the time index, θ is a vector of parameters, S_k is a random vector corresponding to the time-varying state at time k (with the corresponding realization denoted by s_k), and Z_k is a random vector corresponding to the observation process at time k (with the corresponding realization denoted by z_k). We assume that the time evolution of the system is characterized by

$$s_{k+1} = f_k\left(s_k, \theta, \xi_k^s\right) \tag{1}$$

$$z_k = h_k\left(s_k, \theta, \xi_k^z\right) \tag{2}$$

$$s_0 \sim p(s_0) \tag{3}$$

where f_k and h_k are arbitrary nonlinear functions, and $(\Xi_k^s)_{k=0}^{\infty}$ and $(\Xi_k^z)_{k=1}^{\infty}$ (with realizations denoted by $(\xi_k^s)_{k=0}^{\infty}$ and $(\xi_k^z)_{k=1}^{\infty}$) are mutually independent and time-independent noise sequences, also independent from $p(s_0)$.

Our goal is to obtain estimates $\hat{\theta}$ and $\hat{s}_k$ respectively of θ and S_k, given all available observations $Z^k \triangleq (z_1, \ldots, z_k)$. We tackle the problem using a *coupled Bayesian strategy*, where states and parameters are treated as a single augmented state $\left[S_k^T, \Theta^T\right]^T$, with realizations given by $\left[s_k^T, \theta^T\right]^T$, and the statistical information about the augmented state summarized by the joint posterior density $p\left(s_k, \theta \left| Z^k\right.\right)$. For our described system, we assume that the joint prior density $p(s_0, \theta)$ is independent of $(\Xi_k^s)_{k=0}^{\infty}$ and $(\Xi_k^z)_{k=1}^{\infty}$.

2 Derivation of the Unscented RBMPF

The RBMPF uses the following approximation for the joint posterior density:

$$p\left(s_k, \theta \left| Z^k\right.\right) \approx \sum_{i=1}^{N_P} w_k(i)\delta(s_k - s_k(i)) p\left(\theta \left| s_k(i), Z^k\right.\right) \tag{4}$$

and hence, in order to devise a practical implementation of the RBMPF, we must be able to:

1. Approximate $p\left(s_k \left| Z^k\right.\right)$ using a SMC method;
2. Calculate or approximate $p\left(\theta \left| s_k(i), Z^k\right.\right)$ using a non-SMC method.

In the Unscented RBMPF (URBMPF) we will assume that $p\left(\theta \left| s_k(i), Z^k\right.\right)$ is Gaussian, such that (4) has the form

$$p\left(s_k, \theta \left| Z^k\right.\right) \approx \sum_{i=1}^{N_P} w_k(i)\delta(s_k - s_k(i)) \mathcal{N}\left(\theta; \hat{\theta}_k(i), P_k^{\theta}(i)\right) \tag{5}$$

where $\hat{\theta}_k(i)$ and $P_k^{\theta}(i)$ are respectively the mean and covariance of $p\left(\theta \left| s_k(i), Z^k\right.\right)$.

The algorithm consists of using a MPF to estimate $p\left(s_k \left| Z^k\right.\right)$, and a multiple model Unscented filter for $p\left(\theta \left| s_k(i), Z^k\right.\right)$.

In our proposed solution, we will sometimes require expressions for expectations taken over $p\left(\theta \left| s_k(i), Z^k \right.\right)$. As a suboptimal solution, we proposed to simply use the mean $\hat{\theta}_k(i)$ on computation of such expectations, such that $E\left[g(S_k, \Theta) \left| Z^k \right.\right]$ (where $g(\cdot, \cdot)$ is an arbitrary function of s_k and θ) is approximated as

$$E\left[g(S_k, \Theta) \left| Z^k \right.\right] \approx \sum_{i=1}^{N_P} w_k(i) g(s_k(i), \hat{\theta}_k(i)) \tag{6}$$

and the output of the filter at each time step k consists of

$$\left\{ s_k(i), w_k(i), \hat{\theta}_k(i), P_k^{\theta}(i) \right\}_{i=1}^{N_P}. \tag{7}$$

From (7), each iteration of the filter must have two steps: one to obtain $\{s_k(i), w_k(i)\}_{i=1}^{N_P}$, and another one to obtain $\left\{ \hat{\theta}_k(i), P_k^{\theta}(i) \right\}_{i=1}^{N_P}$. We will now derive each of these two steps.

2.1 Obtaining the particle states/weights (MPF step)

In order to obtain $\{s_k(i), w_k(i)\}_{i=1}^{N_P}$, we will derive a modified version of the MPF. Observe first that

$$\begin{aligned} p\left(s_k \left| Z^k \right.\right) &= \int \int p\left(s_k, s_{k-1}, \theta \left| Z^k \right.\right) ds_{k-1} d\theta \\ &= \int \int \frac{p\left(z_k \left| s_k, s_{k-1}, \theta, Z^{k-1} \right.\right)}{p\left(z_k \left| Z^{k-1} \right.\right)} p\left(s_k, s_{k-1}, \theta \left| Z^{k-1} \right.\right) ds_{k-1} d\theta. \end{aligned} \tag{8}$$

From (1) and (2), we have

$$p\left(z_k \left| s_k, s_{k-1}, \theta, Z^{k-1} \right.\right) = p\left(z_k \left| s_k, \theta \right.\right) \tag{9}$$

$$p\left(s_k \left| s_{k-1}, \theta, Z^{k-1} \right.\right) = p\left(s_k \left| s_{k-1}, \theta \right.\right) \tag{10}$$

and therefore

$$\begin{aligned} p(s_k|Z^k) &= \int \int \frac{p(z_k|s_k, \theta) p(s_k|s_{k-1}, \theta)}{p\left(z_k \left| Z^{k-1} \right.\right)} p\left(s_{k-1}, \theta \left| Z^{k-1} \right.\right) ds_{k-1} d\theta \\ &= \frac{E\left[p(z_k|s_k, \Theta) p(s_k|S_{k-1}, \Theta) \left| Z^{k-1} \right.\right]}{p\left(z_k \left| Z^{k-1} \right.\right)}. \end{aligned} \tag{11}$$

