

A Two Filter Particle Smoother for Wiener State-Space Systems

Roland Hostettler

This is a post-print of a paper published in *IEEE Conference on Control Applications*. When citing this work, you must always cite the original article:

R. Hostettler, "A two filter particle smoother for Wiener state-space systems," in *Control Applications, 2015 IEEE Conference on*, Sydney, Australia, September 2015

DOI:

10.1109/CCA.2015.7320664

Copyright:

© 2016 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

A Two Filter Particle Smoother for Wiener State-Space Systems

Roland Hostettler¹

Abstract—In this article, a two filter particle smoothing algorithm for Wiener state-space systems is proposed. The smoother is obtained by exploiting the model structure. This leads to a suitable proposal density for the backward filter inherent in the problem instead of introducing an artificial one. Numerical examples are provided in order to illustrate the proposed algorithm’s performance and to compare it to current state of the art smoothers from the literature. It is found that the proposed method yields comparable results with less computational complexity as backward simulation-based particle smoothing algorithms.

I. INTRODUCTION

Bayesian filtering and smoothing are two closely related and important tasks in many different applications ranging from state estimation for control in feedback systems to target tracking in marine, space, or vehicular applications. The filtering problem is concerned with inferring the state x_t (where the subscript $t = 1, \dots, T$ denotes a discrete time instant) given a set of data up to t , $y_{1:t} = \{y_1, y_2, \dots, y_t\}$, that is, we are interested in finding the (marginal) filtering density $p(x_t|y_{1:t})$. The problem of smoothing is closely related but here we are given a set of data $y_{1:T}$ where $1 \leq t \leq T$, and we are interested in the marginal smoothing density $p(x_t|y_{1:T})$ instead. Because of their importance, they have been studied extensively, see, for example, [2]. It is well known that analytical solutions can only be found for a few special cases such as linear systems with Gaussian process and measurement noises, where the resulting algorithms are the Kalman-Bucy filter [3] and the Rauch-Tung-Striebel smoother [4]. Otherwise, we have to resort to some kind of approximative method, for example unscented Kalman filters [5], unscented Rauch-Tung-Striebel smoothers [6], or sequential Monte Carlo (SMC) methods [7]-[9]. SMC methods have proven to be very useful and popular, partly due to the cheaply available computational power of today. While filtering can, in many cases, be done efficiently, smoothing is somewhat more complicated.

Roughly, SMC state smoothing algorithms can be divided into two categories: (a) Forward filtering backward smoothing algorithms that use a particle filter in the forward pass and refine the particles by resampling or reweighing in the backward pass, and (b) two filter smoothers that run two individual filters (one that iterates through the data in the forward direction and one that iterates in the backward direction) and combine the information from the two filters [10]. In both methods, however, a few practical problems arise. For

the forward filtering backward smoothing algorithms, one of these problems is the fact that the computational cost can increase quadratically with the number of particles, which is very impractical. Even though solutions that (asymptotically) achieve linear computational cost have been proposed [11]-[13], these methods might not be feasible in practice due to numerical issues or high rejection rates. Other approaches try to exploit the inherent structure of the system in order to reduce the dimensionality of the problem, for example by using Rao-Blackwellization [14], [15]. When using two filter smoothers on the other hand, it is not quite obvious how to address the backward filtering pass and different solutions have been proposed. Briers et. al. [16] introduced a thorough analysis of two filter smoothing using SMC. There, an additional artificial proposal density was introduced in order to run the backward filter. This approach was further extended in [17] where it was shown that it is valid for a broader class of systems than originally thought.

In this paper, we consider systems with linear, Gaussian process dynamics and non-linear measurements, that is, Wiener state-space systems of the form

$$x_t = A(t)x_{t-1} + v_t \quad (1a)$$

$$y_t = g(x_t, e_t, t). \quad (1b)$$

Here, $x_t \in \mathbb{R}^{N_x}$ is the state at the discrete time t , $A(t) \in \mathbb{R}^{N_x \times N_x}$ is the state transition matrix (possibly time-varying), and $v_t \sim \mathcal{N}(0, Q(t))$ is the process noise. (Note that for brevity, the explicit dependence of $A(t)$ and $Q(t)$ on t will be discarded for the remainder of this paper.) In (1b), $y_t \in \mathbb{R}^{N_y}$ is the measurement at time t , $g(\cdot)$ is the non-linear measurement function, and $e_t \sim p(e_t)$ is the measurement noise. Furthermore, we assume that the initial state x_0 is a Gaussian random variable according to

$$p(x_0) = \mathcal{N}(x_0; \mu_0, \Sigma_0). \quad (2)$$

The main contribution of this paper is an explicit formulation of a bootstrap filter-based two filter smoother addressing the system in (1). The resulting smoother is closely related to the work in [16]-[17] but here, we show how a formulation of the backward filter naturally arises by exploiting the model structure in (1) instead of introducing an artificial prior distribution of the state (Section II). One of the biggest advantages of the proposed smoother is the fact that the model structure will allow for efficient implementation. The properties of the proposed smoother are illustrated using two simulation examples in Section III.

