Ancestor Sampling for Particle Gibbs

Fredrik Lindsten
Div. of Automatic Control
Linköping University
lindsten@isy.liu.se

Michael I. Jordan
Dept. of EECS and Statistics
University of California, Berkeley
jordan@cs.berkeley.edu

Thomas B. Schön
Div. of Automatic Control
Linköping University
schon@isy.liu.se

Abstract

We present a novel method in the family of particle MCMC methods that we refer to as *particle Gibbs with ancestor sampling* (PG-AS). Similarly to the existing *PG with backward simulation* (PG-BS) procedure, we use backward sampling to (considerably) improve the mixing of the PG kernel. Instead of using separate forward and backward sweeps as in PG-BS, however, we achieve the same effect in a single forward sweep. We apply the PG-AS framework to the challenging class of non-Markovian state-space models. We develop a truncation strategy of these models that is applicable in principle to any backward-simulation-based method, but which is particularly well suited to the PG-AS framework. In particular, as we show in a simulation study, PG-AS can yield an order-of-magnitude improved accuracy relative to PG-BS due to its robustness to the truncation error. Several application examples are discussed, including Rao-Blackwellized particle smoothing and inference in degenerate state-space models.

1   Introduction

State-space models (SSMs) are widely used to model time series and dynamical systems. The strong assumptions of linearity and Gaussianity that were originally invoked in state-space inference have been weakened by two decades of research on sequential Monte Carlo (SMC) and Markov chain Monte Carlo (MCMC). These Monte Carlo methods have not, however, led to substantial weakening of a further strong assumption, that of Markovianity. It remains a major challenge to develop inference algorithms for non-Markovian SSMs:

$$x_{t+1} \sim f(x_{t+1} \mid \theta, x_{1:t}), \qquad\qquad y_t \sim g(y_t \mid \theta, x_{1:t}), \qquad\qquad (1)$$

where $\theta \in \Theta$ is a static parameter with prior density $p(\theta)$, x_t is the latent state and y_t is the observation at time t, respectively. Models of this form arise in many different application scenarios, either from direct modeling or via a transformation or marginalization of a larger model. We provide several examples in Section 5.

To tackle the challenging problem of inference for non-Markovian SSMs, we work within the framework of particle MCMC (PMCMC), a family of inferential methods introduced in [1]. The basic idea in PMCMC is to use SMC to construct a proposal kernel for an MCMC sampler. Assume that we observe a sequence of measurements $y_{1:T}$. We are interested in finding the density $p(x_{1:T}, \theta \mid y_{1:T})$, i.e., the joint posterior density of the state sequence and the parameter. In an idealized Gibbs sampler we would target this density by sampling as follows: *(i)* Draw $\theta^\star \mid x_{1:T} \sim p(\theta \mid x_{1:T}, y_{1:T})$; *(ii)* Draw $x_{1:T}^\star \mid \theta^\star \sim p(x_{1:T} \mid \theta^\star, y_{1:T})$. The first step of this procedure can be carried out exactly if conjugate priors are used. For non-conjugate models, one option is to replace Step *(i)* with a Metropolis-Hastings step. However, Step *(ii)*—sampling from the joint smoothing density $p(x_{1:T} \mid \theta, y_{1:T})$—is in most cases very difficult. In PMCMC, this is addressed by instead sampling a particle trajectory $x_{1:T}^\star$ based on an SMC approximation of the joint smoothing density. More precisely, we run an SMC sampler targeting $p(x_{1:T} \mid \theta^\star, y_{1:T})$. We then sample one of the particles

1

at the final time T, according to their importance weights, and trace the ancestral lineage of this particle to obtain the trajectory $x_{1:T}^\star$. This overall procedure is referred to as *particle Gibbs* (PG).

The flexibility provided by the use of SMC as a proposal mechanism for MCMC seems promising for tackling inference in non-Markovian models. To exploit this flexibility we must address a drawback of PG in the high-dimensional setting, which is that the mixing of the PG kernel can be very poor when there is path degeneracy in the SMC sampler [2, 3]. This problem has been addressed in the generic setting of SSMs by adding a backward simulation step to the PG sampler, yielding a method denoted *PG with backward simulation* (PG-BS). It has been found that this considerably improves mixing, making the method much more robust to a small number of particles as well as larger data records [2, 3].

