

Computation and visualization of posterior densities in scalar nonlinear and non-Gaussian Bayesian filtering and smoothing problems

Michael Roth and Fredrik Gustafsson

Conference Publication

N.B.: When citing this work, cite the original article.

Part of: 2017 IEEE INTERNATIONAL CONFERENCE ON ACOUSTICS, SPEECH AND SIGNAL PROCESSING (ICASSP), 2017, pp. 4686-4690. ISBN: 978-1-5090-4117-6

Book Series: International Conference on Acoustics Speech and Signal Processing ICASSP, 1520-6149.

DOI: <https://doi.org/10.1109/ICASSP.2017.7953045>

Copyright: IEEE

©2017 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

Available at: Linköping University Institutional Repository (DiVA)

<http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-144282>



COMPUTATION AND VISUALIZATION OF POSTERIOR DENSITIES IN SCALAR NONLINEAR AND NON-GAUSSIAN BAYESIAN FILTERING AND SMOOTHING PROBLEMS

Michael Roth, Fredrik Gustafsson

Department of Electrical Engineering, Linköping University, Sweden
e-mail: michael.roth@liu.se, fredrik.gustafsson@liu.se

ABSTRACT

One-dimensional Bayesian filtering and smoothing problems can be solved numerically using a number of algorithms, even in nonlinear and non-Gaussian cases. In this educational paper we advocate for the benefits of visualizing the obtained posterior densities as complement to, e.g., estimation error analysis. In addition to a review of Bayesian filtering and smoothing and the respective point mass and particle solutions, we devise a novel algorithm for filtering when the likelihood cannot be evaluated. Several instructive examples are discussed and easily adjustable matlab code is provided as complement to this paper.

Index Terms— Bayesian filtering, smoothing, point mass filter, particle filter.

1. INTRODUCTION

Bayesian state estimation [1] is an established and flexible signal processing framework that can accommodate many real world problems such as target tracking and navigation [2], audio restoration [3], speech processing [4], and processing of financial data [5]. The main idea is to obtain a probabilistic description of a state x_k by (sequentially) processing measurements $y_{1:l} = \{y_1, \dots, y_l\}$, where the temporal evolution of the state and its relation to the measurement are described by a state space model

$$x_{k+1} = f(x_k, v_k), \quad y_k = h(x_k, e_k). \quad (1)$$

The Bayesian framework [1, 6, 7, 8] asserts that the involved process and measurement noise, v_k and e_k , respectively, and the initial state x_0 are random variables with a known distribution. Consequently, x_k and y_k are random variables for all k and the estimation task amounts to finding a conditional distribution of x_k given a sequence of measurements $y_{1:l}$.

In this paper we address the marginal filtering and smoothing problems. That is, our aim is to recover the conditional probability density function $p(x_k | y_{1:l})$ with $l = k$ for filtering and $l > k$ for smoothing problems [9]. The focus on the scalar case allows us to circumvent the fact that it is

not feasible to compute $p(x_k | y_{1:l})$ analytically except for a few special cases: The linear Gaussian case is solved by the Kalman filter [6, 10], and a solution exists for discrete x_k that only assume a finite number of values [11]. We exploit the latter result by restricting the continuous x_k to a grid $\{x^1, \dots, x^M\}$, which is a viable option for the scalar case. Point mass [12] and particle [13] methods and a novel algorithm are described to obtain $p(x_k | y_{1:l})$ at the grid points. Here, the novel algorithm solves the filtering problems for models that do not allow evaluation of the likelihood, and hence prohibit an application of the point mass and particle methods.

In addition to the novel filter, our contributions are educational. The basis of Bayesian state estimation as operations on probability density functions is often overlooked in favor of advanced solution concepts, although its understanding can be used to quickly suggest viable solutions. Moreover, visualization of $p(x_k | y_{1:l})$ can facilitate a much deeper understanding of the problem at hand than analysis of the mean squared estimation error. For instance, it can be used to pick reasonable estimates (e.g., maximum a posteriori vs. minimum variance [9]), and is useful to assess how promising other approximate filters [14] are. For teaching purposes, easily adjustable matlab code is available for download¹.

Our paper can be viewed as complement to [1] in that it highlights how $p(x_k | y_{1:l})$ can be computed and visualized for the scalar case. Although algorithmic tools to achieve this are mostly known [12, 13, 8], their use to visualize densities and gain insights appeared secondary. Recent approaches to address the filtering problem with intractable likelihood functions include [15, 16], but go beyond the novel approach that is suggested in Sec. 4.3.