¹Roland Hostettler is with the Division of Signals and Systems, Department of Computer Science, Electrical and Space Engineering, Luleå University of Technology, Luleå, Sweden. roland.hostettler@ltu.se

II. METHOD

In this section, we derive the proposed two filter particle smoother. First, note that the marginal smoothing density can be factorized as

$$\begin{aligned} p(x_t|y_{1:T}) &= p(x_t|y_{1:t-1}, y_{t:T}) \\ &= \frac{p(x_t|y_{1:t-1})p(y_{t:T}|x_t)}{p(y_{t:T}|y_{1:t-1})} \end{aligned} \quad (3)$$

which is the standard formulation for the two filter smoother [18]. However, we do not stop here and apply Bayes' rule a second time in order to obtain

$$\begin{aligned} p(x_t|y_{1:T}) &= \frac{p(x_t|y_{1:t-1})p(x_t|y_{t:T})p(y_{t:T})}{p(y_{t:T}|y_{1:t-1})p(x_t)} \\ &\propto \frac{p(x_t|y_{1:t-1})}{p(x_t)}p(x_t|y_{t:T}). \end{aligned} \quad (4)$$

Note that the measurement y_t was included in the backward part instead of the forward part here. It turns out that this factorization makes it possible to use a regular particle filter that targets the factor $p(x_t|y_{1:t-1})$ and a backward particle filter targeting $p(x_t|y_{t:T})$. Each of the two filters is introduced in the sections below, which is then followed by the formulation of the smoother itself.

A. Forward Filter

The numerator of the first factor, $p(x_t|y_{1:t-1})$, in (4) can easily be found from a regular particle filter targeting $p(x_{1:t}|y_{1:t})$. A bootstrap sampling importance resampling (SIR) particle filter can be used to obtain the following approximation of the joint filtering distribution (see [7] or [8] for detailed introductions to particle filtering)

$$p(x_{1:t}|y_{1:t}) \approx \sum_{n=1}^M w_{t|t}^{(n)} \delta(x_{1:t} - x_{1:t|t}^{(n)}) \quad (5)$$

where $\delta(x)$ is the Dirac-delta function of x , $x_{1:t|t}^{(n)}$ is the n th state trajectory from 1 up to t given the measurements up to time t , and $w_{t|t}^{(n)}$ is the importance weight of the corresponding trajectory at time t . In the bootstrap particle filter, the particles are propagated using the state transition density according to

$$x_{t|t}^{(n)} \sim p(x_t|x_{t-1}^{(n)}) \quad (6)$$

which can readily be found from (1a) to be

$$p(x_t|x_{t-1}) = \mathcal{N}(x_t; Ax_{t-1}, Q). \quad (7)$$

The particle weights are given by

$$\begin{aligned} \tilde{w}_{t|t}^{(n)} &= w_{t-1|t-1}^{(n)} p(y_t|x_t^{(n)}) \\ w_{t|t}^{(n)} &= \frac{\tilde{w}_{t|t}^{(n)}}{\sum_{n=1}^M \tilde{w}_{t|t}^{(n)}}. \end{aligned} \quad (8)$$

After calculating the weights, a resampling step might be used if the concentration of particles in the interesting area of the state space is low. By doing so, trajectories with low importance weight are replaced by trajectories with high importance weight instead.

Next, given the approximation (5) and the state transition density (7) an approximation for the one step ahead prediction $p(x_t|y_{1:t-1})$ can easily be obtained as

$$\begin{aligned} p(x_t|y_{1:t-1}) &= \int_{-\infty}^{\infty} p(x_t|x_{t-1}) \\ &\quad \times p(x_{1:t-1}|y_{1:t-1}) dx_{1:t-1} \\ &\approx \int_{-\infty}^{\infty} p(x_t|x_{t-1}) \sum_{m=1}^M w_{t-1|t-1}^{(m)} \\ &\quad \times \delta(x_{1:t-1} - x_{t-1|t-1}^{(m)}) dx_{1:t-1} \\ &= \sum_{m=1}^M w_{t-1|t-1}^{(m)} p(x_t|x_{t-1}^{(m)}) \\ &= \sum_{m=1}^M w_{t-1|t-1}^{(m)} \mathcal{N}(x_t; Ax_{t-1}^{(m)}, Q) \end{aligned} \quad (9)$$