Unfortunately, however, the application of backward simulation is problematic for non-Markovian models. The reason is that we need to consider full state trajectories during the backward simulation pass, leading to $O(T^2)$ computational complexity (see Section 4 for details). To address this issue, we develop a novel PMCMC method which we refer to as *particle Gibbs with ancestor sampling* (PG-AS) that achieves the effect of backward sampling without an explicit backward pass. As part of our development, we also develop a truncation method geared to non-Markovian models. This method is a generic method that is also applicable to PG-BS, but, as we show in a simulation study in Section 6, the effect of the truncation error is much less severe for PG-AS than for PG-BS. Indeed, we obtain up to an order of magnitude increase in accuracy in using PG-AS when compared to PG-BS in this study.

Since we assume that it is straightforward to sample the parameter θ of the idealized Gibbs sampler, we will not explicitly include sampling of θ in the subsequent sections to simplify our presentation.

2 Sequential Monte Carlo

We first review the standard auxiliary SMC sampler, see e.g. [4, 5]. Let $\gamma_t(x_{1:t})$ for $t = 1, \ldots, T$ be a sequence of unnormalized densities on X^t, which we assume can be evaluated pointwise in linear time. Let $\bar\gamma_t(x_{1:t})$ be the corresponding normalized probability densities. For an SSM we would typically have $\bar\gamma_t(x_{1:t}) = p(x_{1:t} \mid y_{1:t})$ and $\gamma_t(x_{1:t}) = p(x_{1:t}, y_{1:t})$. Assume that $\{x_{1:t-1}^m, w_{t-1}^m\}_{m=1}^N$ is a weighted particle system targeting $\bar\gamma_{t-1}(x_{1:t-1})$. This particle system is propagated to time t by sampling independently from a proposal kernel,

$$M_t(a_t, x_t) = \frac{w_{t-1}^{a_t}\nu_{t-1}^{a_t}}{\sum_l w_{t-1}^l\nu_{t-1}^l} R_t(x_t \mid x_{1:t-1}^{a_t}). \tag{2}$$

In this formulation, the resampling step is implicit and corresponds to sampling the ancestor indices a_t. Note that a_t^m is the index of the ancestor particle of x_t^m. When we write $x_{1:t}^m$ we refer to the ancestral path of x_t^m. The factors $\nu_t^m = \nu_t(x_{1:t}^m)$, known as adjustment multiplier weights, are used in the auxiliary SMC sampler to increase the probability of sampling ancestors that better can describe the current observation [5]. The particles are then weighted according to $w_t^m = W_t(x_{1:t}^m)$, where the weight function is given by

$$W_t(x_{1:t}) = \frac{\gamma_t(x_{1:t})}{\gamma_{t-1}(x_{1:t-1})\nu_{t-1}(x_{1:t-1})R_t(x_t \mid x_{1:t-1})}, \tag{3}$$

for $t \geq 2$. The procedure is initiated by sampling from a proposal density $x_1^m \sim R_1(x_1)$ and assigning importance weights $w_1^m = W_1(x_1^m)$ with $W_1(x_1) = \gamma_1(x_1)/R_1(x_1)$. In PMCMC it is instructive to view this sampling procedure as a way of generating a single sample from the density

$$\psi(\mathbf{x}_{1:T}, \mathbf{a}_{2:T}) \triangleq \prod_{m=1}^N R_1(x_1^m) \prod_{t=2}^T \prod_{m=1}^N M_t(a_t^m, x_t^m) \tag{4}$$

on the space $\mathsf{X}^{NT} \times \{1, \ldots, N\}^{N(T-1)}$. Here we have introduced the boldface notation $\mathbf{x}_t = \{x_t^1, \ldots, x_t^N\}$ and similarly for the ancestor indices.

