Technical report from Automatic Control at Linköpings universitet

Inference in Mixed Linear/Nonlinear State-Space Models using Sequential Monte Carlo

Fredrik Lindsten, Thomas B. Schön

Division of Automatic Control

E-mail: lindsten@isy.liu.se, schon@isy.liu.se

31st March 2010

Report no.: LiTH-ISY-R-2946

Address:
Department of Electrical Engineering
Linköpings universitet
SE-581 83 Linköping, Sweden

WWW: http://www.control.isy.liu.se

AUTOMATIC CONTROL
REGLERTEKNIK
LINKÖPINGS UNIVERSITET

Abstract

In this work we apply sequential Monte Carlo methods, i.e., particle filters and smoothers, to estimate the state in a certain class of mixed linear/nonlinear state-space models. Such a model has an inherent conditionally linear Gaussian substructure. By utilizing this structure we are able to address even high-dimensional nonlinear systems using Monte Carlo methods, as long as only a few of the states enter nonlinearly. First, we consider the filtering problem and give a self-contained derivation of the well known Rao-Blackwellized particle filter. Thereafter we turn to the smoothing problem and derive a Rao-Blackwellized particle smoother capable of handling the fully interconnected model under study.

Keywords: SMC, Particle filter, Particle smoother, Rao-Blackwellization

Inference in Mixed Linear/Nonlinear State-Space Models using Sequential Monte Carlo

Fredrik Lindsten and Thomas B. Schön

2010-03-31

Abstract

In this work we apply sequential Monte Carlo methods, i.e., particle filters and smoothers, to estimate the state in a certain class of mixed linear/nonlinear state-space models. Such a model has an inherent conditionally linear Gaussian substructure. By utilizing this structure we are able to address even high-dimensional nonlinear systems using Monte Carlo methods, as long as only a few of the states enter nonlinearly. First, we consider the filtering problem and give a self-contained derivation of the well known Rao-Blackwellized particle filter. Thereafter we turn to the smoothing problem and derive a Rao-Blackwellized particle smoother capable of handling the fully interconnected model under study.

1 Introduction

A common problem in many different fields of science is that of estimating the state of a dynamical system, based on noisy observations from the system. If the system under study is linear and afflicted with Gaussian noise, the posterior distribution of the states, conditioned on the observations, is available in closed form, which allows for optimal inference. However, if the system is nonlinear and/or non-Gaussian, this is no longer the case. To be able to deal with such systems, we thus need to resort to approximations. One popular approach is to use sequential Monte Carlo (SMC) methods, which rely on random samples from the sought distributions, see e.g., [2, 4]. SMC is known to perform well for systems of fairly low dimension, but for high-dimensional systems the performance can be seriously degraded.

In this document we shall consider a special kind of nonlinear systems, containing conditionally linear Gaussian substructures (see Section 2). By utilizing this structure, it is possible to address even high-dimensional systems using Monte Carlo methods. This idea, known as marginalization or Rao-Blackwellization, is well known in the literature. This is especially true when it comes to Rao-Blackwellized particle filtering (RBPF), which is discussed in for instance [11, 3]. Rao-Blackwellized particle smoothing (RBPS) is somewhat more immature, but two different smoothers are presented in [5] and [1] respectively. In [9] the RBPS derived in this work is used in an expectation maximization algorithm to estimate unknown parameters in a mixed linear/nonlinear state-space model.

The purpose of this work is to give an explanatory derivation of the RBPF given in [11], and the RBPS, previously given in [5], in a unified, self-contained document. We shall also extend the smoother of [5] to be able to handle the fully interconnected model (1) under study. To the best of the authors' knowledge, this is the first time that a RBPS applicable to this kind of model is presented.

2 Problem Formulation

Consider the following mixed linear/nonlinear state-space model

$$a_{t+1} = f^a(a_t) + A^a(a_t)z_t + w_t^a, \tag{1a}$$

$$z_{t+1} = f^z(a_t) + A^z(a_t)z_t + w_t^z, \tag{1b}$$

$$y_t = h(a_t) + C(a_t)z_t + e_t. \tag{1c}$$

The model is nonlinear in a_t, which will be denoted the nonlinear states, and affine in z_t, which will be denoted the linear states[1]. The process noise is assumed to be white and Gaussian according to

$$w_t = \begin{bmatrix} w_t^a \\ w_t^z \end{bmatrix} \sim \mathcal{N}(0, Q(a_t)), \qquad Q(a_t) = \begin{bmatrix} Q^a(a_t) & Q^{az}(a_t) \\ (Q^{az}(a_t))^T & Q^z(a_t) \end{bmatrix} \tag{1d}$$

and the measurement noise is assumed to be white and Gaussian according to

$$e_t \sim \mathcal{N}\left(0, R(a_t)\right). \tag{1e}$$

The initial state z_1 is Gaussian according to

$$z_1 \sim \mathcal{N}\left(\bar{z}_{1|0}(a_1), P_{1|0}(a_1)\right). \tag{1f}$$

The matrices $Q(a_t)$, $R(a_t)$ and $P_{1|0}(a_1)$ are all assumed to be non-singular (for all values of their arguments). The density of a_1, $p(a_1)$, is assumed to be known.

Given a set of observations $y_{1:s} \triangleq \{y_1, \dots, y_s\}$ we wish to do inference in this model. More precisely we seek to compute conditional expectations of some functions of the states

$$\mathrm{E}\left[g(a_{1:t}, z_{1:t}) \mid y_{1:s}\right].$$

We shall confine ourselves to two special cases of this problem, filtering and smoothing, characterized as follows:

1. **Filtering:** At each time $t = 1, \dots, T$, compute expectations of functions of the state at time t, conditioned on the measurements up to time $s = t$, i.e.,

$$\mathrm{E}\left[g(a_t, z_t) \mid y_{1:t}\right]. \tag{2}$$

2. **Smoothing:** At each time $t = 1, \dots, T-1$, compute expectations of functions of the states at time t and $t+1$, conditioned on the measurements up to time $s = T > t$, i.e.,

$$\mathrm{E}\left[g(a_{t:t+1}, z_{t:t+1}) \mid y_{1:T}\right]. \tag{3}$$

[1]This type of model is often called conditionally linear Gaussian, even though conditionally affine Gaussian would be a more suiting name. However, the difference is of minor importance, and we shall use the former name in this report as well.

The reason for why we, in the smoothing case, consider functions of the states at time t and $t+1$ is that expectations of this kind often appear in methods that utilizes the smoothing estimates, e.g., parameter estimation using expectation maximization [12]. Clearly, functions of the state at just time t or $t+1$ are covered as special cases.

3 Importance Sampling and Resampling

In the interest of giving self-contained presentation, this section will give a short introduction to importance sampling (IS) and sampling importance resampling (SIR), which is the core of the well known particle filter (PF).

3.1 Importance sampling

Assume that we wish to evaluate the expected value of some function of a random variable $g(z)$, where $z \sim p(z)$, i.e., we seek

$$I_p(g(z)) \triangleq \mathrm{E}_p[g(z)] = \int g(z)p(z)\,dz. \tag{4}$$

Now, if this integral is intractable we can approximate it with the Monte Carlo (MC) expectation

$$\hat{I}_p^{\mathrm{MC}}(g(z)) = \frac{1}{N}\sum_{i=1}^{N} g(z^i), \tag{5}$$

where $\{z^i\}_{i=1}^N$ are independent samples from $p(z)$. This sum will under weak conditions converge to the true expectation as N tends to infinity.

It is convenient to introduce an approximation of the continuous distribution $p(z)$ based on the samples z^i, as

$$p(z) \approx \hat{p}^{\mathrm{MC}}(z) = \frac{1}{N}\sum_{i=1}^{N} \delta(z - z^i), \tag{6}$$

where $\delta(\cdot)$ is the Dirac δ-function. $\hat{p}(z)$ will be referred to as a point-mass approximation of $p(z)$ since it has "probability mass" only in a finite number of points. If (6) is plugged into (4), the approximation (5) is obtained.

The problem that one often faces is that it is hard to sample from the desired distribution $p(z)$ (which we will refer to as the target distribution). However, this can be handled using importance sampling. Introduce a proposal distribution $q(z)$, which we easily can draw samples from. The support of the proposal should cover the support of the target, but besides from this we can choose it arbitrarily. We then have

$$\begin{aligned} I_p(g(z)) &= \int g(z)p(z)\,dz = \int g(z)\frac{p(z)}{q(z)}q(z)\,dz = I_q\left(g(z)\frac{p(z)}{q(z)}\right) \\ &\approx \hat{I}_q^{\mathrm{MC}}\left(g(z)\frac{p(z)}{q(z)}\right) = \frac{1}{N}\sum_{i=1}^{N} g(z^i)\frac{p(z^i)}{q(z^i)}, \end{aligned} \tag{7}$$

where $\{z^i\}_{i=1}^N$ are independent samples from $q(z)$. We see that this leads to the same approximation as in (5), but the samples are weighted with the quantities

$$w^i \triangleq \frac{1}{N}\frac{p(z^i)}{q(z^i)}, \tag{8}$$

known as importance weights. This corrects for the bias introduced by sampling from the wrong distribution. For this method to work it is important that the proposal density resembles the target density as good as possible.

It is often the case that the target (and possible also the proposal) density only can be evaluated up to a scaling factor. Let $p(z) = \check{p}(z)/Z_p$ and $q(z) = \check{q}(z)/Z_q$ where $\check{p}(z)$ and $\check{q}(z)$ can be evaluated, but Z_p and Z_q are unknown constants. If this is plugged into (7) we obtain

$$I_p(g(z)) \approx \frac{Z_q}{Z_p}\frac{1}{N}\sum_{i=1}^{N} g(z^i)\frac{\check{p}(z^i)}{\check{q}(z^i)}. \tag{9}$$

To obtain an approximation of the unknown constant Z_q/Z_p we can use the same set of samples and note that

$$\begin{aligned}\frac{Z_p}{Z_q} &= \frac{1}{Z_q}\int \check{p}(z)\,dz = \frac{1}{Z_q}\int \check{p}(z)\frac{Z_q q(z)}{\check{q}(z)}\,dz \\ &= \int \frac{\check{p}(z)}{\check{q}(z)}q(z)\,dz \approx \hat{I}_q^{\text{MC}}\left(\frac{\check{p}(z)}{\check{q}(z)}\right) = \frac{1}{N}\sum_{i=1}^{N}\frac{\check{p}(z^i)}{\check{q}(z^i)} = \sum_{i=1}^{N}\check{w}^i,\end{aligned} \tag{10}$$

where we have introduced the unnormalized importance weights

$$\check{w}^i \triangleq \frac{1}{N}\frac{\check{p}(z^i)}{\check{q}(z^i)}. \tag{11}$$

An approximation of the (normalized) importance weights is then

$$w^i = \frac{Z_q}{Z_p}\frac{1}{N}\frac{\check{p}(z^i)}{\check{q}(z^i)} = \frac{Z_q}{Z_p}\check{w}^i \approx \frac{\check{w}^i}{\sum_{i=1}^{N}\check{w}^i}. \tag{12}$$

From now on we shall drop the ˇ from the target and the proposal distributions, but keep in mind that the normalization of the importance weights is due to the unknown scaling factors.