Furthermore, the denominator $p(x_t)$ in (4) can be found as

$$\begin{aligned} p(x_t) &= \int_{-\infty}^{\infty} p(x_t, x_{t-1}) dx_{t-1} \\ &= \int_{-\infty}^{\infty} p(x_t|x_{t-1})p(x_{t-1}) dx_{t-1}. \end{aligned} \quad (10)$$

It is straight forward to see that (10) is a recursive expression and since both $p(x_0)$ and $p(x_t|x_{t-1})$ are Gaussian, $p(x_t)$ will be Gaussian too according to

$$p(x_t) = \mathcal{N}(x_t; \mu_t, \Sigma_t) \quad (11)$$

with

$$\mu_t = A\mu_{t-1} \quad (12a)$$

$$\Sigma_t = Q + A\Sigma_{t-1}A^T. \quad (12b)$$

Putting together (4), (9), and (11)-(12) yields

$$\frac{p(x_t|y_{1:t-1})}{p(x_t)} \approx \frac{\sum_{m=1}^M w_{t-1|t-1}^{(m)} \mathcal{N}(x_t; Ax_{t-1}^{(m)}, Q)}{\mathcal{N}(x_t; \mu_t, \Sigma_t)} \quad (13)$$

for the first term in (4).

B. Backward Filter

The term $p(x_t|y_{t:T})$ can be seen as a marginal backward filtering density which can be found as follows. Let $p(x_{t:T}|y_{t:T})$ be the joint backward filtering density. Then,

$$p(x_t|y_{t:T}) = \int_{-\infty}^{\infty} p(x_{t:T}|y_{t:T}) dx_{t+1:T}. \quad (14)$$

Furthermore, we can express the joint backward density as

$$\begin{aligned} p(x_{t:T}|y_{t:T}) &= \frac{p(y_t|x_{t:T}, y_{t+1:T})p(x_{t:T}|y_{t+1:T})}{p(y_t|y_{t+1:T})} \\ &\propto p(y_t|x_t)p(x_{t:T}|y_{t+1:T}) \\ &= p(y_t|x_t)p(x_t|x_{t+1:T}, y_{t+1:T}) \\ &\quad \times p(x_{t+1:T}|y_{t+1:T}) \\ &= p(y_t|x_t)p(x_t|x_{t+1}) \\ &\quad \times p(x_{t+1:T}|y_{t+1:T}). \end{aligned} \quad (15)$$

In (15), $p(y_t|x_t)$ is the likelihood (which is given through the model). The inverse process dynamics $p(x_t|x_{t+1})$ are key in the development of the proposed smoother. Since the state dynamics are linear and Gaussian, we can use

$$p(x_t|x_{t+1}) = \frac{p(x_{t+1}|x_t)p(x_t)}{p(x_{t+1})} \quad (16)$$

with $p(x_{t+1}|x_t)$ as in (7), and $p(x_t)$ as well as $p(x_{t+1})$ as in (11). It follows that $p(x_t|x_{t+1})$ is also Gaussian of the form [19, pp. 337-339]

$$p(x_t|x_{t+1}) = \mathcal{N}(x_t; \mu_{t|t+1}, \Sigma_{t|t+1}) \quad (17)$$

with

$$\begin{aligned} \Sigma_{t|t+1} &= \Sigma_t - \Sigma_t A^T (Q + A \Sigma_t A^T)^{-1} A \Sigma_t \\ &= (\Sigma_t^{-1} + A^T Q^{-1} A)^{-1} \end{aligned} \quad (18a)$$

and

$$\mu_{t|t+1} = \Sigma_{t|t+1} (A^T Q x_{t+1} + \Sigma_t^{-1} \mu_t). \quad (18b)$$

The quantity $p(x_{t+1:T}|y_{t+1:T})$ in (15) is simply the joint backward filtering density at $t+1$.