2. BAYESIAN STATE ESTIMATION PROBLEMS

In order to form a stochastic state space model, the state difference and measurement equation of (1) must be supplemented with a probabilistic description of the initial state and the noise. One option would be a joint density²

¹<http://users.isy.liu.se/en/rt/roth/>

²This general formulation also covers the case of known dependencies between x_0 , $v_{0:k}$, and $e_{1:k}$.

This work was supported by the project Scalable Kalman Filters granted by the Swedish Research Council (VR).

$q(x_0, v_{0:k}, e_{1:k})$. If we further assume that there exists a one-to-one mapping $(x_0, v_{0:k}, e_{1:k}) = \psi(x_{0:k}, y_{1:k})$, then

$$p(x_{0:k}, y_{1:k}) = q(\psi(x_{0:k}, y_{1:k})) |\det(D\psi(x_{0:k}, y_{1:k}))| \quad (2a)$$

follows from a transformation theorem [17] of random variables. The extra factor includes the Jacobian of the variable change relation $D\psi(x_{0:k}, y_{1:k})$. A related result has been noted in [6]. From (2a) we can obtain the filtering distribution via marginalization and conditioning

$$p(x_k, y_{1:k}) = \int p(x_{0:k}, y_{1:k}) dx_{0:k-1}, \quad (2b)$$

$$p(x_k | y_{1:k}) = \frac{p(x_k, y_{1:k})}{p(y_{1:k})}. \quad (2c)$$

In fact, the order of operations can be exchanged in many ways. Similar considerations yield a solution for the smoothing and prediction densities $p(x_k | y_{1:l})$. Also, a recursive filtering solution for independent noise can be obtained in this way. Transformation and marginalization operations yield $p(x_k, y_k | y_{1:k-1})$ from $p(x_{k-1} | y_{1:k-1})$, $p(v_{k-1})$, and $p(e_k)$. A subsequent conditioning on y_k completes the recursion. For smoothing, the fact that past states need to be kept makes the problem more challenging in this framework. The important point of the above paragraph, however, is that Bayesian filtering and smoothing can be expressed in terms of basic operations on probability density functions.

The filtering literature [1] often states the model in terms of the transition density and the likelihood function

$$x_{k+1} \sim p(x_{k+1} | x_k), \quad y_k \sim p(y_k | x_k). \quad (3)$$

For independent white and additive noise in (1), i.e., $x_{k+1} = f(x_k) + v_k$, (3) is easily obtained. Moreover, the aforementioned case implies certain conditional independence properties that are used in the derivation of the recursive Bayesian filtering solution [1, 7]

$$p(x_{k+1} | y_{1:k}) = \int p(x_{k+1} | x_k) p(x_k | y_{1:k}) dx_k, \quad (4a)$$

$$p(x_k | y_{1:k}) = \frac{p(y_k | x_k) p(x_k | y_{1:k-1})}{p(y_{1:k-1})}, \quad (4b)$$

with a normalization constant $p(y_{1:k-1})$. Slightly more involved, a backwards recursion for the smoothing density [1, 8] can be obtained. For $l > k$, the filtering and prediction results are processed in

$$\begin{aligned} & p(x_k | y_{1:l}) \\ &= p(x_k | y_{1:k}) \int \frac{p(x_{k+1} | x_k) p(x_{k+1} | y_{1:l})}{p(x_{k+1} | y_{1:k-1})} dx_{k+1}. \end{aligned} \quad (5)$$

3. THE ALGORITHMIC TREATMENT OF ONE-DIMENSIONAL DENSITIES

We here discuss algorithmic aspects of working with one-dimensional densities that are relevant for Sec. 4.

3.1. Sampled densities on a grid

Assume a random $x \in \mathbb{R}$ with continuous density $p(x)$ and its probability mass within a known interval, i.e., $p(x) = 0$ outside that interval. A reasonable representation of $p(x)$ in a digital computer can be achieved by storing $p(x^i)$ for grid points x^i , $i = 1, \dots, M$. For sufficiently dense uniformly spaced grids with increments Δ we can associate a point mass $p(x^i)\Delta$ to each x^i . Expected values simplify to finite sums $E\{f(x)\} = \sum_{i=1}^M f(x^i)p(x^i)\Delta$. The computations of Sec. 4 often yield densities that do not integrate to one. Normalization then ensures $\sum_{i=1}^M p(x^i)\Delta = 1$.