We can use the result of the importance sampling to approximate the target as a point-mass distribution similar to (6), yielding

$$p(z) \approx \hat{p}^{\text{IS}}(z) = \sum_{i=1}^{N} w^i \delta(z - z^i), \tag{13}$$

where z^i are sampled from the proposal. The importance sampling method is summarized in Algorithm 1.

3.2 Sampling importance resampling

As pointed out in the previous section, the IS sampling scheme will result in a weighted sample from the target distribution, $\{z^i, w^i\}_{i=1}^N$. If we for some reason

Algorithm 1 Importance sampling

1. Choose an appropriate proposal density $q(z)$.
2. Draw N independent samples from the proposal
$$z^i \sim q(z), \qquad i = 1, \ldots, N.$$
3. Compute the importance weights and normalize
$$w^i = \frac{\breve{w}^i}{\sum_{i=1}^N \breve{w}^i}, \qquad \breve{w}^i = \frac{p(z^i)}{q(z^i)}. \tag{14}$$
4. Approximate the target distribution as
$$\hat{p}^{\text{IS}}(z) = \sum_{i=1}^N w^i \delta(z - z^i), \tag{15a}$$
which can be used to compute expectations according to
$$\hat{I}_p^{\text{IS}}(g(z)) = \int g(z)\hat{p}^{\text{IS}}(z)\, dz = \sum_{i=1}^N w^i g(z^i). \tag{15b}$$

seek an unweighted sample from the target (this is for instance important in SMC), we can employ sampling importance resampling (SIR).

The idea is very simple. Since IS gives us an approximation of the target distribution (13), we can draw N new, independent samples from this distribution
$$\zeta^j \sim \hat{p}^{\text{IS}}(z), \qquad j = 1, \ldots, N. \tag{16}$$
Since (13) can be seen as a discrete distribution with support at N different points, each with probability w^i, $i = 1, \ldots, N$, sampling from this distribution is straightforward. We simply set $\zeta^j = z^i$ with probability w^i, i.e., $P(\zeta^j = z^i) = w^i$ for $j = 1, \ldots, N$. The sample $\{\zeta^j\}_{j=1}^N$ will be an approximate sample from the target $p(z)$. Since the approximation (13) improves as N tends to infinity, so will the quality of the sample $\{\zeta^j\}_{j=1}^N$.

4 Rao-Blackwellized Particle Filter

The Rao-Blackwellized particle filter (RBPF) is a Monte Carlo method used to compute expectations of the type (2). The filter uses SIR in a way that exploits the structure in model (1). The sought expectations can be expressed as
$$\begin{aligned}\mathrm{E}\left[g(a_t, z_t) \mid y_{1:t}\right] &= \iint g(a_t, z_t) p(a_t, z_t \mid y_{1:t})\, da_t dz_t \\ &= \iint g(a_t, z_t) p(z_t \mid a_{1:t}, y_{1:t}) p(a_{1:t} \mid y_{1:t})\, da_{1:t} dz_t. \end{aligned} \tag{17}$$

The trick is that the distribution $p(z_t \mid a_{1:t}, y_{1:t})$ (i.e., the filtering distribution for the linear states conditioned on the nonlinear state trajectory $a_{1:t}$ and the measurements $y_{1:t}$) can be computed analytically. We thus only need to use sampling techniques for the nonlinear states, which reduces the variance of the estimator. This is known as Rao-Blackwellization after the Rao-Blackwell theorem, see [8]. Let us rewrite (17) as

$$\begin{aligned} \mathrm{E}\left[g(a_t, z_t) \mid y_{1:t}\right] &= \int \left(\int g(a_t, z_t) p(z_t \mid a_{1:t}, y_{1:t}) \, dz_t \right) p(a_{1:t} \mid y_{1:t}) \, da_{1:t} \\ &= \int \mathrm{E}\left[g(a_t, z_t) \mid a_{1:t}, y_{1:t}\right] p(a_{1:t} \mid y_{1:t}) \, da_{1:t} \\ &\approx \sum_{i=1}^{N} w_t^i \mathrm{E}\left[g(a_t^i, z_t) \mid a_{1:t}^i, y_{1:t}\right] \end{aligned} \tag{18}$$

where we have made use of the IS approximation (15). Observe that the expectations in (18) are with respect to z_t, conditioned on the nonlinear state trajectory (and the measurements).

The task at hand can thus be formulated as follows; given $y_{1:t} = \{y_1, \ldots, y_t\}$, draw N samples from the distribution $p(a_{1:t} | y_{1:t})$ using importance sampling. For each of these samples $\{a_{1:t}^i\}_{i=1}^N$, find the sufficient statistics for the density $p(z_t \mid a_{1:t}^i, \; y_{1:t})$. Do this sequentially for $t = 1, \ldots, T$.

4.1 Updating the linear states

We shall start the derivation of the RBPF by showing how we can obtain the distribution $p(z_t \mid a_{1:t}, y_{1:t})$ sequentially. As already stated this distribution will be available in closed form. More specifically it will turn out to be Gaussian, and we thus only need to keep track of its first and second moments.

The derivation will be given as a proof by induction. By the end of this section we shall see that $p(z_1 \mid a_1, y_1)$ is Gaussian and can thus be written according to $p(z_1 \mid a_1, y_1) = \mathcal{N}(z_1; \bar{z}_{1|1}(a_1), P_{1|1}(a_1))$ where we have defined $\bar{z}_{1|1}(a_1)$ and $P_{1|1}(a_1)$ as the mean and covariance of the distribution, respectively. Hence, assume that, for $t \geq 2$,

$$p(z_{t-1} \mid a_{1:t-1}, y_{1:t-1}) = \mathcal{N}\left(z_{t-1}; \bar{z}_{t-1|t-1}(a_{1:t-1}), P_{t-1|t-1}(a_{1:t-1})\right), \tag{19}$$

where the mean and covariance are functions of the state trajectory $a_{1:t-1}$ (naturally, they do also depend on the measurements $y_{1:t-1}$, but we shall not make that dependence explicit). We shall now see that this implies

$$p(z_t \mid a_{1:t}, y_{1:t}) = \mathcal{N}\left(z_t; \bar{z}_{t|t}(a_{1:t}), P_{t|t}(a_{1:t})\right) \tag{20}$$

and show how we can obtain the sufficient statistics for this distribution.

Using the Markov property and the state transition density given by the model (1), we have

$$\begin{aligned} &p(z_t, a_t \mid z_{t-1}, a_{1:t-1}, y_{1:t-1}) = p(z_t, a_t \mid z_{t-1}, a_{t-1}) \\ &\quad = \mathcal{N}\left(\begin{bmatrix} a_t \\ z_t \end{bmatrix}; \begin{bmatrix} f^a(a_{t-1}) \\ f^z(a_{t-1}) \end{bmatrix} + \begin{bmatrix} A^a(a_{t-1}) \\ A^z(a_{t-1}) \end{bmatrix} z_{t-1}, \begin{bmatrix} Q^a(a_{t-1}) & Q^{az}(a_{t-1}) \\ (Q^{az}(a_{t-1}))^T & Q^z(a_{t-1}) \end{bmatrix} \right) \end{aligned} \tag{21}$$

which is affine in z_{t-1}. A basic result for Gaussian variables, given in Corollary A.1 in Appendix A, is that an affine transformation of a Gaussian variable will remain Gaussian. If we apply this result to (19) and (21) we get

$$p(z_t, a_t \mid a_{1:t-1}, y_{1:t-1}) \\ = \mathcal{N}\left(\begin{bmatrix} a_t \\ z_t \end{bmatrix}; \begin{bmatrix} \alpha_{t|t-1}(a_{1:t-1}) \\ \zeta_{t|t-1}(a_{1:t-1}) \end{bmatrix}, \begin{bmatrix} \Sigma^a_{t|t-1}(a_{1:t-1}) & \Sigma^{az}_{t|t-1}(a_{1:t-1}) \\ (\Sigma^{az}_{t|t-1}(a_{1:t-1}))^T & \Sigma^z_{t|t-1}(a_{1:t-1}) \end{bmatrix}\right), \tag{22a}$$

with (the dependencies on a_{t-1} and $a_{1:t-1}$ have been dropped to keep the notation simple)

$$\alpha_{t|t-1}(a_{1:t-1}) = f^a + A^a \bar{z}_{t-1|t-1}, \tag{22b}$$
$$\zeta_{t|t-1}(a_{1:t-1}) = f^z + A^z \bar{z}_{t-1|t-1}, \tag{22c}$$
$$\Sigma^a_{t|t-1}(a_{1:t-1}) = Q^a + A^a P_{t-1|t-1}(A^a)^T, \tag{22d}$$
$$\Sigma^{az}_{t|t-1}(a_{1:t-1}) = Q^{az} + A^a P_{t-1|t-1}(A^z)^T, \tag{22e}$$
$$\Sigma^z_{t|t-1}(a_{1:t-1}) = Q^z + A^z P_{t-1|t-1}(A^z)^T. \tag{22f}$$

This is simply a prediction of the state at time t, conditioned on $a_{1:t-1}$ and $y_{1:t-1}$. In (22b)–(22c) the system dynamics is simulated and (22d)–(22f) shows how the uncertainty in the prediction depends on the process noise and the prior uncertainty in the linear state.

Using Theorem A.2 we can marginalize (22) to obtain

$$p(a_t \mid a_{1:t-1}, y_{1:t-1}) = \mathcal{N}\left(a_t; \alpha_{t|t-1}(a_{1:t-1}), \Sigma^a_{t|t-1}(a_{1:t-1})\right) \tag{23}$$

and from Theorem A.1 we can condition (22) on a_t to get

$$p(z_t \mid a_{1:t}, y_{1:t-1}) = \mathcal{N}\left(z_t; \bar{z}_{t|t-1}(a_{1:t}), P_{t|t-1}(a_{1:t})\right), \tag{24a}$$

with

$$\bar{z}_{t|t-1}(a_{1:t}) = \zeta_{t|t-1} + (\Sigma^{az}_{t|t-1})^T (\Sigma^a_{t|t-1})^{-1}(a_t - \alpha_{t|t-1}), \tag{24b}$$
$$P_{t|t-1}(a_{1:t}) = \Sigma^z_{t|t-1} - (\Sigma^{az}_{t|t-1})^T (\Sigma^a_{t|t-1})^{-1}(\Sigma^{az}_{t|t-1}). \tag{24c}$$

The above expressions constitutes the time update of the filter. The prediction of the nonlinear state, which will be used in the sampling (see Section 4.2), is given by (23). Once the nonlinear state trajectory is augmented with a new sample we can condition the prediction of the linear state on this sample, according to (24). In doing so we provide some information about the linear state, through the connection between the linear and the nonlinear parts of the state vector. From (24) we see that the estimate is updated accordingly and that the covariance is reduced. This update is very similar to a Kalman filter measurement update, and is therefore sometimes denoted the "extra measurement update" of the RBPF. Observe however, that we have not used any information about the current measurement y_t up to this point. This is what we will do next.