We can now formulate the following backward filtering recursion. Assume that we want to target the non-normalized joint backward filtering density, that is, $p(x_{t:T}|y_{t:T})$ using importance sampling. Then, we can express the non-normalized importance weights $\tilde{v}_{t|T}^{(m)}$ as

$$\tilde{v}_{t|T}^{(m)} = \frac{p(y_t|x_t^{(m)})p(x_t^{(m)}|x_{t+1}^{(m)})p(x_{t+1:T}^{(m)}|y_{t+1:T})}{q(x_t^{(m)})} \quad (19)$$

where $q(x_t)$ is the proposal density. Since we aim at running the filter backwards in time, a filtering approximation of $p(x_{t+1:T}|y_{t+1:T})$ of the form

$$p(x_{t+1:T}|y_{t+1:T}) \approx \sum_{m=1}^M v_{t+1|T}^{(m)} \delta(x_{t+1:T} - x_{t+1:T}^{(m)}) \quad (20)$$

will be available at time t . Furthermore, choosing the inverse process dynamics as the proposal density

$$q(x_t) = p(x_t|x_{t+1}) \quad (21)$$

yields the non-normalized backward filter weights

$$\tilde{v}_{t|T}^{(m)} \propto v_{t+1|T}^{(m)} p(y_t|x_t^{(m)}) \quad (22)$$

which is essentially the same expression as obtained for a particle filter targeting $p(x_{1:t}|y_{1:t})$, but in the backward direction. We then obtain the approximation of the form (20) by normalizing the weights (22)

$$v_{t|T}^{(m)} = \frac{\tilde{v}_{t|T}^{(m)}}{\sum_{m=1}^M \tilde{v}_{t|T}^{(m)}}. \quad (23)$$

It is very important to point out that in this case, we have been able to exploit the model structure in order to find a proposal distribution inherent in the problem that we can sample from (similar to the bootstrap proposal in the forward filter). This is, however, not the theoretically optimal proposal.

The final problem is how to initialize the backward recursion. First, note that at $t = T$ we have the backward filtering density as

$$p(x_T|y_T) \propto p(y_T|x_T)p(x_T). \quad (24)$$

A straight-forward choice would then be to initialize the backward filter by using $p(x_T)$. Since $p(x_T)$ has quadratically increasing variance (see (12)) this is impractical, though, as this density would propose many particles in irrelevant areas of the state space. However, we know that the forward filter is generally capable of keeping the particles in the interesting area. Hence, one approach is to initialize the backward filter by re-using the forward particles $x_{T|T}^{(m)}$. These particles were generated by drawing samples from $p(x_T|x_{T-1}^{(m)})$ and thus, they have to be reweighed for the backward pass according to

$$\begin{aligned} v_{T|T}^{(m)} &\propto \frac{p(x_T^{(m)}|y_T)}{p(x_T^{(m)}|x_{T-1}^{(m)})} \\ &\propto \frac{p(y_T|x_T^{(m)})p(x_T^{(m)})}{p(x_T^{(m)}|x_{T-1}^{(m)})} \\ &= \frac{w_{T|T}^{(m)} p(x_T^{(m)})}{w_{t-1|t-1}^{(m)} p(x_T^{(m)}|x_{T-1}^{(m)})} \end{aligned} \quad (25)$$

where (8) was used to arrive at the last expression.

C. Smoother

We can now collect the results from the previous two sections in order to obtain the final formulation of the two-filter smoother. Recall that

$$\begin{aligned} p(x_t|y_{1:T}) &\propto \frac{p(x_t|y_{1:t-1})}{p(x_t)} p(x_t|y_{t:T}) \\ &\propto \frac{p(x_t|y_{1:t-1})}{p(x_t)} \int_{-\infty}^{\infty} p(x_{t:T}|y_{t:T}) dx_{t+1:T} \end{aligned} \quad (26)$$

Then, using (13) and (20) in (26) yields the approximation (up to proportionality)

$$\begin{aligned} p(x_t|y_{1:T}) &\approx \frac{\sum_{n=1}^M w_{t-1|t-1}^{(n)} \mathcal{N}(x_t; Ax_{t-1}^{(n)}, Q)}{\mathcal{N}(x_t; \mu_t, \Sigma_t)} \\ &\quad \times \int_{-\infty}^{\infty} \sum_{m=1}^M v_{t|T}^{(m)} \\ &\quad \quad \times \delta(x_{t:T} - x_{t:T}^{(m)}) dx_{t+1:T} \\ &= \frac{\sum_{n=1}^M w_{t-1|t-1}^{(n)} \mathcal{N}(x_t; Ax_{t-1}^{(n)}, Q)}{\mathcal{N}(x_t; \mu_t, \Sigma_t)} \\ &\quad \times \sum_{m=1}^M v_{t|T}^{(m)} \delta(x_t - x_{t|T}^{(m)}). \end{aligned} \quad (27)$$