Now, observe that a conditional expectation of the form $E\left[g(S_k) \left| Z^k \right.\right]$ (where $g(\cdot)$ is an arbitrary function of s_k) is given by

$$\begin{aligned} E\left[g(S_k) \left| Z^k \right.\right] &= \int g(s_k) p(s_k|Z^k) ds_k \\ &= \int g(s_k) \frac{E\left[p(z_k|s_k, \Theta) p(s_k|S_{k-1}, \Theta) \left| Z^{k-1} \right.\right]}{p\left(z_k \left| Z^{k-1} \right.\right)} ds_k \\ &= \int g(s_k) \frac{E\left[p(z_k|s_k, \Theta) p(s_k|S_{k-1}, \Theta) \left| Z^{k-1} \right.\right]}{p\left(z_k \left| Z^{k-1} \right.\right) q\left(s_k \left| Z^k \right.\right)} q\left(s_k \left| Z^k \right.\right) ds_k \end{aligned} \tag{12}$$

where $q\left(s_k \left| Z^k \right.\right)$ is an appropriate proposal density for S_k. If we generate N_P independent, identically distributed particle states $s_k(i)$ by sampling from $q\left(s_k \left| Z^k \right.\right)$, then using the law of large numbers, $E\left[g(S_k) \left| Z^k \right.\right]$ may be approximated by

$$E\left[g(S_k) \left| Z^k \right.\right] \approx \sum_{i=1}^{N_P} g(s_k(i)) w_k(i) \tag{13}$$

where the particle weights are given by

$$w_k(i) = \frac{E\left[p(z_k|s_k(i),\Theta)p(s_k(i)|S_{k-1},\Theta)\left|Z^{k-1}\right.\right]}{N_P p\left(z_k\left|Z^{k-1}\right.\right) q\left(s_k(i)\left|Z^k\right.\right)}. \tag{14}$$

To calculate (14), we approximate the numerator of (14) using (6), i.e. we use the output (7) produced at the previous iteration $k-1$:

$$\begin{aligned} &E\left[p(z_k|s_k(i),\Theta)p(s_k(i)|S_{k-1},\Theta)\left|Z^{k-1}\right.\right] \\ &\approx \sum_{j=1}^{N_P} w_{k-1}(j)p(z_k|s_k(i),\hat{\theta}_{k-1}(j))p(s_k(i)|s_{k-1}(j),\hat{\theta}_{k-1}(j)). \end{aligned} \tag{15}$$

Approximation (6) will also typically be needed to obtain $q\left(s_k(i)\left|Z^k\right.\right)$. Observe that the term $N_P p\left(z_k\left|Z^{k-1}\right.\right)$ in (14) is irrelevant as it does not depend on the particle state $s_k(i)$, and hence can be taken in account by normalizing the weights. Note also that unlike the SIR particle filter, the MPF does not contain a resampling step. Let us now give a look at the options for the proposal density $q\left(s_k\left|Z^k\right.\right)$.

2.1.1 Optimal importance sampling

From (14), we can see that the variance of weights would be minimized by making $q\left(s_k\left|Z^k\right.\right) = \frac{E\left[K(s_k,S_{k-1},\Theta,z_k,z_{k-1})\left|Z^{k-1}\right.\right]}{p(z_k|Z^{k-1})}$. The optimal proposal density can be approximated as

$$q\left(s_k\left|Z^k\right.\right) \approx \frac{\sum_{j=1}^{N_P} w_{k-1}(j)\int p(z_k|s_k,\theta)p(s_k|s_{k-1}(j),\theta)\mathcal{N}\left(\theta;\hat{\theta}_{k-1}(j),P^{\theta}_{k-1}(j)\right)d\theta}{p\left(z_k\left|Z^{k-1}\right.\right)}. \tag{16}$$

As in the case of particle filters in general, in most cases it is not possible to sample directly from (16), such that optimal importance sampling can be at best approximated. Note that, although the MPF does not contain a resampling step, to sample from (16) it is necessary to use a resampling-like mechanism due to the term $\sum_{j=1}^{N_P} w_{k-1}(j)$. Any resampling scheme (such as systematic resampling) can be used in this step.

2.1.2 Blind importance sampling

In particle filtering, a popular alternative to optimal importance sampling (which is also generally easier to implement) is "blind" importance sampling, i.e. the choice of proposal density that disregards the last observation z_k. This can be accomplished by making

$$\begin{aligned} q\left(s_k\left|Z^k\right.\right) &= p\left(s_k\left|Z^{k-1}\right.\right) \\ &= E\left[p(s_k|S_{k-1},\Theta)\left|Z^{k-1}\right.\right] \end{aligned} \tag{17}$$

which can be approximated as

$$q\left(s_k\left|Z^k\right.\right) \approx \sum_{j=1}^{N_P} w_{k-1}(j)\int p\left(s_k\left|s_{k-1}(j),\theta\right.\right)\mathcal{N}\left(\theta;\hat{\theta}_{k-1}(j),P^{\theta}_{k-1}(j)\right)d\theta \tag{18}$$

and hence, like optimal importance sampling, blind importance sampling also requires a resampling-like mechanism.

2.2 Obtaining Gaussian approximations for the parameter conditionals (UKF step)

In order to obtain $\left\{\hat{\theta}_k(i),P^{\theta}_k(i)\right\}_{i=1}^{N_P}$, first, observe that

$$\begin{aligned} p\left(\theta\left|s_k(i),Z^k\right.\right) &= \frac{p\left(\theta,s_k(i),z_k\left|Z^{k-1}\right.\right)}{p\left(s_k(i),z_k\left|Z^{k-1}\right.\right)} \\ &= \frac{\int p\left(\theta,s_k(i),z_k\left|s_{k-1},Z^{k-1}\right.\right)p\left(s_{k-1}\left|Z^{k-1}\right.\right)ds_{k-1}}{p\left(s_k(i),z_k\left|Z^{k-1}\right.\right)} \end{aligned} \tag{19}$$