From (1) we have the measurement density

$$p(y_t \mid a_{1:t}, z_t, y_{1:t-1}) = p(y_t \mid a_t, z_t) = \mathcal{N}\left(y_t; h(a_t) + C(a_t)z_t, R(a_t)\right), \tag{25}$$

which is affine in z_t. We can thus use Corollary A.1 and the result (24) to obtain the measurement likelihood

$$p(y_t \mid a_{1:t}, y_{1:t-1}) = \mathcal{N}\left(y_t; \hat{y}_t(a_{1:t}), S_t(a_{1:t})\right), \tag{26a}$$

with

$$\hat{y}_t(a_{1:t}) = h + C\bar{z}_{t|t-1}, \tag{26b}$$
$$S_t(a_{1:t}) = R + CP_{t|t-1}C^T, \tag{26c}$$

and also the posterior of z_t conditioned on the new measurement

$$p(z_t \mid a_{1:t}, y_{1:t}) = \mathcal{N}\left(z_t; \bar{z}_{t|t}(a_{1:t}), P_{t|t}(a_{1:t})\right), \tag{27a}$$

with

$$\bar{z}_{t|t}(a_{1:t}) = \bar{z}_{t|t-1} + K_t(y_t - \hat{y}_t), \tag{27b}$$
$$P_{t|t}(a_{1:t}) = P_{t|t-1} - K_t C P_{t|t-1}, \tag{27c}$$
$$K_t(a_{1:t}) = P_{t|t-1}C^T S_t^{-1}. \tag{27d}$$

Now, if we define $y_{1:0} \triangleq \emptyset$, so that $p(z_1 \mid a_{1:1}, y_{1:0}) = p(z_1 \mid a_1)$ and analogously for other distributions, we see that the expression (24a) coincides with the prior (1f) at $t = 1$. The computations in (24) – (27) will thus hold at $t = 1$, which in turn implies that $p(z_1 \mid a_1, y_1) = \mathcal{N}(z_1; \bar{z}_{1|1}(a_1), P_{1|1}(a_1))$ and the assumption (19) is valid for $t \geq 2$.

4.2 Sampling the nonlinear states

As we saw in the previous section, the filtering distribution for the linear states z_t could be computed analytically when conditioned on the nonlinear state trajectory $a_{1:t}$. However, the filtering distribution for $a_{1:t}$ is not available in closed form and we must thus resort to approximations. In this work we use Monte Carlo approximation, in which a distribution is represented by a number of samples from it. In this section we shall see how we sequentially can sample from the filtering distribution for the nonlinear states $p(a_{1:t} \mid y_{1:t})$ using importance sampling.

Let us assume that $t \geq 2$. Sampling at time $t = 1$ can be done from straightforward modifications of the results given here. Hence, the target distribution can be expressed as

$$\begin{aligned} p(a_{1:t} \mid y_{1:t}) &\propto p(y_t \mid a_{1:t}, y_{1:t-1}) p(a_{1:t} \mid y_{1:t-1}) \\ &= p(y_t \mid a_{1:t}, y_{1:t-1}) p(a_t \mid a_{1:t-1}, y_{1:t-1}) p(a_{1:t-1} \mid y_{1:t-1}). \end{aligned} \tag{28}$$

To sample from this target distribution, we choose a proposal distribution, which factorizes according to

$$q(a_{1:t} \mid y_{1:t}) = q(a_t \mid a_{1:t-1}, y_{1:t}) \underbrace{q(a_{1:t-1} \mid y_{1:t-1})}_{\text{previous proposal}}. \tag{29}$$

Observe that this is not a generally applicable factorization of a probability density function, due to the missing conditioning on y_t in the second factor.

Sampling from (29) can be done by first sampling from $q(a_{1:t-1} \mid y_{1:t-1})$ (which is already done at time $t-1$) and then append samples from $q(a_t \mid a^i_{1:t-1}, y_{1:t})$,

$$\begin{aligned} a^i_t &\sim q(a_t \mid a^i_{1:t-1}, y_{1:t}), \\ a^i_{1:t} &= \{a^i_{1:t-1}, a^i_t\}. \end{aligned} \tag{30}$$

The importance weights are given by, using (23) and (26),

$$\begin{aligned} w^i_t &= \frac{p(a^i_{1:t} \mid y_{1:t})}{q(a^i_{1:t} \mid y_{1:t})} \propto \frac{p(y_t \mid a^i_{1:t}, y_{1:t-1}) p(a^i_t \mid a^i_{1:t-1}, y_{1:t-1})}{q(a^i_t \mid a^i_{1:t-1}, y_{1:t})} \underbrace{\frac{p(a^i_{1:t-1} \mid y_{1:t-1})}{q(a^i_{1:t-1} \mid y_{1:t-1})}}_{=w^i_{t-1}} \\ &= w^i_{t-1} \frac{\mathcal{N}\left(y_t; \hat{y}_t(a^i_{1:t}), S_t(a^i_{1:t})\right) \mathcal{N}\left(a^i_t; \alpha_{t|t-1}(a^i_{1:t-1}), \Sigma^a_{t|t-1}(a^i_{1:t-1})\right)}{q(a^i_t \mid a^i_{1:t-1}, y_{1:t})}. \end{aligned} \tag{31}$$

Since we only know the importance weights up to proportionality they should, according to (12), be normalized so that

$$\sum_{i=1}^{N} w^i_t = 1. \tag{32}$$

4.3 Resampling

Just as in "standard" particle filtering we need to resample the trajectories to avoid degeneracy, see for instance [3]. The basic idea is to discard particles with low weights and duplicate particles with high weights. This is done in a resampling step, similar to what is discussed in Section 3.2. Many different resampling procedures have been proposed, see e.g., [2]. Any method of choice can be used in the RBPF.

4.4 RBPF algorithm

We summarize the Rao-Blackwellized particle filter in Algorithm 2. To simplify the notation, for functions in argument a_t or $a_{1:t}$, e.g., $R(a_t)$ and $\bar{z}_{t|t}(a_{1:t})$, let us write $R^i_t \triangleq R(a^i_t)$ and $\bar{z}^i_{t|t} \triangleq \bar{z}_{t|t}(a^i_{1:t})$ etc.

In the interest of giving a somewhat more compact presentation, the algorithm is only given for time $t \geq 2$ and does not show how to do the initialization at $t = 1$. However, this initialization is very similar to the steps given in the algorithm. In step 1, we choose a proposal $q(a_1 \mid y_1)$, since we do not have any "old" trajectory to condition on. Step 2 is not needed since we have an initial "prediction" of the linear states, $\bar{z}_{1|0}(a_1)$, $\hat{P}_{1|0}(a_1)$ from the prior distribution (1f). In step 3, the weights are given by

$$\breve{w}^i_1 = \frac{\mathcal{N}\left(y_1; \hat{y}^i_1, S^i_1\right) p(a^i_1)}{q(a^i_1 \mid y_1)}, \qquad w^i_t = \frac{\breve{w}^i_t}{\sum_{i=1}^{N} \breve{w}^i_t},$$

with $\hat{y}^i_1$ and S^i_1 as in Algorithm 2. Finally, step 4 and step 5 are identical to Algorithm 2.

Algorithm 2 RBPF (for $t \geq 2$)

1. **Sampling:** Choose a proposal $q(a_t \mid a_{1:t-1}, y_{1:t})$, draw new samples and append to the nonlinear state trajectories. For $i = 1, \ldots, N$,

$$a_t^i \sim q(a_t \mid a_{1:t-1}^i, y_{1:t}),$$
$$a_{1:t}^i = \{a_{1:t-1}^i, a_t^i\}.$$

2. **Prediction:** Predict the state and condition the linear state on the newly drawn a_t^i. For $i = 1, \ldots, N$,

$$\begin{aligned}
\alpha_{t|t-1}^i &= f_{t-1}^{a,i} + A_{t-1}^{a,i} \bar{z}_{t-1|t-1}^i, \\
\bar{z}_{t|t-1}^i &= f_{t-1}^{z,i} + A_{t-1}^{z,i} \bar{z}_{t-1|t-1}^i + (\Sigma_{t|t-1}^{az,i})^T (\Sigma_{t|t-1}^{a,i})^{-1} (a_t^i - \alpha_{t|t-1}^i), \\
P_{t|t-1}^i &= \Sigma_{t|t-1}^{z,i} - (\Sigma_{t|t-1}^{az,i})^T (\Sigma_{t|t-1}^{a,i})^{-1} (\Sigma_{t|t-1}^{az,i}),
\end{aligned}$$

with

$$\begin{aligned}
\Sigma_{t|t-1}^{a,i} &= Q_{t-1}^{a,i} + A_{t-1}^{a,i} P_{t-1|t-1}^i (A_{t-1}^{a,i})^T, \\
\Sigma_{t|t-1}^{az,i} &= Q_{t-1}^{az,i} + A_{t-1}^{a,i} P_{t-1|t-1}^i (A_{t-1}^{z,i})^T, \\
\Sigma_{t|t-1}^{z,i} &= Q_{t-1}^{z,i} + A_{t-1}^{z,i} P_{t-1|t-1}^i (A_{t-1}^{z,i})^T.
\end{aligned}$$

3. **Weighting:** Evaluate and normalize the importance weights

$$\begin{aligned}
\breve{w}_t^i &= \frac{\mathcal{N}\left(y_t; \hat{y}_t^i, S_t^i\right) \mathcal{N}\left(a_t^i; \alpha_{t|t-1}^i, \Sigma_{t|t-1}^{a,i}\right)}{q(a_t^i \mid a_{1:t-1}^i, y_{1:t})} w_{t-1}^i, \qquad i = 1, \ldots, N, \\
w_t^i &= \frac{\breve{w}_t^i}{\sum_{i=1}^N \breve{w}_t^i},
\end{aligned}$$

with

$$\hat{y}_t^i = h_t^i + C_t^i \bar{z}_{t|t-1}^i,$$
$$S_t^i = R_t^i + C_t^i P_{t|t-1}^i (C_t^i)^T.$$

4. **Update the linear states:** Compute the sufficient statistics for the linear states, given the current measurement. For $i = 1, \ldots, N$,

$$\begin{aligned}
\bar{z}_{t|t}^i &= \bar{z}_{t|t-1}^i + K_t^i (y_t - \hat{y}_t^i), \\
P_{t|t}^i &= P_{t|t-1}^i - K_t^i C_t^i P_{t|t-1}^i, \\
K_t^i &= P_{t|t-1}^i (C_t^i)^T (S_t^i)^{-1}.
\end{aligned}$$

5. **Resampling:** Use a resampling scheme of choice and update the importance weights $\{w_t^i\}_{i=1}^N$ accordingly.

5 Rao-Blackwellized Forward Filter Backward Simulator

In this section we shall derive a Rao-Blackwellized particle smoother (RBPS). This smoother was first derived in [5], but for a slightly different model structure, in which the nonlinear state dynamics are independent of the linear states. In this section we will make the derivation for the fully interconnected model (1). In Section 5.4, the relationship between the smoother given here and the one presented in [5] is discussed.

The smoother is a so called forward filter backward simulator (FFBSi) type of smoother since it is based on a forward pass of the standard RBPF presented above, and a backward simulation where new "smoothed" samples are drawn from the grid spanned by the forward filtering (FF) pass.