3.2. Random samples from sampled densities

Given uniformly spaced x^i and $p(x^i)$, realizations of x can be generated via the inversion method [18, 11], which uses the cumulative distribution function and uniform random numbers. Below is a simple algorithm that furthermore adds noise to obtain values in between grid points.

- 1: Sample $u^{(i)} \sim \mathcal{U}(0, 1)$.
- 2: Pick the largest x^i such that $\sum_{j=1}^i p(x^j)\Delta < u^{(i)}$.
- 3: Add uniform noise $\sim \mathcal{U}(-\frac{\Delta}{2}, \frac{\Delta}{2})$ to form $x^{(i)}$.

There is a close relation to the resampling algorithms in particle filters [19].

3.3. From samples to densities

Kernel density estimates [11], or KDEs, can approximate $p(x^j)$ given random samples $x^{(i)}$ and weights $w^{(i)}$, $i = 1, \dots, N$, according to $\hat{p}(x^j) = \sum_{i=1}^N w^{(i)} \mathcal{N}(x^j; x^{(i)}, \sigma^2)$. Here, σ^2 is the variance of the Gaussian kernels. The alternative use of rectangular kernels discloses a relationship to histograms. The concept can be applied to vector x with the only change that a corresponding multivariate Gaussian density with diagonal covariance is used. This is exploited in Sec. 4.3.

4. FILTERING AND SMOOTHING ALGORITHMS FOR THE ONE-DIMENSIONAL CASE

In this section we review two existing and devise one novel algorithm for the Bayesian filtering and smoothing problem. The description is tailored to the one-dimensional case with the intention of being easy to implement.

4.1. Point mass filters and smoothers

Point mass approaches [12] evaluate (4) and (5) for a finite number of grid points. Hence, they apply to models that are formulated in terms of (3).

The first point mass filter was presented in [20]. Later, advanced versions were discussed in [8, 21]. The first related smoother was presented in [8], in addition to the theoretical

result (5). Instead of mere point masses, however, piecewise linear density approximations were employed. Both [20, 21] discuss how to best choose the grid values at each time step, and how to rotate rectangular grids in higher dimensions. For the one-dimensional case, however, we can skip such considerations and focus on straightforward approximation of (4) and (5).

Below is a filtering/smoothing algorithm. The only user input is the choice of a grid as in Sec. 3.1.

- 1: Evaluate $p(x_0)$ for all M grid values x^i to get $p(x_0^i)$.
- 2: **for** $k = 1, 2, \dots$ **do**
- 3: Evaluate $p(x_k^i | x_{k-1}^j)$ for all M^2 x_k^i and x_{k-1}^j .
- 4: Multiply $p(x_k^i | x_{k-1}^j)$ by $p(x_{k-1}^j | y_{1:k-1})$ to obtain the joint density $p(x_k^i, x_{k-1}^j | y_{1:k-1})$.
- 5: Marginalize by summing $p(x_k^i | y_{1:k-1}) \propto \sum_{j=1}^M p(x_k^i, x_{k-1}^j | y_{1:k-1})$ and normalize to obtain the prediction density $p(x_k^i | y_{1:k-1})$.
- 6: Evaluate $p(y_k | x_k^i)$ for $x_k^i, i = 1, \dots, M$.
- 7: Multiply $p(x_k^i | y_{1:k-1})p(y_k | x_k^i)$ and normalize to obtain the filtering density $p(x_k^i | y_{1:k})$.
- 8: **end for**
- 9: For $k = l$ set smoothing result to filtering result.
- 10: **for** $k = l - 1, l - 2, \dots, 0$ **do**
- 11: Evaluate $p(x_{k+1}^j | x_k^i)$ for all M^2 x_{k+1}^j and x_k^i .
- 12: Multiply $p(x_{k+1}^j | x_k^i)$ by $\frac{p(x_{k+1}^j | y_{1:l})}{p(x_{k+1}^j | y_{1:k-1})}$.
- 13: Marginalize x_{k+1} by summing over all j .
- 14: Multiply by $p(x_k^i | y_{1:k})$ and normalize to obtain the smoothing density $p(x_k^i | y_{1:l})$.
- 15: **end for**

Both smoothing and filtering require M^2 evaluations of the transition density, which dominates the cost.