By using the set of particles obtained at the previous iteration $k-1$, we can approximate (19) by

$$\begin{aligned} p\left(\theta \left| s_k(i), Z^k\right.\right) &\approx \frac{\sum_{j=1}^{N_P} w_{k-1}(j) p\left(\theta, s_k(i), z_k \left| s_{k-1}(j), Z^{k-1}\right.\right)}{p\left(s_k(i), z_k \left| Z^{k-1}\right.\right)} \\ &= \frac{\sum_{j=1}^{N_P} w_{k-1}(j) p\left(s_k(i), z_k \left| s_{k-1}(j), Z^{k-1}\right.\right) p\left(\theta \left| s_k(i), z_k, s_{k-1}(j), Z^{k-1}\right.\right)}{p\left(s_k(i), z_k \left| Z^{k-1}\right.\right)} \end{aligned} \tag{20}$$

where, using the suboptimal approximation (6), the term $p\left(s_k(i), z_k \left| s_{k-1}(j), Z^{k-1}\right.\right)$ may be approximated as

$$\begin{aligned} p\left(s_k(i), z_k \left| s_{k-1}(j), Z^{k-1}\right.\right) &\approx p\left(s_k(i), z_k \left| \hat{\theta}_{k-1}(j), s_{k-1}(j), Z^{k-1}\right.\right) \\ &= p\left(z_k \left| s_k(i), \hat{\theta}_{k-1}(j)\right.\right) p\left(s_k(i) \left| \hat{\theta}_{k-1}(j), s_{k-1}(j)\right.\right) \end{aligned} \tag{21}$$

and by defining the "cross-particle" weight

$$\tilde{w}_{k-1}(i,j) \triangleq \frac{w_{k-1}(j) p\left(z_k \left| s_k(i), \hat{\theta}_{k-1}(j)\right.\right) p\left(s_k(i) \left| \hat{\theta}_{k-1}(j), s_{k-1}(j)\right.\right)}{p\left(s_k(i), z_k \left| Z^{k-1}\right.\right)} \tag{22}$$

we can rewrite (20) as

$$p\left(\theta \left| s_k(i), Z^k\right.\right) \approx \sum_{j=1}^{N_P} \tilde{w}_{k-1}(i,j) p\left(\theta \left| s_k(i), z_k, s_{k-1}(j), Z^{k-1}\right.\right) \tag{23}$$

2.2.1 Obtaining a Gaussian approximation for the cross-particle parameter conditional density

Now, let us consider a Gaussian approximation for $p\left(\theta \left| s_k(i), z_k, s_{k-1}(j), Z^{k-1}\right.\right)$, i.e.

$$p\left(\theta \left| s_k(i), z_k, s_{k-1}(j), Z^{k-1}\right.\right) \approx \mathcal{N}\left(\theta; \tilde{\hat{\theta}}_k(i,j), \tilde{P}_k^{\theta}(i,j)\right) \tag{24}$$

and since we are relying on the approximation

$$p\left(\theta \left| s_{k-1}(j), Z^{k-1}\right.\right) \approx \mathcal{N}\left(\theta; \hat{\theta}_{k-1}(j), P_{k-1}^{\theta}(j)\right) \tag{25}$$

then our problem is of obtaining a posterior Gaussian density from a prior Gaussian density and an (augmented) random observation

$$Y \triangleq \begin{bmatrix} S_k \\ Z_k \end{bmatrix} \tag{26}$$

(with realization denoted by y), a problem that we can address using linear Bayesian estimation techniques, such as the Extended Kalman Filter (EKF), the Unscented Kalman filter (UKF) or higher-order versions of these two algorithms.

In this work, we will use the UKF. By applying the principle of orthogonality to find the optimal linear unbiased estimate, we have

$$\tilde{\hat{\theta}}_k(i,j) = \hat{\theta}_{k-1}(j) + P^{\theta y}(j) P^{y}(j)^{-1} \left(y(i) - \hat{y}(j)\right) \tag{27}$$

$$\tilde{P}_k^{\theta}(i,j) = P_{k-1}^{\theta}(j) - P^{\theta y}(j) P^{y}(j)^{-1} P^{\theta y}(j)^T \tag{28}$$

where

$$\begin{aligned} \hat{y}(j) &= E\left[Y_k \left| s_{k-1}(j), Z^{k-1}\right.\right] \\ P^{\theta y}(j) &= E\left[\left(\Theta - \hat{\theta}_{k-1}(j)\right)\left(Y_k - \hat{y}(j)\right)^T \right| s_{k-1}(j), Z^{k-1}\Big] \\ P^{y}(j) &= E\left[\left(Y_k - \hat{y}(j)\right)\left(Y_k - \hat{y}(j)\right)^T \right| s_{k-1}(j), Z^{k-1}\Big]. \end{aligned}$$

We will obtain $\hat{y}(j)$, $P^{\theta y}(j)$ and $P^{y}(j)$ using the Unscented transform. First, we define the augmented random vector

$$X \triangleq \begin{bmatrix} \Theta \\ \Xi_{k-1}^{s} \\ \Xi_{k}^{z} \end{bmatrix} \tag{29}$$

with realizations denoted by x, such that the mean and covariance of X given $s_{k-1}(j)$ and Z^{k-1} would be given by

$$\hat{x}(j) = \begin{bmatrix} \hat{\theta}_{k-1}(j) \\ \hat{\xi}_{k-1}^{s} \\ \hat{\xi}_{k}^{z} \end{bmatrix}, P^{x}(j) = \begin{bmatrix} P_{k-1}^{\theta}(j) & 0 & 0 \\ 0 & P_{k-1}^{\xi^s} & 0 \\ 0 & 0 & P_{k}^{\xi^z} \end{bmatrix}. \tag{30}$$

where $\hat{\xi}_{k-1}^{s}$ and $\hat{\xi}_{k}^{z}$ correspond to the means of the noise sequences, and $P_{k-1}^{\xi^s}$ and $P_{k}^{\xi^z}$ to their covariances. Note that augmenting the parameter vector with the noise terms can be avoided when the noises are addictive (see [1]).

We then generate a set of weighted sigma points using

$$\left\{\mathcal{X}(j,m), \mathcal{W}(j,m), \mathcal{W}^{P}(j,m)\right\}_{m=1}^{N_S} = \text{SG}\left(\hat{x}(j), P^{x}(j)\right) \tag{31}$$

where N_S is the number of sigma points, SG is a sigma point generation method, and for $m = 1, \ldots, N_S$, $\mathcal{X}(j,m)$ corresponds to the sigma point itself, $\mathcal{W}(j,m)$ to the sigma point weight, and $\mathcal{W}^{P}(j,m)$ to the sigma point weight for the purpose of computing 2nd order moments. Sigma point generation is discussed more in-depth in Section 3.