5.1 Derivation

The Rao-Blackwellized FFBSi (RB-FFBSi) is a Monte Carlo method used to compute expectations of the type (3), i.e.,

$$\begin{aligned}
&\mathrm{E}\left[g(a_{t:t+1}, z_{t:t+1}) \mid y_{1:T}\right] \\
&\quad = \iint g(a_{t:t+1}, z_{t:t+1}) p(a_{t:t+1}, z_{t:t+1} \mid y_{1:T})\, da_{t:t+1} dz_{t:t+1} \\
&\quad = \iint g(a_{t:t+1}, z_{t:t+1}) p(z_{t:t+1} \mid a_{t:T}, y_{1:T}) p(a_{t:T} \mid y_{1:T})\, da_{t:T} dz_{t:t+1}. \quad (33)
\end{aligned}$$

As previously mentioned, the reason for why we consider functions of the states at time t and $t+1$ is that expectations of this kind often appear in methods that utilizes the smoothing estimates, e.g., parameter estimation using expectation maximization [12]. Expectations of functions of the state at either time t *or* $t+1$ are clearly special cases and are thus also covered. However, it can be instructive to consider such functions explicitly anyway. We shall thus focus the derivation of the smoother on finding (approximate) expressions for the densities

$$\begin{aligned}
&p(a_{t:T} \mid y_{1:T}) && \text{for } t = 1, \ldots, T, && (34a) \\
&p(z_t \mid a_{t:T}, y_{1:T}) && \text{for } t = 1, \ldots, T, && (34b) \\
&p(z_{t:t+1} \mid a_{t:T}, y_{1:T}) && \text{for } t = 1, \ldots, T-1. && (34c)
\end{aligned}$$

We shall assume that we have performed the forward filtering already. We have thus, for $t = 1, \ldots, T$, obtained N nonlinear state trajectories with corresponding importance weights, $\{a^i_{1:t}, w^i_t\}_{i=1}^N$, sampled from the distribution $p(a_{1:t} \mid y_{1:t})$. We have also, for each of these trajectories, evaluated the sufficient statistics for the linear states,

$$\{\bar{z}_{t|t}(a^i_{1:t}), P_{t|t}(a^i_{1:t})\}_{i=1}^N. \quad (35)$$

As indicated by (35), the sufficient statistics are functions of the nonlinear state *trajectory*. This implies that if we take a different path forward through the nonlinear part of the state-space, this will influence our belief about the linear states. However, in the FFBSi we will sample trajectories backward in time which typically will be different from the forward trajectories (see Figure 1 and

Figure 2 for an illustration). During the backward simulation we can hence not allow ourselves to condition on the entire forward nonlinear state trajectory. To circumvent this we will make the following approximation.

Approximation 1 *At each time $t = 1, \ldots, T$, the filtering distribution for the linear state z_t does not depend on the entire nonlinear state trajectory, but merely on the endpoint of this trajectory, i.e.,*

$$p(z_t \mid a_{1:t}, y_{1:t}) = p(z_t \mid a_t, y_{1:t}) = \mathcal{N}\left(z_t; \bar{z}_{t|t}(a_t), P_{t|t}(a_t)\right) \tag{36a}$$

with

$$\bar{z}_{t|t}(a_t) = \bar{z}_{t|t}(a_{1:t}), \tag{36b}$$

$$P_{t|t}(a_t) = P_{t|t}(a_{1:t}), \tag{36c}$$

where $\bar{z}_{t|t}(a_{1:t})$ and $P_{t|t}(a_{1:t})$ are given by the RBPF recursions.

The above approximation can be motivated by considering the point-mass approximation of the filtering distribution from the RBPF,

$$p(z_t, a_{1:t} \mid y_{1:t}) \approx \sum_{i=1}^{N} w_t^i \mathcal{N}\left(z_t; \bar{z}_{t|t}(a_{1:t}^i), P_{t|t}(a_{1:t}^i)\right) \delta(a_{1:t} - a_{1:t}^i). \tag{37}$$

If we marginalize this distribution over $a_{1:t-1}$ we obtain

$$p(z_t, a_t \mid y_{1:t}) \approx \sum_{i=1}^{N} w_t^i \mathcal{N}\left(z_t; \bar{z}_{t|t}(a_{1:t}^i), P_{t|t}(a_{1:t}^i)\right) \delta(a_t - a_t^i), \tag{38}$$

which implies

$$p(z_t \mid a_t = a_t^i, y_{1:t}) \approx \mathcal{N}\left(z_t; \bar{z}_{t|t}(a_{1:t}^i), P_{t|t}(a_{1:t}^i)\right). \tag{39}$$

From (39) we get precisely Approximation 1. The expression (39) is indeed an approximation, as opposed to (20) which is exact. The reason for this is that in the marginalization (38), the point-mass (particle) approximation representing the distribution in the $a_{1:t-1}$-dimensions of the state-space is "injected" into the linear states as well. It should be mentioned that Approximation 1 is required also in the original RB-FFBSi derived in [5].

The task at hand is now to draw the backward trajectories, i.e., samples from the smoothed distribution $\tilde{a}_{t:T}^j \sim p(a_{t:T} \mid y_{1:T})$, $j = 1, \ldots, N$ and thereafter evaluate $p(z_t \mid \tilde{a}_{t:T}^j, y_{1:T})$ and $p(z_{t:t+1} \mid \tilde{a}_{t:T}^j, y_{1:T})$ (the latter only for $t < T$) for each sample. We see that the task is already fulfilled at time $t = T$, since the FF then supplies the sought samples and distributions (under Approximation 1). These samples are however, due to the importance sampling nature of the RBPF, associated with corresponding weights. The RB-FFBSi does not use importance sampling, but is instead designed to sample on the grid spanned by the FF. This can be seen as a kind of resampling of the FF where the "future" measurements are taken into account. To initialize this procedure at time $t = T$ we shall thus conduct an initial resampling of the FF.

The derivation is now presented as a proof by induction. We shall assume that we have the samples $\{\tilde{a}_{t+1:T}^j\}_{j=1}^N$ and that the distributions for the linear states are given by

$$p(z_{t+1} \mid a_{t+1:T}, y_{1:T}) = \mathcal{N}\left(z_{t+1}; \bar{z}_{t+1|T}(a_{t+1:T}), P_{t+1|T}(a_{t+1:T})\right) \tag{40}$$

and show how to complete the recursions at time t.

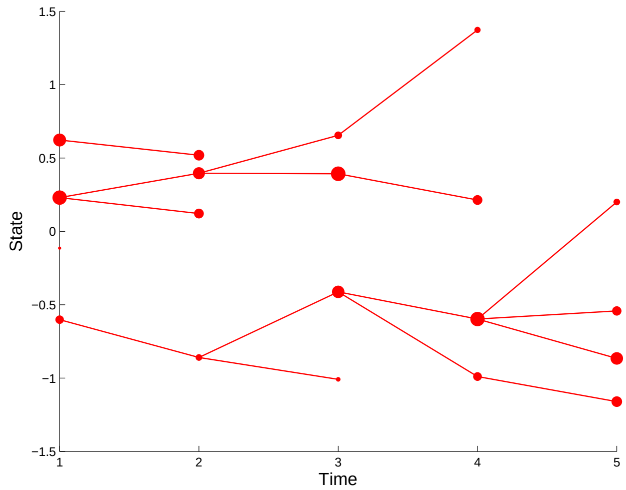

Figure 1: Particle trajectories for $N = 4$ particles over $T = 5$ time steps after a completed FF pass. The sizes of the dots represent the particle weights.

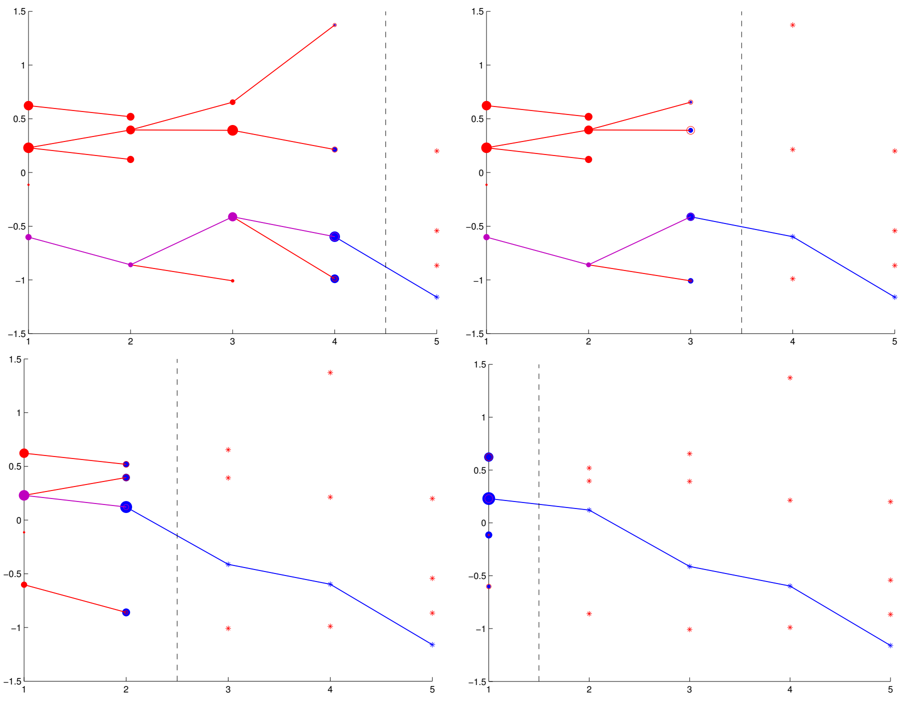

Figure 2: The simulation of a single backward trajectory. Upper left; one of the FF particles is drawn randomly at $t = 5$, shown as a blue asterisk (*). The particle weights at $t = 4$ are thereafter recomputed and another particle is drawn and added to the backward trajectory. Upper right and lower left; the trajectory is appended with new particles at $t = 3$ and $t = 2$, respectively. Lower right; a final particle is appended at $t = 1$, forming a complete backward trajectory. Observe that the trajectory differs from the ancestral line of the particle as it was in the FF.

5.2 Sampling

Our target distribution (for the nonlinear states) can be factorized as

$$p(a_{t:T} \mid y_{1:T}) = p(a_t \mid a_{t+1:T}, y_{1:T}) \underbrace{p(a_{t+1:T} \mid y_{1:T})}_{\text{previous target}}. \tag{41}$$

We can thus, as will be shown, sample $\tilde{a}_t^j \sim p(a_t \mid \tilde{a}_{t+1:T}^j, y_{1:T})$ and append the samples to the previous ones, $\tilde{a}_{t:T}^j = \{\tilde{a}_t^j, \tilde{a}_{t+1:T}^j\}$.

It turns out that it is in fact easier to sample from the joint distribution (see Appendix B)

$$p(z_{t+1}, a_{1:t} \mid a_{t+1:T}, y_{1:T}) = p(a_{1:t} \mid z_{t+1}, a_{t+1:T}, y_{1:T}) \underbrace{p(z_{t+1} \mid a_{t+1:T}, y_{1:T})}_{\text{known Gaussian from time } t+1} \tag{42}$$

We can easily sample $\tilde{z}_{t+1}^j \sim p(z_{t+1} \mid \tilde{a}_{t+1:T}^j, y_{1:T})$ and thereafter $\tilde{a}_{1:t}^j \sim p(a_{1:t} \mid \tilde{z}_{t+1}^j, \tilde{a}_{t+1:T}^j, y_{1:T})$ (which we will show next) to obtain a sample, $\{\tilde{z}_{t+1}^j, \tilde{a}_{1:t}^j\}$, from the joint distribution. We can then simply discard everything but $\tilde{a}_t^j$.