From (27), the non-normalized smoothed weight can be found as

$$\tilde{v}_{t|T}^{(m)} \propto v_{t|T}^{(m)} \frac{\sum_{n=1}^M w_{t-1|t-1}^{(n)} \mathcal{N}(x_{t|T}^{(m)}; Ax_{t-1|t-1}^{(n)}, Q)}{\mathcal{N}(x_{t|T}^{(m)}; \mu_t, \Sigma_t)} \quad (28)$$

and equation (27) can be rewritten as

$$p(x_t|y_{1:T}) \approx \sum_{m=1}^M \nu_{t|T}^{(m)} \delta(x_t - x_{t|T}^{(m)}) \quad (29)$$

where

$$\nu_{t|T}^{(m)} = \frac{\tilde{\nu}_{t|T}^{(m)}}{\sum_{m=1}^M \tilde{\nu}_{t|T}^{(m)}}. \quad (30)$$

This finally yields the two-filter particle smoother summarized in Algorithm 1. Typical resampling techniques for mitigating sample impoverishment have been included in both the forward and backward filters in steps 2c) and 4d) [7]. Other improvements such as jittering, prior editing, resample-move (for example using MCMC kernels), or Rao-Blackwellization could be incorporated easily as well if necessary [7], [9].

Algorithm 1 Two-Filter Particle Smoother with Linear State Dynamics

- 1) Initialize the forward filter $x_{0|0}^{(n)} \sim p(x_0)$, $w_{0|0}^{(n)} = 1/M$
 - 2) For $t = 1, \dots, T$
 - a) Propagate the particles according to (6)-(7)
 - b) Calculate the forward particle weight using (8)
 - c) If $(\sum_{n=1}^M (w_{t|t}^{(n)})^2)^{-1} < M_T$
 - i) Resample with replacement such that
 - ii) Set $w_{t|t}^{(n)} = 1/M$
 - d) Update the prior statistics according to (12)
 - 3) Initialize the backward filter using (25), (28) and (30)
 - 4) For $t = T - 1, \dots, 1$
 - a) Backward propagate particles by using (17)-(18)
 - b) Calculate the backward particle weights according to (22)-(23)
 - c) Calculate the smoothed particle weights using (28) and (30)
 - d) If $(\sum_{n=1}^M (v_{t|T}^{(n)})^2)^{-1} < M_T$
 - i) Resample with replacement such that
 - ii) Set $w_{t|T}^{(m)} = 1/M$
-

III. NUMERICAL ILLUSTRATION

In order to illustrate the performance of the proposed smoother, two different simulation examples are provided here and the results are discussed.

A. Linear System

In the first example, the proposed smoother is applied to a linear Gaussian state space system. This in order to compare the performance of the smoother to the Rauch-Tung-Striebel smoother [4] which is known to be the minimum mean

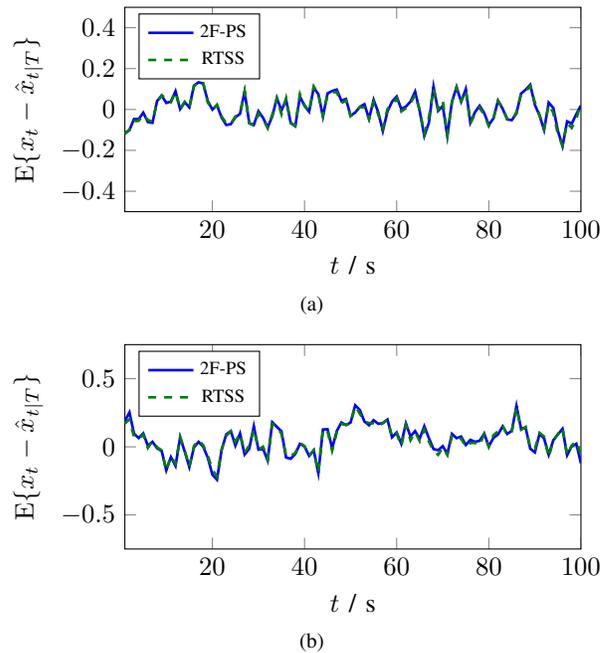


Fig. 1. Mean estimation error of the 100 MC simulations for the linear system (31) for the Rauch-Tung-Striebel smoother (dashed) and two filter particle smoother (solid).

squared error estimator for such systems. This comparison will hence allow us to see how the proposed particle approximation method performs compared to the analytical solution.