Now, by noting that each sigma point $\mathcal{X}(j,m)$, $m = 1, \ldots, N_S$, is given by

$$\mathcal{X}(j,m) \triangleq \begin{bmatrix} \mathcal{X}^{\theta}(j,m) \\ \mathcal{X}^{\xi_s}(m) \\ \mathcal{X}^{\xi_z}(m) \end{bmatrix} \tag{32}$$

where $\mathcal{X}^{\theta}(j,m)$, $\mathcal{X}^{\xi_s}(m)$ and $\mathcal{X}^{\xi_z}(m)$ denote the components of the vector corresponding to Θ, Ξ_{k-1}^{s} and Ξ_{k}^{z} respectively, we can generate a set of sigma points for Y conditioned on $s_{k-1}(j)$ and Z^{k-1} using

$$\mathcal{Y}(j,m) = \begin{bmatrix} f_{k-1}\left(s_{k-1}(j), \mathcal{X}^{\theta}(j,m), \mathcal{X}^{\xi_s}(m)\right) \\ h_k\left(f_{k-1}\left(s_{k-1}(j), \mathcal{X}^{\theta}(j,m), \mathcal{X}^{\xi_s}(m)\right), \mathcal{X}^{\theta}(j,m), \mathcal{X}^{\xi_z}(m)\right) \end{bmatrix}, m = 1, \ldots, N_S \tag{33}$$

and finally, $\hat{y}(j)$, $P^{\theta y}(j)$ and $P^{y}(j)$ and can be calculated using

$$\begin{aligned} \hat{y}(j) &\approx \sum_{m=1}^{N_S} \mathcal{W}(j,m)\mathcal{Y}(j,m) \\ P^{\theta y}(j) &\approx \sum_{m=1}^{N_S} \mathcal{W}^{P}(j,m)\left(\mathcal{X}^{\theta}(j,m) - \hat{\theta}_{k-1}(j)\right)\left(\mathcal{Y}(j,m) - \hat{y}(j)\right)^{T} \\ P^{y}(j) &\approx \sum_{m=1}^{N_S} \mathcal{W}^{P}(j,m)\left(\mathcal{Y}(j,m) - \hat{y}(j)\right)\left(\mathcal{Y}(j,m) - \hat{y}(j)\right)^{T}. \end{aligned}$$

2.2.2 Obtaining a Gaussian approximation for the particle parameter conditional density

After obtaining (24) using (28), we obtain the following approximation for $p\left(\theta \left| s_k(i), Z^k \right.\right)$:

$$p\left(\theta \left| s_k(i), Z^k \right.\right) \approx \sum_{j=1}^{N_P} \tilde{w}_{k-1}(i,j)\mathcal{N}\left(\theta; \tilde{\hat{\theta}}_k(i,j), \tilde{P}_k^{\theta}(i,j)\right). \tag{34}$$

Now, all we need is to obtain a Gaussian approximation $\mathcal{N}\left(\theta; \hat{\theta}_k(i), P_k^\theta(i)\right)$ for $p\left(\theta \left| s_k(i), Z^k \right.\right)$, which can be done in a straightforward manner:

$$\hat{\theta}_k(i) = \sum_{j=1}^{N_P} \tilde{w}_{k-1}(i,j)\tilde{\hat{\theta}}_k(i,j) \tag{35}$$

$$P_k^\theta(i) = \sum_{j=1}^{N_P} \tilde{w}_{k-1}(i,j)\left(\tilde{P}_k^\theta(i,j) + \left(\tilde{\hat{\theta}}_k(i,j) - \hat{\theta}_k(i)\right)\left(\tilde{\hat{\theta}}_k(i,j) - \hat{\theta}_k(i)\right)^T\right). \tag{36}$$

2.3 Algorithm

Initialization:

1. For each particle $i = 1, \ldots, N_P$
   (a) Sample $s_0(i) \sim p(s_0)$
   (b) Make $w_0(i) = \frac{1}{N_P}$
   (c) Make $\hat{\theta}_0(i) = E\left[\theta | s_0(i)\right]$
   (d) Make $P_0^\theta(i) = E\left[\left(\theta - \hat{\theta}_0(i)\right)\left(\theta - \hat{\theta}_0(i)\right)^T \middle| s_0(i)\right]$

At every time step $k = 1, 2, \ldots$:

1. For each particle $i = 1, \ldots, N_P$
   (a) Perform particle importance sampling by drawing

$$s_k(i) \sim q\left(s_k \middle| Z^k\right)$$

   where $q\left(s_k \middle| Z^k\right)$ is a proposal density:

   - *Optimal proposal density* (generally not feasible)

$$q\left(s_k \middle| Z^k\right) \approx \frac{\sum_{j=1}^{N_P} w_{k-1}(j) \int p(z_k|s_k,\theta) p(s_k|s_{k-1}(j),\theta) \mathcal{N}\left(\theta; \hat{\theta}_{k-1}(j), P_{k-1}^\theta(j)\right) d\theta}{p\left(z_k \left| Z^{k-1} \right.\right)}$$

   - *Blind proposal density*

$$q\left(s_k \middle| Z^k\right) \approx \sum_{j=1}^{N_P} w_{k-1}(j) \int p\left(s_k \left| s_{k-1}(j), \theta \right.\right) \mathcal{N}\left(\theta; \hat{\theta}_{k-1}(j), P_{k-1}^\theta(j)\right) d\theta$$

   (b) Calculate the unnormalized particle weight according to

$$\overline{w}_k(i) = \frac{\sum_{j=1}^{N_P} w_{k-1}(j) p(z_k|s_k(i), \hat{\theta}_{k-1}(j)) p(s_k(i)|s_{k-1}(j), \hat{\theta}_{k-1}(j))}{q\left(s_k(i) \left| Z^k \right.\right)}$$

   where we consider the approximations:

   - *Optimal proposal density*

$$q\left(s_k(i) \middle| Z^k\right) \approx \sum_{j=1}^{N_P} w_{k-1}(j) p(z_k|s_k(i), \hat{\theta}_{k-1}(j)) p(s_k(i)|s_{k-1}(j), \hat{\theta}_{k-1}(j))$$