The first factor in (42) is given by (see Appendix C)

$$p(a_{1:t} \mid z_{t+1}, a_{t+1:T}, y_{t:T}) = p(a_{1:t} \mid z_{t+1}, a_{t+1}, y_{1:t}). \tag{43}$$

This result is rather natural; given the states at time $t+1$, there is no extra information available in the states at time $\tau > t+1$ or in the measurements at time $\tau > t$. We can write

$$\begin{aligned} p(a_{1:t} \mid z_{t+1}, a_{t+1}, y_{1:t}) &= \frac{p(z_{t+1}, a_{t+1} \mid a_{1:t}, y_{1:t}) p(a_{1:t} \mid y_{1:t})}{p(z_{t+1}, a_{t+1} \mid y_{1:t})} \\ &\propto \big/ \text{in argument } a_{1:t} \big/ \propto p(z_{t+1}, a_{t+1} \mid a_{1:t}, y_{1:t}) p(a_{1:t} \mid y_{1:t}), \end{aligned} \tag{44}$$

where, from (22), the first factor is given by

$$\begin{aligned} &p(z_{t+1}, a_{t+1} \mid a_{1:t}, y_{1:t}) \\ &= \mathcal{N}\left(\begin{bmatrix} a_{t+1} \\ z_{t+1} \end{bmatrix}; \begin{bmatrix} \alpha_{t+1|t}(a_{1:t}) \\ \zeta_{t+1|t}(a_{1:t}) \end{bmatrix}, \begin{bmatrix} \Sigma_{t+1|t}^a(a_{1:t}) & \Sigma_{t+1|t}^{az}(a_{1:t}) \\ (\Sigma_{t+1|t}^{az}(a_{1:t}))^T & \Sigma_{t+1|t}^z(a_{1:t}) \end{bmatrix} \right). \end{aligned} \tag{45}$$

For the second factor in (44), our best approximation is a point-mass distribution (from the FF),

$$p(a_{1:t} \mid y_{1:t}) \approx \sum_{i=1}^N w_t^i \delta(a_{1:t} - a_{1:t}^i). \tag{46}$$

The way in which we can sample from (44) is thus to draw among the particles given by the FF, with probabilities updated according to the samples $\tilde{z}_{t+1}^j$ and $\tilde{a}_{t+1}^j$. Summarizing the above we obtain

$$\begin{aligned} p(a_{1:t} \mid \tilde{z}_{t+1}^j, \tilde{a}_{t+1:T}^j, y_{1:T}) &\approx \frac{\sum_{i=1}^N w_t^i p(\tilde{z}_{t+1}^j, \tilde{a}_{t+1}^j \mid a_{1:t}^i, y_{1:t}) \delta(a_{1:t} - a_{1:t}^i)}{\sum_{k=1}^N w_t^k p(\tilde{z}_{t+1}^j, \tilde{a}_{t+1}^j \mid a_{1:t}^k, y_{1:t})} \\ &= \sum_{i=1}^N w_{t|T}^{i,j} \delta(a_{1:t} - a_{1:t}^i), \end{aligned} \tag{47}$$

with

$$w_{t|T}^{i,j} \triangleq \frac{w_t^i p(\tilde{z}_{t+1}^j, \tilde{a}_{t+1}^j \mid a_{1:t}^i, y_{1:t})}{\sum_{k=1}^N w_t^k p(\tilde{z}_{t+1}^j, \tilde{a}_{t+1}^j \mid a_{1:t}^k, y_{1:t})}. \tag{48}$$

The backward simulation is illustrated in Figure 1 and Figure 2.

5.3 Smoothing the linear states

Once we have sampled the nonlinear backward trajectories $\{\tilde{a}_{t:T}^j\}_{j=1}^N$, the next step is to find the sufficient statistics for the linear states, that will turn out to be approximately Gaussian,

$$p(z_t \mid a_{t:T}, y_{1:T}) \approx \mathcal{N}\left(z_t; \bar{z}_{t|T}(a_{t:T}), P_{t|T}(a_{t:T})\right). \tag{49}$$

This will be done in the following way:

1. Use the FF solution to find the distribution $p(z_t \mid z_{t+1}, a_{t:T}, y_{1:T})$ which is Gaussian and affine in z_{t+1}.
2. Approximate the distribution $p(z_{t+1} \mid a_{t:T}, y_{1:T})$ as the conditional smoothing distribution for the linear states at time $t+1$, $p(z_{t+1} \mid a_{t:T}, y_{1:T}) \approx p(z_{t+1} \mid a_{t+1:T}, y_{1:T})$.
3. Combine $p(z_t \mid z_{t+1}, a_{t:T}, y_{1:T})$ and $p(z_{t+1} \mid a_{t:T}, y_{1:T})$ to get the conditional joint (in z_t and z_{t+1}) smoothing distribution for the linear states at time t, $p(z_{t:t+1} \mid a_{t:T},\ y_{1:T})$ and also the marginal "version" of this $p(z_t \mid a_{t:T},\ y_{1:T})$.

We will now address these three steps in order.

5.3.1 Step 1 - Using the filter information

We shall now find an expression for the distribution

$$p(z_t \mid z_{t+1}, a_{t:T}, y_{1:T}) = p(z_t \mid z_{t+1}, a_t, a_{t+1}, y_{1:t}) \tag{50}$$

(see Appendix C for the derivation of this equality). We have the transition density

$$\begin{aligned} p(z_{t+1}, a_{t+1} \mid z_t, a_t, y_{1:t}) &= p(z_{t+1}, a_{t+1} \mid z_t, a_t) \\ &= \mathcal{N}\left(\begin{bmatrix} a_{t+1} \\ z_{t+1} \end{bmatrix}; \underbrace{\begin{bmatrix} f^a(a_t) \\ f^z(a_t) \end{bmatrix}}_{\triangleq f(a_t)} + \underbrace{\begin{bmatrix} A^a(a_t) \\ A^z(a_t) \end{bmatrix}}_{\triangleq A(a_t)} z_t, \underbrace{\begin{bmatrix} Q^a(a_t) & Q^{az}(a_t) \\ (Q^{az}(a_t))^T & Q^z(a_t) \end{bmatrix}}_{=Q(a_t)} \right) \end{aligned} \tag{51}$$

and, using Approximation 1, the filtering distribution

$$p(z_t \mid a_t, y_{1:t}) = \mathcal{N}\left(z_t; \bar{z}_{t|t}(a_t), P_{t|t}(a_t)\right). \tag{52}$$

This is an affine transformation of Gaussian variables, and from Corollary A.1 we thus get

$$p(z_t \mid z_{t+1}, a_t, a_{t+1}, y_{1:t}) = \mathcal{N}\left(z_t; \bar{z}_{t|t}^+(a_{t:t+1}), P_{t|t}^+(a_{t:t+1})\right), \tag{53}$$

with

$$\bar{z}^+_{t|t}(a_{t:t+1}) = P^+_{t|t}(a_{t:t+1})\Big(A(a_t)^T Q(a_t)^{-1} \left(\begin{bmatrix} a_{t+1}^T & z_{t+1}^T \end{bmatrix}^T - f(a_t)\right) \\ + P_{t|t}(a_t)^{-1}\bar{z}_{t|t}(a_t)\Big). \quad (54)$$

To expand the above expression we introduce

$$Q(a_t)^{-1} = \begin{bmatrix} \Lambda^a(a_t) & \Lambda^{az}(a_t) \\ (\Lambda^{az}(a_t))^T & \Lambda^z(a_t) \end{bmatrix} \quad (55a)$$

$$\begin{aligned}\begin{bmatrix} W^a(a_t) & W^z(a_t) \end{bmatrix} &= A(a_t)^T Q(a_t)^{-1} \\ &= \Big[(A^a(a_t))^T \Lambda^a(a_t) + (A^z(a_t))^T (\Lambda^{az}(a_t))^T \quad \dots \\ &\quad (A^a(a_t))^T \Lambda^{az}(a_t) + (A^z(a_t))^T \Lambda^z(a_t)\Big] \end{aligned} \quad (55b)$$

yielding (dropping the arguments a_t and a_{t+1} in the first two rows to keep the notation uncluttered)

$$\begin{aligned}\bar{z}^+_{t|t}(a_{t:t+1}) &= P^+_{t|t}\left(W^a(a_{t+1} - f^a) + W^z z_{t+1} - W^z f^z + P^{-1}_{t|t}\bar{z}_{t|t}\right) \\ &= P^+_{t|t} W^z z_{t+1} + \underbrace{P^+_{t|t}\left(W^a(a_{t+1} - f^a) - W^z f^z + P^{-1}_{t|t}\bar{z}_{t|t}\right)}_{\triangleq c^+_{t|t}(a_{t:t+1})} \\ &= P^+_{t|t}(a_{t:t+1}) W^z(a_t) z_{t+1} + c^+_{t|t}(a_{t:t+1}). \end{aligned} \quad (56)$$

Furthermore, the covariance matrix is given by

$$\begin{aligned}P^+_{t|t}(a_{t:t+1}) &= \left(P_{t|t}(a_t)^{-1} + A(a_t)^T Q(a_t)^{-1} A(a_t)\right)^{-1} \\ &= \left(P_{t|t}(a_t)^{-1} + W^a(a_t)A^a(a_t) + W^z(a_t)A^z(a_t)\right)^{-1} \end{aligned} \quad (57a)$$

or alternatively

$$P^+_{t|t}(a_{t:t+1}) = P_{t|t}(a_t) - P_{t|t}(a_t)A(a_t)^T \left(Q(a_t) + A(a_t)P_{t|t}(a_t)A(a_t)^T\right)^{-1} \\ \times A(a_t)P_{t|t}(a_t). \quad (57b)$$

5.3.2 Step 2 - Approximating the smoothed distribution

From the above discussion we have that $p(z_t \mid z_{t+1}, a_{t:T}, y_{1:T})$ is Gaussian and affine in z_{t+1}. Thus, if also $p(z_{t+1} \mid a_{t:T}, y_{1:T})$ would be Gaussian, we could apply Theorem A.3 to obtain the sought smoothing distribution. This will be done in Section 5.3.3.

However, the distribution $p(z_{t+1} \mid a_{t:T}, y_{1:T})$ is typically not Gaussian. To circumvent this we shall use the following approximation.

Approximation 2 *For all backward trajectories, $\{\tilde{a}^j_{t:T}\}_{j=1}^N$ we shall assume that*

$$p(z_{t+1} \mid \tilde{a}^j_{t:T}, y_{1:T}) \approx p(z_{t+1} \mid \tilde{a}^j_{t+1:T}, y_{1:T}) = \mathcal{N}\left(z_{t+1}; \bar{z}^j_{t+1|T}, P^j_{t+1|T}\right). \quad (58)$$

The approximation implies that we assume that the smoothing estimate for z_{t+1} is independent of which FF particle a_t^i that is appended to the backward trajectory. This approximation can be motivated by the fact that a particle a_t^i is more probable to be drawn if it has a good fit to the current smoothing trajectory. Hence, it should not affect the smoothing estimate at time $t+1$ to any significant extent, a claim that has been confirmed empirically through simulations (see Section 6). It should be mentioned that this approximation is required also in the original RB-FFBSi derived in [5].