The system under consideration is given by

$$\begin{aligned} x_t &= \begin{bmatrix} 1 & 0.5 \\ 0 & 1 \end{bmatrix} x_{t-1} + v_t \\ y_t &= \begin{bmatrix} 1 & 0 \end{bmatrix} x_t + e_t \end{aligned}$$

with

$$\begin{aligned} \mu_0 &= \begin{bmatrix} 5 \\ 3 \end{bmatrix}, \\ \Sigma_0 &= \begin{bmatrix} 10 & 0 \\ 0 & 20 \end{bmatrix}, \\ Q &= I_2, \end{aligned}$$

and

$$R = 1.$$

In total, 100 Monte Carlo simulations were run. The number of particles in the particle smoother was chosen to be $M = 500$ and systematic resampling [20] with a threshold for the effective sample size $M_T = M/3$ was used.

Fig. 1 shows the mean estimation error for both states. As it can be seen, the error for the proposed method (2F-PS) matches the error for the Rauch-Tung-Striebel smoother (RTSS) almost exactly. Furthermore, the mean squared error estimated using the 100 Monte Carlo simulations is depicted in Fig. 2. Even the mean squared error is essentially equal for both smoothers and both states. This indicates that the proposed smoother indeed is a minimum mean squared error estimator.

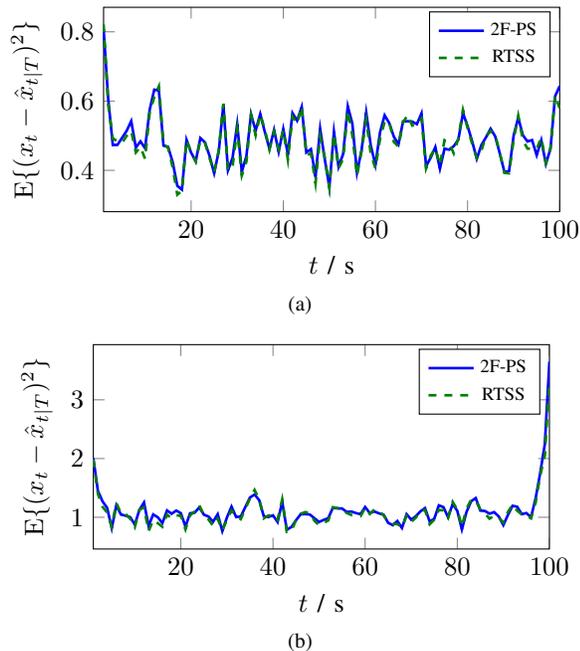


Fig. 2. Mean squared error of the 100 MC simulations for the linear system (31) for the Rauch-Tung-Striebel smoother (dashed) and two filter particle smoother (solid).

B. Target Tracking

In the second example, the smoother is used to estimate the trajectory of a target in a tracking scenario where a target's range and bearing are measured. This is a very common problem, for example in air traffic control, robotics, or autonomous vehicles. In this example, the proposed smoother is compared to a standard forward filtering, backward simulation (FFBSi) particle smoother [9] in order to assess the performance.

The dynamics are modeled using a constant velocity motion model of the form

$$x_t = \begin{bmatrix} 1 & 0 & T_s & 0 \\ 0 & 1 & 0 & T_s \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} x_{t-1} + v_t.$$

Here, the state vector x_t is given by

$$x_t = [p_t^x \quad p_t^y \quad v_t^x \quad v_t^y]^T$$

where p_t^x and p_t^y are the x- and y-positions of the target at time t , respectively, and v_t^x and v_t^y are the velocities in the respective direction. The measurements are the range and bearing of the target at time t which are given by

$$y_t = \begin{bmatrix} \sqrt{(p_t^x)^2 + (p_t^y)^2} \\ \text{atan2}(p_t^y, p_t^x) \end{bmatrix} + e_t.$$

($\text{atan2}(\cdot)$ is the four-quadrant inverse tangent function.) The parameters used in the simulations are as follows:

$$\mu_0 = \begin{bmatrix} -10 \\ 25 \\ 2 \\ -1 \end{bmatrix},$$

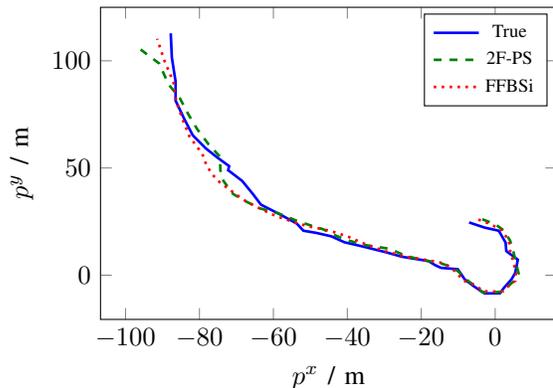


Fig. 3. Example trajectory (solid) together with the estimated trajectories for the two filter smoother (dashed) and FFBSi smoother (dotted).

$$\Sigma_0 = \begin{bmatrix} 10 & 0 & 0 & 0 \\ 0 & 5 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$Q = I_4,$$

$$R = \begin{bmatrix} 1 & 0 \\ 0 & 0.01 \end{bmatrix},$$

and $T_s = 1$ s.