   - *Blind proposal density*

$$q\left(s_k(i) \middle| Z^k\right) \approx \sum_{j=1}^{N_P} w_{k-1}(j) p\left(s_k(i) \middle| s_{k-1}(j), \hat{\theta}_{k-1}(j)\right)$$

(c) Generate a set of N_S weighted sigma points for $\Theta, \Xi^s_{k-1}, \Xi^z_k$ conditioned on $s_{k-1}(i), Z^{k-1}$ using

$$\left\{\mathcal{X}(i,m), \mathcal{W}(i,m), \mathcal{W}^P(i,m)\right\}_{m=1}^{N_S} = \text{SG}\left(\hat{x}(i), P^x(i)\right)$$

where SG is a sigma point generation method for an Unscented transform (our proposed method is described in Section 3.2) and

$$\hat{x}(i) = \begin{bmatrix} \hat{\theta}_{k-1}(i) \\ \hat{\xi}^s_{k-1} \\ \hat{\xi}^z_k \end{bmatrix}, P^x(i) = \begin{bmatrix} P^\theta_{k-1}(i) & 0 & 0 \\ 0 & P^{\xi^s}_{k-1} & 0 \\ 0 & 0 & P^{\xi^z}_k \end{bmatrix}$$

where $\hat{\xi}^s_{k-1}, \hat{\xi}^z_k$ correspond respectively to the means of the noise sequences Ξ^s_{k-1}, Ξ^z_k, and $P^{\xi^s}_{k-1}, P^{\xi^z}_k$ to their respective covariances

(d) By noting that, for $m = 1, \ldots, N_S$

$$\mathcal{X}(i,m) \triangleq \begin{bmatrix} \mathcal{X}^\theta(i,m) \\ \mathcal{X}^{\xi_s}(m) \\ \mathcal{X}^{\xi_z}(m) \end{bmatrix}$$

generate the sigma points for S_k, Z_k conditioned on $s_{k-1}(i), Z^{k-1}$

$$\mathcal{Y}(i,m) = \begin{bmatrix} f_{k-1}\left(s_{k-1}(i), \mathcal{X}^\theta(i,m), \mathcal{X}^{\xi_s}(m)\right) \\ h_k\left(f_{k-1}\left(s_{k-1}(i), \mathcal{X}^\theta(i,m), \mathcal{X}^{\xi_s}(m)\right), \mathcal{X}^\theta(i,m), \mathcal{X}^{\xi_z}(m)\right) \end{bmatrix}, m = 1, \ldots, N_S$$

(e) Calculate the following statistics of S_k, Z_k conditioned on $s_{k-1}(i), Z^{k-1}$ using

$$\hat{y}(i) \approx \sum_{m=1}^{N_S} \mathcal{W}(i,m)\mathcal{Y}(i,m)$$

$$P^{\theta y}(i) \approx \sum_{m=1}^{N_S} \mathcal{W}^P(i,m)\left(\mathcal{X}^\theta(i,m) - \hat{\theta}_{k-1}(i)\right)\left(\mathcal{Y}(i,m) - \hat{y}(i)\right)^T$$

$$P^y(i) \approx \sum_{m=1}^{N_S} \mathcal{W}^P(i,m)\left(\mathcal{Y}(i,m) - \hat{y}(i)\right)\left(\mathcal{Y}(i,m) - \hat{y}(i)\right)^T$$

2. Normalize the particle weights according to

$$w_k(i) = \frac{\overline{w}_k(i)}{\sum_{j=1}^{N_P} \overline{w}_k(i)}, \quad i = 1, \ldots, N_P$$

3. For each particle $i = 1, \ldots, N_P$

(a) For each particle $j = 1, \ldots, N_P$

i. Calculate the mean and covariance of Θ conditioned on $s_k(i), s_{k-1}(j), Z^k$ using

$$\tilde{\hat{\theta}}_k(i,j) = \hat{\theta}_{k-1}(j) + P^{\theta y}(j)P^y(j)^{-1}\left(y(i) - \hat{y}(j)\right)$$
$$\tilde{P}^\theta_k(i,j) = P^\theta_{k-1}(j) - P^{\theta y}(j)P^y(j)^{-1}P^{\theta y}(j)^T$$

where $y(i) = \begin{bmatrix} s_k(i) \\ z_k \end{bmatrix}$

ii. Calculate the unnormalized cross-particle weight

$$\overline{\tilde{w}}_{k-1}(i,j) = w_{k-1}(j)p\left(z_k \left| s_k(i), \hat{\theta}_{k-1}(j)\right.\right) p\left(s_k(i) \left| \hat{\theta}_{k-1}(j), s_{k-1}(j)\right.\right)$$

(b) Normalize the cross-particle weights according to

$$\tilde{w}_{k-1}(i,j) = \frac{\overline{\tilde{w}}_{k-1}(i,j)}{\sum_{j=1}^{N_P} \overline{\tilde{w}}_{k-1}(i,j)}, \quad j = 1, \ldots, N_P$$

(c) Calculate the mean and covariance of Θ conditioned on $s_k(i), Z^k$ using

$$\hat{\theta}_k(i) = \sum_{j=1}^{N_P} \tilde{w}_{k-1}(i,j)\tilde{\hat{\theta}}_k(i,j)$$

$$P^\theta_k(i) = \sum_{j=1}^{N_P} \tilde{w}_{k-1}(i,j)\left(\tilde{P}^\theta_k(i,j) + \left(\tilde{\hat{\theta}}_k(i,j) - \hat{\theta}_k(i)\right)\left(\tilde{\hat{\theta}}_k(i,j) - \hat{\theta}_k(i)\right)^T\right)$$

2.4 Computational complexity

Taking everything but N_P and N_S as constant, the asymptotic complexity of the Unscented RBMPF is $O(N_P^2)$. The number of sigma points N_S does not affect asymptotic complexity, unless it is higher than the number of particles. Therefore the Unscented RBMPF has the same complexity as the regular MPF (although computational cost will typically be higher due to the several extra steps involved).

3 On sigma point generation for the Unscented transform

3.1 Guidelines for sigma point generation

Several sigma-point generation methods for the Unscented transform are presented in [1, 2, 3]. The task of generating a good set of sigma points is not trivial. As noted in [3], although any sigma point generation method guarantees that the first two moments of the original distribution are captured correctly, they may result in a poor approximation of higher-order moments, such that the sigma point representation of the transformed distribution can also be poor, depending on non-linearities. In this section we will present some guidelines for the sigma point generation, mostly based on personal experience.