5.3.3 Step 3 - Combining the information

We now have

$$p(z_t \mid z_{t+1}, a_{t:T}, y_{1:T}) = \mathcal{N}\left(z_t; \bar{z}^+_{t|t}(a_{t:t+1}), P^+_{t|t}(a_{t:t+1})\right), \tag{59a}$$

$$\bar{z}^+_{t|t}(a_{t:t+1}) = P^+_{t|t}(a_{t:t+1}) W^z(a_t) z_{t+1} + c^+_{t|t}(a_{t:t+1}), \tag{59b}$$

which is affine in z_{t+1} and

$$p(z_{t+1} \mid a_{t:T}, y_{1:T}) = \mathcal{N}\left(z_{t+1}; \bar{z}_{t+1|T}(a_{t+1:T}), P_{t+1|T}(a_{t+1:T})\right). \tag{60}$$

We can thus use Theorem A.3 to obtain

$$p(z_{t:t+1} \mid a_{t:T}, y_{1:T}) = \mathcal{N}\left(\begin{pmatrix} z_t \\ z_{t+1} \end{pmatrix}; \begin{pmatrix} \bar{z}_{t|T} \\ \bar{z}_{t+1|T} \end{pmatrix}, \begin{pmatrix} P_{t|T} & M_{t|T} \\ M^T_{t|T} & P_{t+1|T} \end{pmatrix}\right), \tag{61a}$$

with

$$\bar{z}_{t|T}(a_{t:T}) = P^+_{t|t}(a_{t:t+1}) W^z(a_t) \bar{z}_{t+1|T}(a_{t+1:T}) + c^+_{t|t}(a_{t:t+1}), \tag{61b}$$

$$P_{t|T}(a_{t:T}) = P^+_{t|t}(a_{t:t+1}) + M_{t|T}(a_{t:T})(W^z(a_t))^T P^+_{t|t}(a_{t:t+1}), \tag{61c}$$

$$M_{t|T}(a_{t:T}) = P^+_{t|t}(a_{t:t+1}) W^z(a_t) P_{t+1|T}(a_{t+1:T}). \tag{61d}$$

Finally, using Theorem A.2 we obtain the marginal distribution

$$p(z_t \mid a_{t:T}, y_{1:T}) = \mathcal{N}\left(z_t; \bar{z}_{t|T}(a_{t:T}), P_{t|T}(a_{t:T})\right). \tag{62}$$

We summarize the RB-FFBSi procedure in Algorithm 3.

5.4 A special case

In this work we have considered the mixed linear/nonlinear model (1). Another, very much related, model often found in the literature (e.g., [5, 1]) is

$$a_{t+1} \sim p(a_{t+1} \mid a_t), \tag{63a}$$

$$z_{t+1} = f^z(a_t) + A^z(a_t) z_t + w^z_t, \tag{63b}$$

$$y_t = h(a_t) + C(a_t) z_t + e_t, \tag{63c}$$

with

$$w^z_t \sim \mathcal{N}(0, Q^z(a_t)), \tag{63d}$$

$$e_t \sim \mathcal{N}(0, R(a_t)). \tag{63e}$$

Algorithm 3 RB-FFBSi

1. **Initialize:**

   (a) Run a forward pass of the RBPF and store the following quantities for $i = 1, \ldots, N$:

   - The particles, $a^i_{1:t}$ $(t = 1, \ldots, T)$
   - $\bar{z}^i_{t|t}$ and $P^i_{t|t}$ $(t = 1, \ldots, T)$
   - $\alpha^i_{t+1|t}$, $\zeta^i_{t+1|t}$, $\Sigma^{a,i}_{t+1|t}$, $\Sigma^{az,i}_{t+1|t}$ and $\Sigma^{z,i}_{t+1|t}$ $(t = 1, \ldots, T-1)$

   (b) Resample the FF at time $t = T$, $P(\tilde{a}^j_T = a^i_T) = w^i_T$, $j = 1, \ldots, N$.

   (c) Set $t := T - 1$.

2. **Sampling:** For each backward trajectory, $\{\tilde{a}^j_{t+1:T}\}_{j=1}^N$:

   (a) Draw

$$\tilde{z}^j_{t+1} \sim p(z_{t+1} \mid \tilde{a}^j_{t+1:T}, y_{1:T}) = \mathcal{N}(z_{t+1}; \bar{z}^j_{t+1|T}, P^j_{t+1|T}).$$

   (b) For each particle in the FF, $i = 1, \ldots, N$, evaluate (48) using (22)

$$w^{i,j}_{t|T} = \frac{w^i_t p(\tilde{z}^j_{t+1}, \tilde{a}^j_{t+1} \mid a^i_{1:t}, y_{1:t})}{\sum_{k=1}^N w^k_t p(z^j_{t+1}, \tilde{a}^j_{t+1} \mid a^k_{1:t}, y_{1:t})}.$$

   (c) Set $\tilde{a}^j_{1:t} = a^i_{1:t}$ with probability $w^{i,j}_{t|T}$, i.e., $P(\tilde{a}^j_{1:t} = a^i_{1:t}) = w^{i,j}_{t|T}$.

   (d) Discard $\tilde{a}^j_{1:t-1}$ and set $\tilde{a}^j_{t:T} = \{\tilde{a}^j_t, \tilde{a}^j_{t+1:T}\}$.

3. **Linear states:** For each backward trajectory, $\{\tilde{a}^j_{t:T}\}_{j=1}^N$:

   Update the sufficient statistics according to

$$\begin{aligned}
\bar{z}^j_{t|T} &= P^{+j}_{t|t} W^{z,j} \bar{z}^j_{t+1|T} + c^{+j}_{t|t}, \\
P^j_{t|T} &= P^{+j}_{t|t} + M^j_{t|T} (W^{z,j})^T P^{+j}_{t|t}, \\
M^j_{t|T} &= P^{+j}_{t|t} W^{z,j} P^j_{t+1|T},
\end{aligned}$$

   where

$$\begin{aligned}
c^{+j}_{t|t} &= P^{+j}_{t|t} \left(W^{a,j} (a^j_{t+1} - f^{a,j}) - W^{z,j} f^{z,j} + (P^j_{t|t})^{-1} \bar{z}^j_{t|t} \right), \\
P^{+j}_{t|t} &= \left((P^j_{t|t})^{-1} + W^{a,j} A^{a,j} + W^{z,j} A^{z,j} \right)^{-1},
\end{aligned}$$

   and

$$\begin{aligned}
W^{a,j} &= (A^{a,j})^T \Lambda^{a,j} + (A^{z,j})^T (\Lambda^{az,j})^T, \\
W^{z,j} &= (A^{a,j})^T \Lambda^{az,j} + (A^{z,j})^T \Lambda^{z,j}.
\end{aligned}$$

4. **Termination condition:** If $t > 1$, set $t := t - 1$ and go to step 2, otherwise terminate.

Hence, the transition density for the nonlinear states (a_t) is arbitrary, but it does not depend on the linear states (z_t). In [5] a RB-FFBSi is derived for this model. If the transition density $p(a_{t+1} \mid a_t)$ is Gaussian we can write

$$p(a_{t+1} \mid a_t) = \mathcal{N}\left(a_{t+1}; f^a(a_t), Q^a(a_t)\right), \tag{64}$$

and model (63) is then a special case of model (1) corresponding to $A^a \equiv Q^{az} \equiv 0$. It can be instructive to see how Algorithm 3 will turn out for this special case.

Since the process noise covariance $Q(a_t)$ now is block diagonal we get the information matrices $\Lambda^a(a_t) = Q^a(a_t)^{-1}$, $\Lambda^z(a_t) = Q^z(a_t)^{-1}$ and $\Lambda^{az}(a_t) \equiv 0$. Furthermore, since $A^a(a_t) \equiv 0$, we get from (55)

$$W^a(a_t) \equiv 0, \tag{65a}$$
$$W^z(a_t) = A^z(a_t)^T Q^z(a_t)^{-1}, \tag{65b}$$

which in (56) and (57) gives

$$c^+_{t|t}(a_{t:t+1}) = -P^+_{t|t} W^z f^z + P^+_{t|t} P^{-1}_{t|t} \bar{z}_{t|t}, \tag{66a}$$

and

$$\begin{aligned} P^+_{t|t}(a_{t:t+1}) &= \left(P^{-1}_{t|t} + (A^z)^T (Q^z)^{-1} A^z\right)^{-1} \\ &= P_{t|t} - P_{t|t}(A^z)^T \left(Q^z + A^z P_{t|t}(A^z)^T\right)^{-1} A^z P_{t|t} \\ &= P_{t|t} - T_t A^z P_{t|t}, \end{aligned} \tag{66b}$$

where we have defined

$$T_t(a_{t:t+1}) \triangleq P_{t|t}(A^z)^T \left(Q^z + A^z P_{t|t}(A^z)^T\right)^{-1} = P_{t|t}(A^z)^T P^{-1}_{t+1|t}. \tag{66c}$$

The last equality follows from (22) and (24).

Now, consider the product

$$\begin{aligned} P^+_{t|t} W^z &= P_{t|t}(A^z)^T (Q^z)^{-1} - T_t A^z P_{t|t}(A^z)^T (Q^z)^{-1} \\ &= P_{t|t}(A^z)^T \left(I - P^{-1}_{t+1|t} A^z P_{t|t}(A^z)^T\right) (Q^z)^{-1} \\ &= P_{t|t}(A^z)^T P^{-1}_{t+1|t} \underbrace{\left(P_{t+1|t} - A^z P_{t|t}(A^z)^T\right)}_{=Q^z} (Q^z)^{-1} \\ &= P_{t|t}(A^z)^T P^{-1}_{t+1|t} = T_t. \end{aligned} \tag{67}$$

The expressions in (61) can now be rewritten

$$\begin{aligned} \bar{z}_{t|T}(a_{t:T}) &= T_t \bar{z}_{t+1|T} - T_t f^z + P^+_{t|t} P^{-1}_{t|t} \bar{z}_{t|t} \\ &= \bar{z}_{t|t} + T_t \left(\bar{z}_{t+1|T} - f^z - A^z \bar{z}_{t|t}\right) \\ &= \bar{z}_{t|t} + T_t \left(\bar{z}_{t+1|T} - \bar{z}_{t+1|t}\right), \end{aligned} \tag{68a}$$

where the last equality follows from (22) and (24),

$$\begin{aligned} P_{t|T}(a_{t:T}) &= P_{t|t} - T_t A^z P_{t|t} + T_t P_{t+1|T} T_t^T \\ &= \Big/ A^z P_{t|t} = P_{t+1|t} T_t^T \Big/ \\ &= P_{t|t} - T_t \left(P_{t+1|t} - P_{t+1|T}\right) T_t^T, \end{aligned} \tag{68b}$$

and finally

$$M_{t|T}(a_{t:T}) = T_t P_{t+1|T}. \tag{68c}$$

The above expressions for $\bar{z}_{t|T}$ and $P_{t|T}$ can be recognized as the Rauch-Tung-Striebel (RTS) recursions for the smoothed estimate in linear Gaussian state space models [10].