4.2. Particle filters and smoothers

Particle filters and smoothers employ sequential importance sampling to approximate the Bayesian filtering and smoothing recursions (4) and (5) for arbitrary nonlinear and non-Gaussian models. The algorithms yield a set of N samples $x_k^{(i)}$ and weights $w_{k|1:l}^{(i)}$ that represent $p(x_k | y_{1:l})$. These can be illustrated using kernel density estimates.

Because particle methods are well established, we restrict our presentation to a short literature review. Early attempts to Monte Carlo filtering date back to [22]. However, the first operational particle filter [23] that introduced a crucial resampling step was introduced much later. Different smoothing approaches are suggested in [24, 25]. Convergence results [13] state that (4) and (5) can be recovered as $N \rightarrow \infty$. Whereas filtering enjoys linear complexity in the number of particles N , in contrast to the point mass approach, basic smoothing approaches scale with N^2 .

4.3. A random sampling based filtering algorithm that does not require evaluation of the likelihood

The filtering approaches of the previous sections require the model formulated in terms of (3), or at least evaluation of the likelihood $p(y_k | x_k)$. However, this is not possible for many models of the form (1). One such example is treated in Sec. 5.3.

We here propose a solution that uses random sampling and kernel density estimates as discussed in Sec. 3. The user determines a grid by choosing M point x^i , the number of random samples N , and variances for two-dimensional kernel density estimation in x_k and y_k .

- 1: **for** $k = 1, 2, \dots$ **do**
- 2: Sample $x_{k-1}^{(i)}$ and $v_{k-1}^{(i)}, i = 1, \dots, N$, and compute $x_{k|k-1}^{(i)} = f(x_{k-1}^{(i)}, v_{k-1}^{(i)})$.
- 3: Compute a KDE from the $x_{k|k-1}^{(i)}$ to get the sampled prediction density $p(x_k^j | y_{1:k-1}), j = 1, \dots, M$.
- 4: Sample $e_k^{(i)}, i = 1, \dots, N$, and evaluate $y_k^{(i)} = h(x_{k|k-1}^{(i)}, e_k^{(i)})$.
- 5: Compute a two-dimensional KDE from $(x_{k|k-1}^{(i)}, y_k^{(i)})$ to obtain $p(x_k^j, y_k | y_{1:k-1})$ for $j = 1, \dots, M$. Normalize to obtain $p(x_k^j | y_{1:k})$.
- 6: **end for**

The novelty of the above algorithm is that the filtering result is computed as a KDE for all grid points x_k^j , but only the actual measurement y_k . The computations can be sped up by computing only those kernels that actually contribute to $p(x_k^j | y_{1:k})$, i.e., by only processing $y_k^{(i)}$ close to y_k .

5. EXAMPLES

Three examples are considered in greater detail.

5.1. “The particle filter example”

First, we discuss an often used benchmark problem [8, 19, 23]. The states and measurements are governed by

$$x_{k+1} = \frac{x_k}{2} + 25 \frac{x_k}{1 + x_k^2} + 8 \cos(1.2(k+1)) + v_k, \quad (6a)$$

$$y_k = \frac{x_k^2}{20} + e_k, \quad (6b)$$

with independent $v_k \sim \mathcal{N}(0, 10)$, $e_k \sim \mathcal{N}(0, 1)$, and $x_0 \sim \mathcal{N}(0, 1)$. The filtering and smoothing results using the point mass approach are shown in Fig. 1. Also illustrated is $\pm\sqrt{20}y_k$, a mapping of the measurement under the assumption $e_k = 0$. It can be seen that the prediction and filtering densities are bi-modal, and hence that the mean as estimate would not be a good choice [9]. In the smoothing

result, only the mode close to the measurement persists. Particle methods and the novel filter yield similar results with sufficiently many samples N .

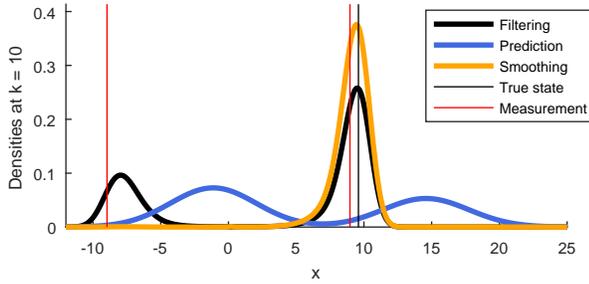


Fig. 1. Prediction, filtering, and smoothing densities for the “particle filter example”.