$M = 500$ particles were used in the particle smoother. In the FFBSi smoother, $M_F = 1,000$ forward particles and $M_S = 500$ backward particles were used such that the smoothed posterior density is approximated by the same number of particles in both smoothers. In this example, a total of 20 Monte Carlo simulations were run.

Fig. 3 and Fig. 4 depict one random example trajectory and the corresponding measurement signals considered in the simulation. In Fig. 3, the true trajectory (solid) and the position estimates for the two filter smoother (dashed) and FFBSi (dotted) smoother are shown. It can be seen that both smoothers are able to accurately track the target, especially close to the sensor at the origin and when a constant motion is maintained.

Fig. 5 shows the estimated mean squared error for the 20 Monte Carlo simulations for both the position as well as the speed. It shows that the MSE for both filters are essentially equivalent and follow the same trend.

The biggest advantage of the proposed method, however, lies in the possibility of propagating the particles by using matrix operations only. This reduces computational requirements. A comparison of the average runtime over the 20 Monte Carlo simulations illustrates this: The two filter smoother took 5.9 s for one run on average, while the same metric is 164.1 s for the FFBSi smoother. (Note that these numbers are without any code optimizations.)

IV. CONCLUSIONS

By exploiting the model structure of systems with linear state dynamics and non-linear measurement functions, a fast two filter particle smoother has been developed. Its properties have been illustrated using numerical simulations.

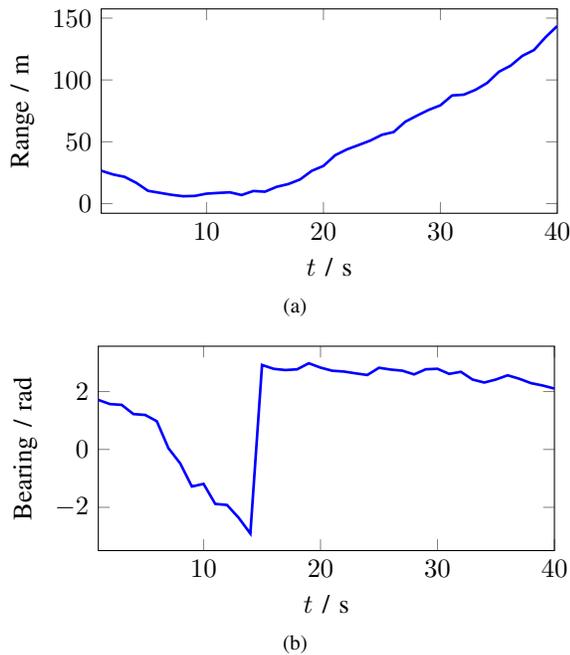


Fig. 4. Example measurement signals corresponding to the trajectory in Fig. 3. (a) Range, and (b) bearing.

The simulations showed that the proposed method performs comparably to the existing state of the art methods but at significantly lower computational cost.

Finally, note that since the method is based on standard importance sampling-based particle filters, the drawbacks of these follow directly to the proposed method. This includes, for example, the curse of dimensionality, meaning that high dimensional problems cannot be solved easily.

REFERENCES

- [1] R. Hostettler, "A two filter particle smoother for Wiener state-space systems," in *Control Applications, 2015 IEEE Conference on*, Sydney, Australia, September 2015.
- [2] S. Särkkä, *Bayesian Filtering and Smoothing*. Cambridge University Press, 2013.
- [3] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME, Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, March 1960.
- [4] H. E. Rauch, C. T. Striebel, and F. Tung, "Maximum likelihood estimates of linear dynamic systems," *AIAA Journal*, vol. 3, no. 8, pp. 1445–1450, August 1965.
- [5] E. A. Wan and R. Van Der Merwe, "The unscented Kalman filter for nonlinear estimation," in *Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000. AS-SPCC. The IEEE 2000*, 2000, pp. 153–158.
- [6] S. Särkkä, "Unscented Rauch-Tung-Striebel smoother," *Automatic Control, IEEE Transactions on*, vol. 53, no. 3, pp. 845–849, April 2008.
- [7] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *Radar and Signal Processing, IEE Proceedings F*, vol. 140, no. 2, pp. 107–113, April 1993.
- [8] P. M. Djurić, J. H. Kotecha, J. Zhang, Y. Huang, T. Ghirmai, M. F. Bugallo, and J. Miguez, "Particle filtering," *Signal Processing Magazine, IEEE*, vol. 20, no. 5, pp. 19–38, September 2003.
- [9] A. Doucet and A. M. Johansen, "A tutorial on particle filtering and smoothing: Fifteen years later," *Handbook of Nonlinear Filtering*, vol. 12, pp. 656–704, 2009.