In the original sigma point generation method presented in [1], we have $N_S = 2n^x + 1$ sigma points (where n^x is the dimension of augmented state X), and the weighted sigma points are calculated using

$$\begin{aligned}
\mathcal{X}(j, 2n^x + 1) &= \hat{x}(j), & \mathcal{W}(j, 2n^x + 1) &= \frac{\kappa}{n^x + \kappa} \\
\mathcal{X}(j, m) &= \hat{x}(j) + \left(\sqrt{(n^x + \kappa)P^x(j)}\right)_i, & \mathcal{W}(j, m) &= \frac{1}{2(n^x + \kappa)}, \quad m = 1, \ldots, n^x \\
\mathcal{X}(j, m + n^x) &= \hat{x}(j) - \left(\sqrt{(n^x + \kappa)P^x(j)}\right)_i, & \mathcal{W}(j, m + n^x) &= \frac{1}{2(n^x + \kappa)}, \quad m = 1, \ldots, n^x \\
\mathcal{W}^P(j, m) &= \mathcal{W}(j, m), & & \quad m = 1, \ldots, N_S
\end{aligned} \tag{37}$$

where $(\cdot)_i$ denotes the i-th row or column of a matrix and $\kappa \in \mathbb{R}$ is a tuning parameter that scales the sigma points towards or away from the mean of the prior.

Since the distance of the sigma points to the mean increases with n_x, for high-dimensional problems, we would like to make κ negative to prevent sigma points getting too far from mean. Unfortunately, this will often result in the covariance $P^y(i)$ (i.e. of the transformed distribution) becoming non-positive definite. A solution is to use the *scaled Unscented transform* presented in [3]. The scaled Unscented transform needs to be combined with an existing sigma point generation method. When combined with the method given by (37), the sigma point generation is performed according to

$$\begin{aligned}
\mathcal{X}(j, 2n^x + 1) &= \hat{x}(j), & \mathcal{W}(j, 2n^x + 1) &= \frac{\lambda}{n^x + \lambda} \\
\mathcal{X}(j, m) &= \hat{x}(j) + \left(\sqrt{(n^x + \lambda)P^x(j)}\right)_i, & \mathcal{W}(j, m) &= \frac{1}{2(n^x + \lambda)}, \quad m = 1, \ldots, n^x \\
\mathcal{X}(j, m + n^x) &= \hat{x}(j) - \left(\sqrt{(n^x + \lambda)P^x(j)}\right)_i, & \mathcal{W}(j, m + n^x) &= \frac{1}{2(n^x + \lambda)}, \quad m = 1, \ldots, n^x \\
\mathcal{W}^P(j, m) &= \mathcal{W}(j, m), & & \quad m = 1, \ldots, 2n^x \\
\mathcal{W}^P(j, 2n^x + 1) &= \mathcal{W}(j, 2n^x + 1) - \alpha^2 + \beta, & &
\end{aligned} \tag{38}$$

where $\alpha > 0$, $\lambda = \alpha^2(n^x + \kappa) - n^x$, and β is another tuning parameter, with interpretation not really clear to us. In our practical experience, we have just set $\kappa = 0$, $\beta = 0$ and tuned only the α parameter. A high value of n_x is compensated by making α smaller than 1. For Gaussian distributions, the "rule of thumb" proposed in [1] is to make $n^x + \lambda = 3$, or, equivalently, $\alpha = \sqrt{3/n^x}$ for $\kappa = 0$ and $\beta = 0$.

As we have verified in [4], the Unscented transform may also perform poorly when the entries of the state vector (in our case, the augmented parameter vector X) have large difference in the magnitudes of their values. This is quite common when the different parameters represent dissimilar physical quantities.

As an example, let us assume that we have two parameters, one parameter typically assumes values between -1 and 1, and another between 0 and $1 \cdot 10^6$. We may then have, for some j, $\hat{x}(j) = \begin{bmatrix} -0.6 \\ 4 \cdot 10^5 \end{bmatrix}$ and $P^x(j) =$

$\begin{bmatrix} 0.04 & -6000 \\ -6000 & 7 \cdot 10^9 \end{bmatrix}$. If we then generate the sigma points according to method (38), computing the matrix square root using the Cholesky method, the set of sigma points will be

$$\left\{ \begin{bmatrix} -0.2536 \\ 4 \cdot 10^5 \end{bmatrix}, \begin{bmatrix} -5.1962 \cdot 10^4 \\ 5.3528 \cdot 10^5 \end{bmatrix}, \begin{bmatrix} -0.9464 \\ 4 \cdot 10^5 \end{bmatrix}, \begin{bmatrix} 5.1961 \cdot 10^4 \\ 2.6472 \cdot 10^5 \end{bmatrix}, \begin{bmatrix} -0.6 \\ 4 \cdot 10^5 \end{bmatrix} \right\} \tag{39}$$

It is easy to see why (39) is a poor sigma point representation of the underlying distribution $p(x) = \mathcal{N}(x; \hat{x}(j), P^x(j))$. For two of the sigma points (with a combined weight of $1/3$), the first parameter has a value of $-5.1962 \cdot 10^4$ and $5.1961 \cdot 10^4$ respectively, despite it having a mean and standard deviation of -0.6 and 0.2 respectively! Therefore, it is expected that the Unscented transform applied to this set of sigma points will also perform poorly.