5.2. Student’s t random walk in noise

Second, a random walk that is observed in heavy-tailed noise is considered. The model is given by

$$x_{k+1} = x_k + v_k, \quad y_k = x_k + e_k, \quad (7)$$

with $v_k \sim \text{St}(0, 1, 3)$, $e_k \sim \text{St}(0, 1, 3)$, and $x_0 \sim \text{St}(0, 1, 3)$.

The system is linear and the prediction, filtering, and smoothing densities are unimodal most of the times. However, several modes can be spawned when the likelihood and the prediction density are in conflict, as shown in Fig. 2 using the point mass approach. This behavior has been described as “moment of indecision” [26]. The smoothing result is again unimodal.

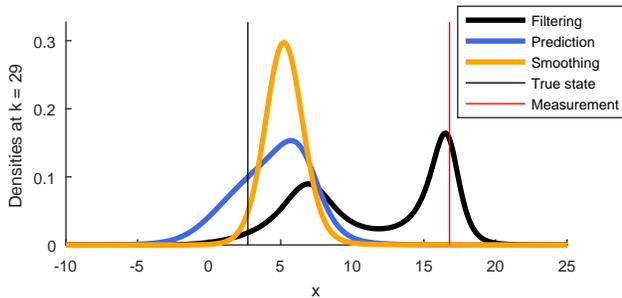


Fig. 2. Prediction, filtering, and smoothing densities for the Student’s t random walk in noise.

Due to the heavy tails of the involved signals, particle methods and the novel filter face difficulties for cases such as the illustrated “moment of indecision”. Still, the results are mostly similar except for the variation due to sampling.

5.3. Saturated measurements

The last example considers the model

$$x_{k+1} = 0.7x_k + v_k, \quad y_k = \text{sat}(x_k + e_k), \quad (8)$$

with $v_k \sim \mathcal{N}(0, 1)$, $e_k \sim \mathcal{N}(0, 0.5)$, and $x_0 \sim \mathcal{N}(0, 0.1)$. The sat function truncates its input beyond -1.5 and 1.5 . Hence, the measurement model reflects the realistic case of a sensor with a limited range, e.g., a clipping accelerometer or microphone. One realization of a trajectory and the measurements is given in Fig. 3.

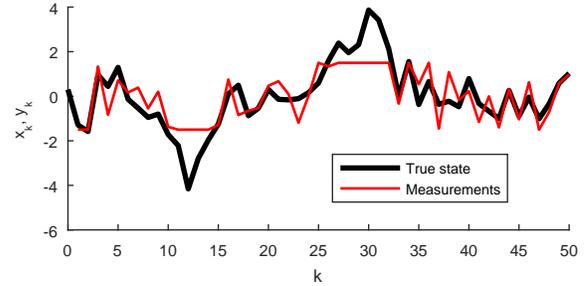


Fig. 3. Trajectory of the saturated measurements example.

Filtering for (8) is difficult because the likelihood cannot be obtained easily for the many-to-one saturation function. Hence, the only applicable approach is the filter of Sec. 4.3. Its output for a saturated measurement is shown in Fig 4. The filtering density keeps most of its probability mass beyond the saturation interval. Clearly, a Kalman filter unaware of the saturation would yield a Gaussian that is centered between the predicted mode and the measurement.

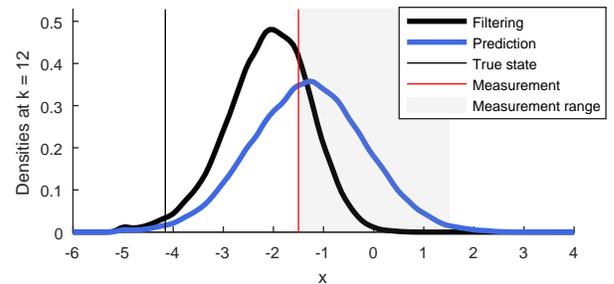


Fig. 4. Prediction and filtering densities for the saturated measurements example. The shaded area indicates the measurement range.

6. CONCLUDING REMARKS

In this paper we have advocated for the computation and visualization of Bayesian filtering and smoothing densities, and showed how this can be achieved using a range of algorithms. Furthermore, a simple novel filter was suggested for the general state space models that neither requires the transition density nor the likelihood of the model to be executed.