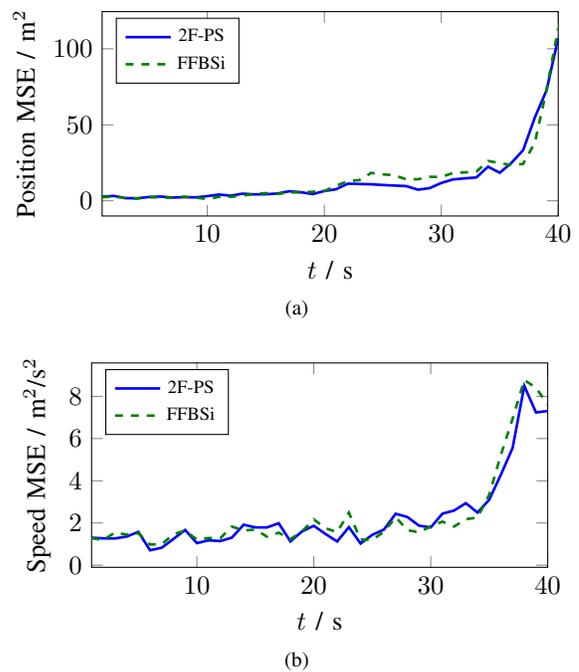


Fig. 5. Estimated mean squared error for the proposed two filter smoother (solid) and the FFBSi smoother (dashed) for (a) the position, and (b) the speed.

- [10] G. Kitagawa, "Monte Carlo filter and smoother for non-Gaussian non-linear state space models," *Journal of Computational and Graphical Statistics*, vol. 5, no. 1, pp. 1–25, 1996.
- [11] R. Douc, A. Garivier, E. Moulines, and J. Olsson, "Sequential Monte Carlo smoothing for general state space hidden Markov models," *The Annals of Applied Probability*, vol. 21, no. 6, pp. 2109–2145, December 2011.
- [12] E. Taghavi, F. Lindsten, L. Svensson, and T. B. Schön, "Adaptive stopping for fast particle smoothing," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, Vancouver, BC, Canada, May 2013, pp. 6293–6297.
- [13] J. Olsson and J. Westerborn, "Efficient particle-based online smoothing in general hidden Markov models: the PaRIS algorithm," *ArXiv e-prints*, December 2014, arXiv:1412.7550.
- [14] S. Särkkä, P. Bunch, and S. J. Godsill, "A backward-simulation based Rao-Blackwellized particle smoother for conditionally linear Gaussian models," in *Proceedings of the 16th IFAC Symposium on System Identification*, Brussels, Belgium, July 2012.
- [15] F. Lindsten, P. Bunch, S. J. Godsill, and T. B. Schön, "Rao-Blackwellized particle smoothers for mixed linear/nonlinear state-space models," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, Vancouver, BC, Canada, May 2013, pp. 6288–6292.
- [16] M. Briers, A. Doucet, and S. Maskell, "Smoothing algorithms for state-space models," *Annals of the Institute of Statistical Mathematics*, vol. 62, no. 1, pp. 61–89, 2010.
- [17] P. Fearnhead, D. Wyncoll, and J. Tawn, "A sequential smoothing algorithm with linear computational cost," *Biometrika*, vol. 97, no. 2, pp. 447–464, 2010.
- [18] G. Kitagawa, "The two-filter formula for smoothing and an implementation of the Gaussian-sum smoother," *Annals of the Institute of Statistical Mathematics*, vol. 46, no. 4, pp. 605–623, 1994.
- [19] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Upper Saddle River, NJ, USA: Prentice Hall, 1993.
- [20] T. Li, M. Bolić, and P. M. Djurić, "Resampling methods for particle filtering: Classification, implementation, and strategies," *Signal Processing Magazine, IEEE*, vol. 32, no. 3, pp. 70–86, May 2015.