The simple but effective solution that we employed in [4] was to perform a rescaling of the state vector entries before applying the Unscented transform. Let $\mathcal{F}$ be a diagonal matrix of scaling factors such that $\mathcal{F}$ multiplied by $X(j)$ results in all entries having more-or-less similar magnitudes. We first obtain scaled versions of $\hat{x}(j)$ and $P^x(j)$ by making

$$\begin{aligned} \hat{x}_{\text{s}}(j) &= \mathcal{F}\hat{x}(j) \\ P^x_{\text{s}}(j) &= \mathcal{F}P^x(j)\mathcal{F}. \end{aligned} \tag{40}$$

Note that this rescaling does not change the physical meaning of $\hat{x}(j)$ and $P^x(j)$ - they are equivalent to a change on the units of measurements. Afterwards, we apply the sigma point generation method to $\hat{x}_{\text{s}}(j)$ and $P^x_{\text{s}}(j)$, generating a set of sigma points $\left\{\mathcal{X}_{\text{s}}(j,m), \mathcal{W}(j,m), \mathcal{W}^P(j,m)\right\}_{m=1}^{N_S}$. Finally, we scale back the sigma points by making

$$\mathcal{X}(j,m) = \mathcal{F}^{-1}\mathcal{X}_{\text{s}}(j,m), m = 1, \ldots, N_S \tag{41}$$

which again, does not change the physical interpretation of the sigma points. In our previous example, if we use this re-scaling scheme by choosing $\mathcal{F} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \cdot 10^{-6} \end{bmatrix}$ we obtain the set of sigma points

$$\left\{ \begin{bmatrix} -0.2536 \\ 4 \cdot 10^5 \end{bmatrix}, \begin{bmatrix} -0.6520 \\ 5.3528 \cdot 10^5 \end{bmatrix}, \begin{bmatrix} -0.9464 \\ 4 \cdot 10^5 \end{bmatrix}, \begin{bmatrix} -0.5480 \\ 2.6472 \cdot 10^5 \end{bmatrix}, \begin{bmatrix} -0.6 \\ 4 \cdot 10^5 \end{bmatrix} \right\} \tag{42}$$

which seems a far more reasonable statistical representation of $\mathcal{N}(x; \hat{x}(j), P^x(j))$.

3.2 Proposed sigma point generation algorithm

1. Perform the scaling

$$\begin{aligned} \hat{x}_{\text{s}}(i) &= \mathcal{F}\hat{x}(i) \\ P^x_{\text{s}}(i) &= \mathcal{F}P^x(i)\mathcal{F}. \end{aligned}$$

   where $\mathcal{F}$ is a diagonal scaling matrix that leads to the parameters and noise terms to have similar magnitudes

2. For $m = 1, \ldots, n_x$, make

$$\begin{aligned} \mathcal{X}_{\text{s}}(i,m) &= \hat{x}_{\text{s}}(i) + \left(\sqrt{(n^x + \lambda)P^x_{\text{s}}(i)}\right)_i \\ \mathcal{X}_{\text{s}}(i,m+n^x) &= \hat{x}_{\text{s}}(i) - \left(\sqrt{(n^x + \lambda)P^x_{\text{s}}(i)}\right)_i \\ \mathcal{W}(i,m+n^x) &= \mathcal{W}(i,m) = \frac{1}{2(n^x + \lambda)} \end{aligned}$$

   where $\lambda = 3 - n_x$

3. Make

$$\begin{aligned} \mathcal{X}_{\text{s}}(i, 2n^x + 1) &= \hat{x}_{\text{s}}(i) \\ \mathcal{W}(i, 2n^x + 1) &= \frac{\lambda}{n^x + \lambda} \\ \mathcal{W}^P(i, 2n^x + 1) &= \mathcal{W}(i, 2n^x + 1) - \alpha^2 + \beta \end{aligned}$$

   where $\alpha = \sqrt{3/n^x}$ and $\beta = 0$

4. For $m = 1, \ldots, 2n_x + 1$, scale back the sigma points by making

$$\mathcal{X}(i,m) = \mathcal{F}^{-1}\mathcal{X}_{\text{s}}(i,m)$$

4 Experimental results

These experimental results, for the problem of stochastic volatility estimation using the Heston model, were obtained using the same simulation setting described in [5]. As it can be seen from Figs. 1, the novel approach seems to outperform both the Liu and West particle filter (from [6], with smoothing parameter $h = 0.1$) and the Monte Carlo RBMPF (the RBMPF presented in [5]) in terms of RMSE parameter estimation. One can verify that, for the parameters ρ and ξ, the MC RBMPF and the LWPF result in RMSE worse than using only the priors of the respective parameters, whereas the Unscented RBMPF results in practically the same errors as the priors.

In terms of NEES performance, from Fig. 2, the Unscented RBMPF seems somewhat to overestimate its own errors for the parameters ρ and ξ, but this may be due to the fact that the errors are also overestimated in the respective priors (again, noting that none of the filters are able to improve upon the priors for these two parameters). Otherwise, the Unscented RBMPF's statistical consistency seems superior to the other two algorithms.

References

[1] S. J. Julier and J. K. Uhlmann, "A new extension of the Kalman filter to nonlinear systems," in *Proc. SPIE Signal Processing, Sensor Fusion, and Target Recognition 'VI*, vol. 3068, Orlando, FL, Apr. 21–24, 1997, pp. 182–193.

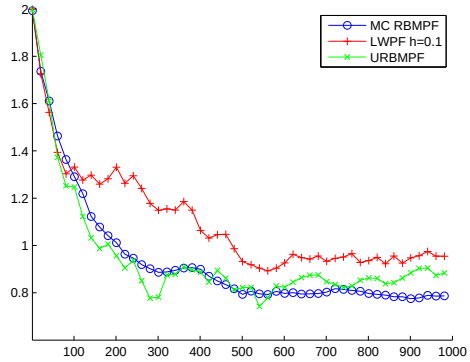
[2] ——, "Simultaneous localisation and map building using split covariance intersection," in *Proc. IROS 2001*, vol. 3, 2001, pp. 1257–1262.

[3] S. J. Julier, "The scaled unscented transformation," in *Proc. American Control Conference*, vol. 6, Anchorage, AK, May 8–10, 2002, pp. 4555–4559.

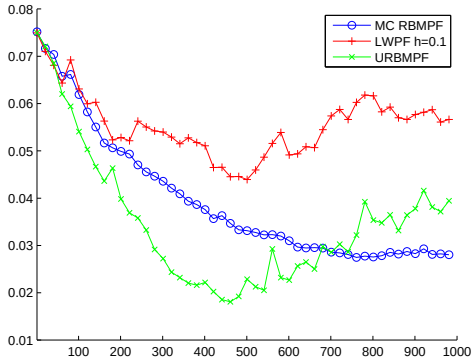
[4] E. H. Aoki, "A general approach for altitude estimation and mitigation of slant range errors on target tracking using 2D radars," in *Proc. 13th International Conference on Information Fusion*, Edinburgh, UK, Jun. 26–29, 2010.