7. REFERENCES

- [1] S. Särkkä, *Bayesian Filtering and Smoothing*. New York: Cambridge University Press, Oct. 2013.
- [2] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P.-J. Nordlund, "Particle filters for positioning, navigation, and tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 425–437, 2002.
- [3] S. J. Godsill and P. J. W. Rayner, *Digital Audio Restoration*. London; New York: Springer, Oct. 1998.
- [4] C. Glaser, M. Heckmann, F. Joublin, and C. Goerick, "Combining auditory preprocessing and Bayesian estimation for robust formant tracking," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 2, pp. 224–236, Feb. 2010.
- [5] P. Date and K. Ponomareva, "Linear and non-linear filtering in mathematical finance: A review," *IMA Journal of Management Mathematics*, vol. 22, no. 3, pp. 195–211, Jul. 2011.
- [6] Y.-C. Ho and R. Lee, "A Bayesian approach to problems in stochastic estimation and control," *IEEE Transactions on Automatic Control*, vol. 9, no. 4, pp. 333–339, 1964.
- [7] A. H. Jazwinski, *Stochastic Processes and Filtering Theory*. Academic Press, Mar. 1970.
- [8] G. Kitagawa, "Non-Gaussian state-space modeling of nonstationary time series," *Journal of the American Statistical Association*, vol. 82, no. 400, pp. 1032–1041, Dec. 1987.
- [9] B. D. Anderson and J. B. Moore, *Optimal Filtering*. Prentice Hall, Jun. 1979.
- [10] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of basic Engineering*, vol. 82, no. 1, pp. 35–45, Mar. 1960.
- [11] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, Aug. 2006.
- [12] H. W. Sorenson, "On the development of practical non-linear filters," *Information Sciences*, vol. 7, pp. 253–270, 1974.
- [13] A. Doucet and A. M. Johansen, "A tutorial on particle filtering and smoothing: fifteen years later," in *The Oxford Handbook of Nonlinear Filtering*, D. Crisan and B. Rozovskii, Eds., 2011, pp. 656–704.
- [14] M. Roth, G. Hendeby, and F. Gustafsson, "Nonlinear Kalman filters explained: A tutorial on moment computations and sigma point methods," *Journal of Advances in Information Fusion*, vol. 11, no. 1, pp. 47–70, Jun. 2016.
- [15] A. Jasra, S. S. Singh, J. S. Martin, and E. McCoy, "Filtering via approximate Bayesian computation," *Statistics and Computing*, vol. 22, no. 6, pp. 1223–1237, May 2010.
- [16] F. Septier, G. W. Peters, and I. Nevat, "Bayesian filtering with intractable likelihood using sequential MCMC," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2013, pp. 6313–6317.
- [17] A. Gut, *An Intermediate Course in Probability*, 2nd ed. Springer, Jun. 2009.
- [18] L. Devroye, *Non-Uniform Random Variate Generation*. New York: Springer, 1986.
- [19] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, Feb. 2002.
- [20] R. Bucy and K. Senne, "Digital synthesis of non-linear filters," *Automatica*, vol. 7, no. 3, pp. 287–298, May 1971.
- [21] S. C. Kramer and H. W. Sorenson, "Recursive Bayesian estimation using piece-wise constant approximations," *Automatica*, vol. 24, no. 6, pp. 789–801, Nov. 1988.
- [22] J. Handschin, "Monte Carlo techniques for prediction and filtering of non-linear stochastic processes," *Automatica*, vol. 6, no. 4, pp. 555–563, Jul. 1970.
- [23] N. J. Gordon, D. J. Salmond, and A. F. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *Radar and Signal Processing, IEE Proceedings F*, vol. 140, no. 2, pp. 107–113, Apr. 1993.
- [24] A. Doucet, S. Godsill, and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," *Statistics and Computing*, vol. 10, no. 3, pp. 197–208, 2000.
- [25] S. J. Godsill, A. Doucet, and M. West, "Monte Carlo smoothing for nonlinear time series," *Journal of the American Statistical Association*, vol. 99, no. 465, pp. 156–168, 2004.
- [26] G. Agamennoni, J. I. Nieto, and E. M. Nebot, "Approximate inference in state-space models with heavy-tailed noise," *IEEE Transactions on Signal Processing*, vol. 60, no. 10, pp. 5024–5037, Oct. 2012.