Furthermore, from (22)-(24) it is straightforward to show that

$$p(z_t, a_t \mid a_{1:t-1}, y_{1:t-1}) = \mathcal{N}(a_t; f^a, Q^a)\, \mathcal{N}\left(z_t; \bar{z}_{t+1|t}, P_{t+1|t}\right). \tag{69}$$

As expected, the RB-FFBSi for the special case presented in this section coincides with the one derived in [5].

6 Numerical Illustrations

In this section we will evaluate the presented filter and smoother on simulated data. Two different examples will be presented, first a linear Gaussian system and thereafter a mixed linear/nonlinear system. The purpose of including a linear Gaussian example is to gain confidence in the presented methods. This is possible, since, for this case, there are closed form solutions available for the filtering and smoothing densities. Optimal filtering can be performed using the Kalman filter (KF) and optimal smoothing using the Rauch-Tung-Striebel (RTS) recursions [10].

For both the linear and the mixed linear/nonlinear examples, we can clearly also address the inference problems using standard particle methods. For the filtering problem we shall employ the bootstrap particle filter (PF) [7], which will be compared with the RBPF presented in Algorithm 2. We shall use the "bootstrap version" of the RBPF as well, meaning that the state transition density will be used as proposal and that resampling is carried out at each time step. For the smoothing problem we will employ the (non-Rao-Blackwellized) FFBSi [6] as well as the RB-FFBSi presented in Algorithm 3.

6.1 A Linear Example

To test the presented filter and smoother, we shall start by considering a linear, second order system according to

$$\begin{pmatrix} a_{t+1} \\ z_{t+1} \end{pmatrix} = \begin{pmatrix} 1 & 0.1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} a_t \\ z_t \end{pmatrix} + w_t, \qquad w_t \sim \mathcal{N}(0, Q), \tag{70a}$$

$$y_t = a_t + e_t, \qquad e_t \sim \mathcal{N}(0, R), \tag{70b}$$

with $Q = 0.1 I_{2\times 2}$ and $R = 0.1$. The initial state of the system is Gaussian according to

$$\begin{pmatrix} a_1 \\ z_1 \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right). \tag{71}$$

When using RBPF and RB-FFBSi the first state a_t is treated as if it is nonlinear, whereas the second state z_t is treated as linear.

The comparison was made by pursuing a Monte Carlo study over 1000 realizations of data $y_{1:T}$ from the system (70), each consisting of $T = 200$ samples (measurements). The three filters, KF, PF and RBPF, and thereafter the three smoothers, RTS, FFBSi, RB-FFBSi, were run in parallel. The particle methods all employed $N = 50$ particles.

Table 1 and Table 2 gives the root mean squared errors (RMSE) for the three filters and smoothers respectively.

Table 1: RMSE for filters

Filter	a_t	z_t
PF	8.69	43.5
RBPF	8.35	33.4
KF	8.08	33.4

Table 2: RMSE for smoothers

Smoother	a_t	z_t
FFBSi	7.45	36.7
RB-FFBSi	7.09	22.8
RTS	6.72	22.7

The results are as expected. First, smoothing clearly decreases the RMSEs when compared to filtering. Second, Rao-Blackwellization has the desired effect of decreasing the RMSE when compared to standard particle methods. When looking at the "linear" state z_t the RBPF and the RB-FFBSi performs very close to the optimal KF and RTS, respectively. The PF and FFBSi shows much worse performance.

The key difference between PF/FFBSi and RBPF/RB-FFBSi is that in the former, the particles have to cover the distribution in two dimensions. In the RBPF/RB-FFBSi we marginalize one of the dimensions analytically and thus only need to deal with one of the dimensions using particles. For PF/FFBSi we could of course obtain better approximation of the distribution by increasing the number of particles. However, we will then run into the infamous curse of dimensionality, requiring an exponential increase in the number of particles and hence also in computational complexity, as the order of the system increases.

6.2 A Mixed Linear/Nonlinear Example

We shall now study a fourth order mixed linear/nonlinear system, where three of the states are conditionally linear Gaussian,

$$a_{t+1} = \arctan a_t + \begin{pmatrix} 1 & 0 & 0 \end{pmatrix} z_t + w_{a,t}, \tag{72a}$$

$$z_{t+1} = \begin{pmatrix} 1 & 0.3 & 0 \\ 0 & 0.92 & -0.3 \\ 0 & 0.3 & 0.92 \end{pmatrix} z_t + w_{z,t}, \tag{72b}$$

$$y_t = \begin{pmatrix} 0.1 a_t^2 \operatorname{sgn}(a_t) \\ 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 & 0 \\ 1 & -1 & 1 \end{pmatrix} z_t + e_t, \tag{72c}$$

with $w_t = \begin{bmatrix} w_{a,t} & w_{z,t}^T \end{bmatrix}^T \sim \mathcal{N}(0, Q)$, $Q = 0.01 I_{4\times 4}$ and $e_t \sim N(0, R)$, $R = 0.1 I_{2\times 2}$. The initial state of the system is Gaussian according to

$$\begin{pmatrix} a_1 \\ z_1 \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} 0 \\ 0_{3\times 1} \end{pmatrix}, \begin{pmatrix} 1 & 0_{1\times 3} \\ 0_{3\times 1} & 0_{3\times 3} \end{pmatrix} \right). \tag{73}$$

The z-system is oscillatory and marginally stable, with poles in 1, $0.92 \pm 0.3i$ and the z-variables are connected to the nonlinear a-system through $z_{1,t}$.

Again, 1000 realizations of data $y_{1:T}$ were generated, each consisting of $T = 200$ samples. Due to the nonlinear nature of this example we cannot employ the KF and the RTS. Hence, in the comparison, presented in Table 3 and Table 4, we have only considered the PF/FFBSi and the RBPF/RB-FFBSi, all using $N = 50$ particles.

Table 3: RMSE for filters

Filter	a_t	$z_{1,t}$	$z_{2,t}$	$z_{3,t}$
PF	27.3	16.2	8.58	6.83
RBPF	14.1	9.19	6.75	5.55

Table 4: RMSE for smoothers

Smoother	a_t	$z_{1,t}$	$z_{2,t}$	$z_{3,t}$
FFBSi	25.2	13.3	6.58	6.45
RB-FFBSi	10.2	4.86	3.81	4.24

The benefits of using Rao-Blackwellization becomes even more evident in this, more challenging, problem. Since we can marginalize three out of the four dimensions analytically, Rao-Blackwellization allows us to handle this fairly high-dimensional system using only 50 particles.

7 Conclusions

The purpose of this work has been to present a self-contained derivation of a Rao-Blackwellized particle filter and smoother. An existing Rao-Blackwellized particle smoother has been extended to be able to handle the fully interconnected model (1) under study. The benefit of using Rao-Blackwellization, whenever possible, is illustrated in two numerical examples. It is shown that Rao-Blackwellization tends to reduce the root mean squared errors of the state estimates, especially when the sate dimension is large. It can be concluded that one of the main strengths of the presented filter and smoother is that it enables the use of particle methods for high-dimensional mixed linear/nonlinear systems, as long as only a few of the states enter nonlinearly. This is a well known result, presented previously in for instance [11], where a Rao-Blackwellized particle filter is used on a nine dimensional system in a real world example.

A Manipulating Gaussian Random Variables

In this appendix we shall give a few results on how the multivariate Gaussian density can be manipulated. The following theorems and corollary gives us all the tools needed to derive the expressions for the so called linear states z_t in this work. The statements are given without proofs, since the proofs are easily found in standard textbooks on the subject.

A.1 Partitioned Gaussian

We shall start by giving two results on partitioned Gaussian variables. Assume (without loss of generality) that we have partitioned a Gaussian variable, its mean and its covariance as

$$x = \begin{pmatrix} a \\ b \end{pmatrix}, \qquad \mu = \begin{pmatrix} \mu_a \\ \mu_b \end{pmatrix}, \qquad \Sigma = \begin{pmatrix} \Sigma_a & \Sigma_{ab} \\ \Sigma_{ba} & \Sigma_b \end{pmatrix}, \tag{74}$$

where for reasons of symmetry $\Sigma_{ba} = \Sigma_{ab}^T$. It is also useful to write down the partitioned information matrix

$$\Lambda = \Sigma^{-1} = \begin{pmatrix} \Lambda_a & \Lambda_{ab} \\ \Lambda_{ba} & \Lambda_b \end{pmatrix}, \tag{75}$$

since this form will provide simpler calculations below. Note that, since the inverse of a symmetric matrix is also symmetric, we have $\Lambda_{ba} = \Lambda_{ab}^T$.

Theorem A.1 ***(Conditioning)*** *Let the stochastic variable $x = \begin{pmatrix} a^T & b^T \end{pmatrix}^T$ be Gaussian distributed with mean and covariance according to* (74)*, then the conditional density $p(a \mid b)$ is given by*

$$p(a \mid b) = \mathcal{N}\left(a; \mu_{a|b}, \Sigma_{a|b}\right),$$

where

$$\begin{aligned} \mu_{a|b} &= \mu_a + \Sigma_{ab}\Sigma_b^{-1}(b - \mu_b), \\ \Sigma_{a|b} &= \Sigma_a - \Sigma_{ab}\Sigma_b^{-1}\Sigma_{ba}, \end{aligned}$$

which using the information matrix can be written,

$$\begin{aligned} \mu_{a|b} &= \mu_a - \Lambda_a^{-1}\Lambda_{ab}(b - \mu_b), \\ \Sigma_{a|b} &= \Lambda_a^{-1}. \end{aligned}$$

Theorem A.2 ***(Marginalization)*** *Let the stochastic variable $x = \begin{pmatrix} a^T & b^T \end{pmatrix}^T$ be Gaussian distributed with mean and covariance according to* (74)*, then the marginal density $p(a)$ is given by*

$$p(a) = \mathcal{N}\left(a; \mu_a, \Sigma_a\right).$$

A.2 Affine transformations

In the previous section we started with the joint distribution for a and b. We then gave expressions for the marginal and the conditional distributions. We shall now take a different starting-point, namely that we are given the marginal density $p(a)$ and the conditional density $p(b \mid a)$ (affine in a) and derive expressions for the joint distribution of a and b, the marginal $p(b)$ and the conditional density $p(a \mid b)$.

Theorem A.3 ***(Affine transformation)*** *Assume that a, as well as b conditioned on a, are Gaussian distributed according to*

$$p(a) = \mathcal{N}\left(a; \mu_a, \Sigma_a\right),$$
$$p(b \mid a) = \mathcal{N}\left(b; Ma + \tilde{b}, \Sigma_{b|a}\right),$$

where M is a matrix (of appropriate dimension) and $\tilde{b}$ is a constant vector. The joint distribution of a and b is then given by

$$p(a, b) = \mathcal{N}\left(\begin{pmatrix} a \\ b \end{pmatrix}; \begin{pmatrix} \mu_a \\ M\mu_a + \tilde{b} \end{pmatrix}, R\right),$$

with

$$R = \begin{pmatrix} M^T\Sigma_{b|a}^{-1}M + \Sigma_a^{-1} & -M^T\Sigma_{b|a}^{-1} \\ -\Sigma_{b|a}^{-1}M & \Sigma_{b|a}^{-1} \end{pmatrix}^{-1} = \begin{pmatrix} \Sigma_a & \Sigma_a M^T \\ M\Sigma_a & \Sigma_{b|a} + M\Sigma_a M^T \end{pmatrix}.$$

Combining the results in Theorems A.1, A.2 and A.3 we get the following corollary.