[5] E. H. Aoki, P. K. Mandal, A. Bagchi, and Y. Boers, "The Rao-Blackwellized marginal particle filter: a general solution to the joint state and parameter estimation problem," *J. Royal Stat. Soc. B*, 2013.

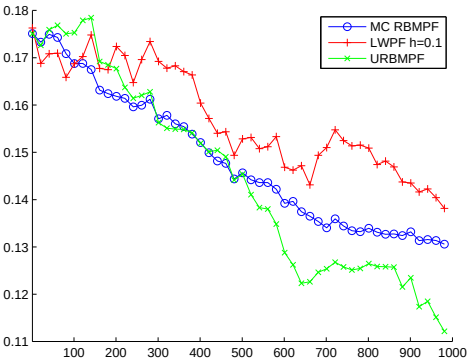
[6] J. Liu and M. West, "Combined parameters and state estimation in simulation-based filtering," in *Sequential Monte Carlo Methods in Practice*, A. Doucet, N. de Freitas, and N. Gordon, Eds. New York: Springer-Verlag, 2001, pp. 197–223.



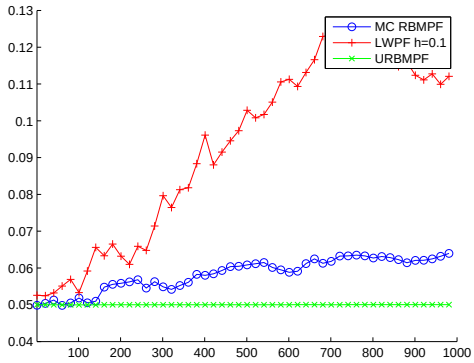
(a) κ



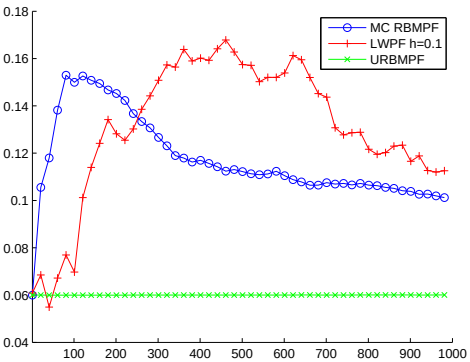
(b) κ'



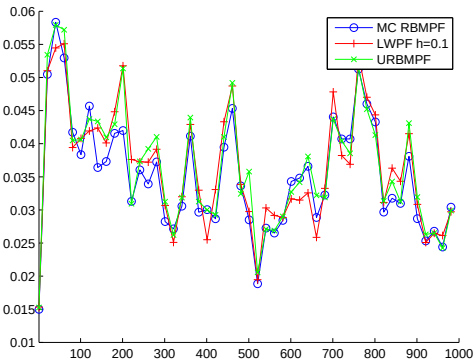
(c) μ_S



(d) ρ

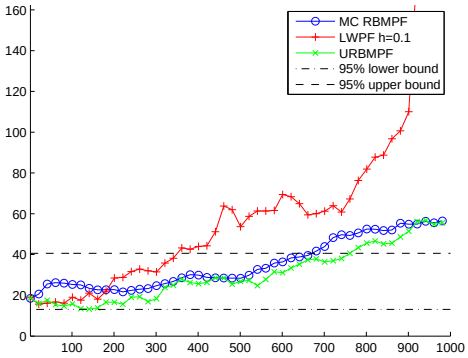


(e) ξ

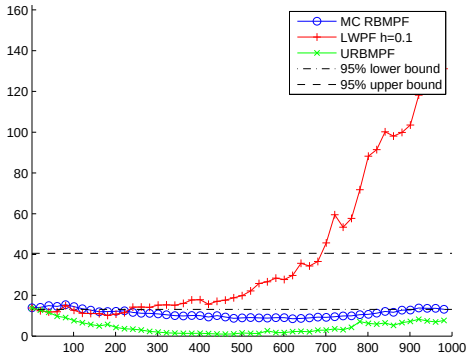


(f) s_k

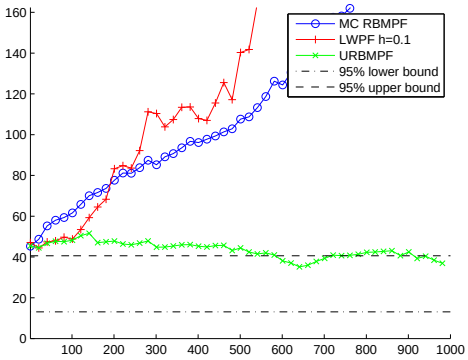
Figure 1: RMSE for the state and parameters



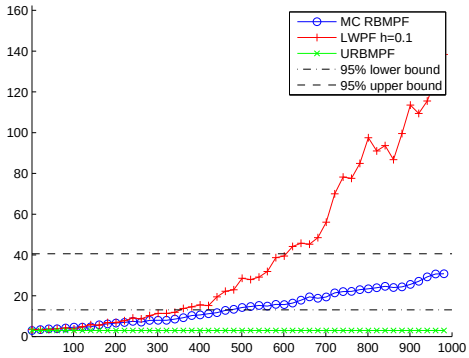
(a) κ



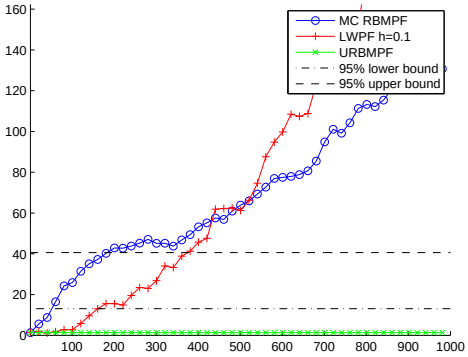
(b) κ'



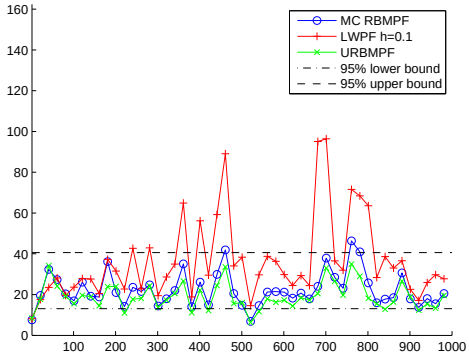
(c) μ_S



(d) ρ



(e) ξ



(f) s_k

Figure 2: NEES for the state and parameters