Corollary A.1 ***(Affine transformation – marginal and conditional)*** *Assume that a, as well as b conditioned on a, are Gaussian distributed according to*

$$p(a) = \mathcal{N}\left(a; \mu_a, \Sigma_a\right),$$
$$p(b \mid a) = \mathcal{N}\left(b; Ma + \tilde{b}, \Sigma_{b|a}\right),$$

where M is a matrix (of appropriate dimension) and $\tilde{b}$ is a constant vector. The marginal distribution of b is then given by

$$p(b) = \mathcal{N}\left(b; \mu_b, \Sigma_b\right),$$

with

$$\mu_b = M\mu_a + \tilde{b},$$
$$\Sigma_b = \Sigma_{b|a} + M\Sigma_a M^T.$$

The conditional distribution of a given b is

$$p(a \mid b) = \mathcal{N}\left(a; \mu_{a|b}, \Sigma_{a|b}\right),$$

with

$$\mu_{a|b} = \Sigma_{a|b}\left(M^T\Sigma_{b|a}^{-1}(b - \tilde{b}) + \Sigma_a^{-1}\mu_a\right) = \mu_a + \Sigma_a M^T \Sigma_b^{-1}(b - \tilde{b} - M\mu_a),$$
$$\Sigma_{a|b} = \left(\Sigma_a^{-1} + M^T\Sigma_{b|a}^{-1}M\right)^{-1} = \Sigma_a - \Sigma_a M^T \Sigma_b^{-1} M\Sigma_a.$$

B Sampling in the RB-FFBSi

The sampling step in the RB-FFBSi at time t appends a new sample $\tilde{a}_t^j$ to a backward trajectory $\tilde{a}_{t+1:T}^j$. Hence, from (41) we see that we wish to draw samples from the distribution $p(a_t \mid a_{t+1:T}, y_{1:T})$. In this appendix we shall see why it is easier to instead sample from the join distribution $\{\tilde{a}_{1:t}^j, z_{t+1}^j\} \sim p(a_{1:t}, z_{t+1} \mid a_{t+1:T}, y_{1:T})$ and thereafter discard everything but $\tilde{a}_t^j$.

First of all we note that the backward simulation makes use of the FF particles, i.e., we only sample among the particles generated by the FF. This means that our target distribution can be written as a weighted point-mass distribution according to

$$p(a_t \mid a_{t+1:T}, y_{1:T}) \approx \sum_{i=1}^{N} \theta^i \delta(a_t - a_t^i), \tag{76}$$

with some, yet unspecified, weights θ^i. Clearly, the tricky part is to compute these weights, once we have them the sampling is trivial.

To see why it is indeed hard to compute the weights, we consider the joint distribution $p(a_{1:t}, z_{t+1} \mid a_{t+1:T}, y_{1:T})$. Following the steps in (42)–(47), this density is approximately

$$\begin{aligned} p(a_{1:t},&z_{t+1} \mid a_{t+1:T}, y_{1:T}) \\ &\approx p(z_{t+1} \mid a_{t+1:T}, y_{1:T}) \frac{\sum_{i=1}^{N} w_t^i p(z_{t+1}, a_{t+1} \mid a_{1:t}^i, y_{1:t})}{\sum_{k=1}^{N} w_t^k p(z_{t+1}, a_{t+1} \mid a_{1:t}^k, y_{1:t})} \delta(a_{1:t} - a_{1:t}^i) \\ &= \sum_{i=1}^{N} p(z_{t+1} \mid a_{t+1:T}, y_{1:T}) w_{t|T}^i(z_{t+1}) \delta(a_{1:t} - a_{1:t}^i), \end{aligned} \tag{77}$$

where we have introduced the z_{t+1}-dependent weights

$$w_{t|T}^i(z_{t+1}) \triangleq \frac{w_t^i p(z_{t+1}, a_{t+1} \mid a_{1:t}^i, y_{1:t})}{\sum_{k=1}^{N} w_t^k p(z_{t+1}, a_{t+1} \mid a_{1:t}^k, y_{1:t})}. \tag{78}$$

To obtain (76) we can marginalize (77) over $a_{1:t-1}$ and z_{t+1}, which results in

$$p(a_t \mid a_{t+1:T}, y_{1:T}) = \sum_{i=1}^{N} \underbrace{\int p(z_{t+1} \mid a_{t+1:T}, y_{1:T}) w_{t|T}^i(z_{t+1})\, dz_{t+1}}_{=\theta^i} \delta(a_t - a_t^i). \tag{79}$$

Hence, if we want to sample "directly" from $p(a_t \mid a_{t+1:T}, y_{1:T})$ we need to evaluate the (likely to be intractable) integrals involved in (79). If we instead sample from the joint distribution $p(a_{1:t}, z_{t+1} \mid a_{t+1:T}, y_{1:T})$ we can use the fact that the marginal $p(z_{t+1} \mid a_{t+1:T}, y_{1:T})$ is Gaussian (and hence easy to sample from). We then only need to evaluate $w_{t|T}^i(z_{t+1})$ at a single point, which is clearly much simpler than evaluating the integrals in (79).

C Complementary computations

In this appendix we shall derive the equalities in (43) and (50). Using the Markov property and Bayes' rule we get

$$\begin{aligned}
p(a_{1:t} \mid z_{t+1}, a_{t+1:T}, y_{t:T}) &= p(a_{1:t} \mid z_{t+1}, a_{t+1}, a_{t+2:T}, y_{1:t}, y_{t+1:T}) \\
&= \frac{p(a_{t+2:T}, y_{t+1:T} \mid a_{1:t}, z_{t+1}, a_{t+1}, y_{1:t}) p(a_{1:t} \mid z_{t+1}, a_{t+1}, y_{1:t})}{p(a_{t+2:T}, y_{t+1:T} \mid z_{t+1}, a_{t+1}, y_{1:t})} \\
&= \frac{p(a_{t+2:T}, y_{t+1:T} \mid z_{t+1}, a_{t+1})}{p(a_{t+2:T}, y_{t+1:T} \mid z_{t+1}, a_{t+1})} p(a_{1:t} \mid z_{t+1}, a_{t+1}, y_{1:t}) \\
&= p(a_{1:t} \mid z_{t+1}, a_{t+1}, y_{1:t}), \qquad (80)
\end{aligned}$$

which gives (43). Furthermore

$$\begin{aligned}
p(z_t \mid z_{t+1}, a_{t:T}, y_{1:T}) &= p(z_t \mid z_{t+1}, a_t, a_{t+1}, a_{t+2:T}, y_{1:t}, y_{t+1:T}) \\
&= \frac{p(a_{t+2:T}, y_{t+1:T} \mid z_t, z_{t+1}, a_t, a_{t+1}, y_{1:t}) p(z_t \mid z_{t+1}, a_t, a_{t+1}, y_{1:t})}{p(a_{t+2:T}, y_{t+1:T} \mid z_{t+1}, a_t, a_{t+1}, y_{1:t})} \\
&= \frac{p(a_{t+2:T}, y_{t+1:T} \mid z_{t+1}, a_{t+1})}{p(a_{t+2:T}, y_{t+1:T} \mid z_{t+1}, a_{t+1})} p(z_t \mid z_{t+1}, a_t, a_{t+1}, y_{1:t}) \\
&= p(z_t \mid z_{t+1}, a_t, a_{t+1}, y_{1:t}), \qquad (81)
\end{aligned}$$

which proves (50). In the above computations we have assumed $t \leq T - 2$. For $t = T - 1$ simply remove $a_{t+2:T}$ from all steps and the equalities still hold.

References

[1] M. Briers, A. Doucet, and S. Maskell. Smoothing algorithms for state-space models. *Annals of the Institute of Statistical Mathematics*, 62(1):61–89, February 2010.

[2] A. Doucet, N. de Freitas, and N. Gordon, editors. *Sequential Monte Carlo Methods in Practice*. Springer Verlag, New York, USA, 2001.

[3] A. Doucet, S. J. Godsill, and C. Andrieu. On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing*, 10(3):197–208, 2000.

[4] A. Doucet and A. Johansen. A tutorial on particle filtering and smoothing: Fifteen years later. In *Handbook of Nonlinear Filtering (to appear)*. Oxford University Press, 2010.

[5] W. Fong, S. J. Godsill, A. Doucet, and M. West. Monte Carlo smoothing with application to audio signal enhancement. *IEEE Transactions on Signal Processing*, 50(2):438–449, February 2002.

[6] S. J. Godsill, A. Doucet, and M. West. Monte Carlo smoothing for nonlinear time series. *Journal of the American Statistical Association*, 99(465):156–168, March 2004.

[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*, 140(2):107 –113, April 1993.

[8] E. L. Lehmann. *Theory of Point Estimation*. Probability and mathematical statistics. John Wiley & Sons, New York, USA, 1983.

[9] F. Lindsten and T. B. Schön. Maximum likelihood estimation in mixed linear/nonlinear state-space models. In *Submitted to the 49th IEEE Conference on Decision and Control (CDC)*, 2010.

[10] H. E. Rauch, F. Tung, and C. T. Striebel. Maximum likelihood estimates of linear dynamic systems. *AIAA Journal*, 3(8):1445–1450, August 1965.

[11] T. B. Schön, F. Gustafsson, and P-J. Nordlund. Marginalized particle filters for mixed linear/nonlinear state-space models. *IEEE Transactions on Signal Processing*, 53(7):2279–2289, July 2005.

[12] T. B. Schön, A. Wills, and B. Ninness. System identification of nonlinear state-space models. *Provisionally accepted to Automatica*, 2010.

Avdelning, Institution
Division, Department

Division of Automatic Control
Department of Electrical Engineering

Datum
Date

2010-03-31

Språk
Language

☐ Svenska/Swedish
☒ Engelska/English
☐ ________________

Rapporttyp
Report category

☐ Licentiatavhandling
☐ Examensarbete
☐ C-uppsats
☐ D-uppsats
☒ Övrig rapport
☐ ________________

ISBN
—

ISRN
—

Serietitel och serienummer
Title of series, numbering

ISSN
1400-3902

LiTH-ISY-R-2946

URL för elektronisk version

`http://www.control.isy.liu.se`

Titel Title

Inference in Mixed Linear/Nonlinear State-Space Models using Sequential Monte Carlo

Författare Author

Fredrik Lindsten, Thomas B. Schön

Sammanfattning
Abstract

In this work we apply sequential Monte Carlo methods, i.e., particle filters and smoothers, to estimate the state in a certain class of mixed linear/nonlinear state-space models. Such a model has an inherent conditionally linear Gaussian substructure. By utilizing this structure we are able to address even high-dimensional nonlinear systems using Monte Carlo methods, as long as only a few of the states enter nonlinearly. First, we consider the filtering problem and give a self-contained derivation of the well known Rao-Blackwellized particle filter. Thereafter we turn to the smoothing problem and derive a Rao-Blackwellized particle smoother capable of handling the fully interconnected model under study.

Nyckelord
Keywords SMC, Particle filter, Particle smoother, Rao-Blackwellization