3 Particle Gibbs with ancestor sampling

PMCMC methods is a class of MCMC samplers in which SMC is used to construct proposal kernels [1]. The validity of these methods can be assessed by viewing them as MCMC samplers on an

2

extended state space in which all the random variables generated by the SMC sampler are seen as auxiliary variables. The target density on this extended space is given by

$$\phi(\mathbf{x}_{1:T}, \mathbf{a}_{2:T}, k) \triangleq \frac{\bar{\gamma}_T(x_{1:T}^k)}{N^T} \frac{\psi(\mathbf{x}_{1:T}, \mathbf{a}_{2:T})}{R_1(x_1^{b_1}) \prod_{t=2}^{T} M_t(a_t^{b_t}, x_t^{b_t})}. \tag{5}$$

By construction, this density admits $\bar{\gamma}_T(x_{1:T}^k)$ as a marginal, and can thus be used as a surrogate for the original target density $\bar{\gamma}_T$ [1]. Here k is a variable indexing one of the particles at the final time point and $b_{1:T}$ corresponds to the ancestral path of this particle: $x_{1:T}^k = x_{1:T}^{b_{1:T}} = \{x_1^{b_1}, \ldots, x_T^{b_T}\}$. These indices are given recursively from the ancestor indices by $b_T = k$ and $b_t = a_{t+1}^{b_{t+1}}$. The PG sampler [1] is a Gibbs sampler targeting ϕ using the following sweep (note that $b_{1:T} = \{a_{2:T}^{b_{2:T}}, b_T\}$),

1. Draw $\mathbf{x}_{1:T}^{\star, -b_{1:T}}, \mathbf{a}_{2:T}^{\star, -b_{2:T}} \sim \phi(\mathbf{x}_{1:T}^{-b_{1:T}}, \mathbf{a}_{2:T}^{-b_{2:T}} \mid x_{1:T}^{b_{1:T}}, b_{1:T})$.
2. Draw $k^\star \sim \phi(k \mid \mathbf{x}_{1:T}^{\star, -b_{1:T}}, \mathbf{a}_{2:T}^{\star, -b_{2:T}}, x_{1:T}^{b_{1:T}}, a_{2:T}^{b_{2:T}})$.

Here we have introduced the notation $\mathbf{x}_t^{-m} = \{x_t^1, \ldots, x_t^{m-1}, x_t^{m+1}, \ldots, x_t^N\}$, $\mathbf{x}_{1:T}^{-b_{1:T}} = \{\mathbf{x}_1^{-b_1}, \ldots, \mathbf{x}_T^{-b_T}\}$ and similarly for the ancestor indices. In [1], a sequential procedure for sampling from the conditional density appearing in Step 1 is given. This method is known as *conditional SMC* (CSMC). It takes the form of an SMC sampler in which we condition on the event that a prespecified path $x_{1:T}^{b_{1:T}} = x'_{1:T}$, with indices $b_{1:T}$, is maintained throughout the sampler (see Algorithm 1 for a related procedure). Furthermore, the conditional distribution appearing in Step 2 of the PG sampler is shown to be proportional to w_T^k, and it can thus straightforwardly be sampled from.

Note that we never sample new values for the variables $\{x_{1:T}^{b_{1:T}}, b_{1:T-1}\}$ in this sweep. Hence, the PG sampler is an "incomplete" Gibbs sampler, since it does not loop over all the variables of the model. It still holds that the PG sampler is ergodic, which intuitively can be explained by the fact that the collection of variables that is left out is chosen randomly at each iteration. However, it has been observed that the PG sampler can have very poor mixing, especially when N is small and/or T is large [2, 3]. The reason for this poor mixing is that the SMC path degeneracy causes the collections of variables that are left out at any two consecutive iterations to be strongly dependent.

We now turn to our new procedure, PG-AS, which aims to address this fundamental issue. Our idea is to sample new values for the *ancestor indices* $b_{1:T-1}$ as part of the CSMC procedure[1]. By adding these variables to the Gibbs sweep, we can considerably improve the mixing of the PG kernel. The CSMC method is a sequential procedure to sample from $\phi(\mathbf{x}_{1:T}^{-b_{1:T}}, \mathbf{a}_{2:T}^{-b_{2:T}} \mid x_{1:T}^{b_{1:T}}, b_{1:T})$ by sampling according to $\{\mathbf{x}_t^{\star, -b_t}, \mathbf{a}_t^{\star, -b_t}\} \sim \phi(\mathbf{x}_t^{-b_t}, \mathbf{a}_t^{-b_t} \mid \mathbf{x}_{1:t-1}^{\star, -b_{1:t-1}}, \mathbf{a}_{2:t-1}^{\star, -b_{2:t-1}}, x_{1:T}^{b_{1:T}}, b_{1:T})$, for $t = 1, \ldots, T$. After having sampled these variables at time t, we add a step in which we generate a new value for $b_{t-1}(= a_t^{b_t})$, resulting in the following sweep:

1′. *(CSMC with ancestor sampling)* For $t = 1, \ldots, T$, draw
$$\mathbf{x}_t^{\star, -b_t}, \mathbf{a}_t^{\star, -b_t} \sim \phi(\mathbf{x}_t^{-b_t}, \mathbf{a}_t^{-b_t} \mid \mathbf{x}_{1:t-1}^{\star, -b_{1:t-1}}, \mathbf{a}_{2:t-1}^{\star}, x_{1:T}^{b_{1:T}}, b_{t-1:T}),$$
$$(a_t^{\star, b_t} =) b_{t-1}^\star \sim \phi(b_{t-1} \mid \mathbf{x}_{1:t-1}^{\star, -b_{1:t-1}}, \mathbf{a}_{2:t-1}^{\star}, x_{1:T}^{b_{1:T}}, b_{t:T}).$$

2′. Draw $(k^\star =) b_T^\star \sim \phi(b_T \mid \mathbf{x}_{1:T}^{\star, -b_{1:T}}, \mathbf{a}_{2:T}^{\star}, x_{1:T}^{b_{1:T}})$.

It can be verified that this corresponds to a partially collapsed Gibbs sampler [6] and will thus leave ϕ invariant. To determine the conditional densities from which the ancestor indices are drawn, consider the following factorization, following directly from (3),

$$\gamma_t(x_{1:t}) = W_t(x_{1:t})\nu_{t-1}(x_{1:t-1})R_t(x_t \mid x_{1:t-1})\gamma_{t-1}(x_{1:t-1})$$

$$\Rightarrow \gamma_t(x_{1:t}^{b_t}) = w_t^{b_t} \frac{\sum_l w_{t-1}^l \nu_{t-1}^l}{w_{t-1}^{b_{t-1}}} \frac{w_{t-1}^{b_{t-1}} \nu_{t-1}^{b_{t-1}}}{\sum_l w_{t-1}^l \nu_{t-1}^l} R_t(x_t^{b_t} \mid x_{1:t-1}^{b_{t-1}})\gamma_{t-1}(x_{1:t-1}^{b_{t-1}})$$

$$= \ldots = w_t^{b_t} \left(\prod_{s=1}^{t-1} \sum_l w_s^l \nu_s^l \right) R_1(x_1^{b_1}) \prod_{s=2}^{t} M_t(a_s^{b_s}, x_s^{b_s}). \tag{6}$$

---

[1]Ideally, we would like to include the variables $x_{1:T}^{b_{1:T}}$ as well, but this is in general not possible since it would be similar to sampling from the original target density (which we assume is infeasible).

3

Furthermore, we have

$$\phi(b_t \mid \mathbf{x}_{1:t}, \mathbf{a}_{2:t}, x_{t+1:T}^{b_{t+1:T}}, b_{t+1:T}) \propto \phi(\mathbf{x}_{1:t}, \mathbf{a}_{2:t}, x_{t+1:T}^{b_{t+1:T}}, b_{t:T})$$

$$\propto \frac{\gamma_T(x_{1:T}^k)\psi(\mathbf{x}_{1:t}, \mathbf{a}_{2:t})}{R_1(x_1^{b_1}) \prod_{s=2}^{t} M_s(a_s^{b_s}, x_s^{b_s})} \propto \frac{\gamma_t(x_{1:t}^{b_t})}{\gamma_t(x_{1:t}^{b_t})} \frac{\gamma_T(x_{1:T}^k)}{R_1(x_1^{b_1}) \prod_{s=2}^{t} M_s(a_s^{b_s}, x_s^{b_s})}. \tag{7}$$

By plugging (6) into the numerator we get,

$$\phi(b_t \mid \mathbf{x}_{1:t}, \mathbf{a}_{2:t}, x_{t+1:T}^{b_{t+1:T}}, b_{t+1:T}) \propto w_t^{b_t} \frac{\gamma_T(x_{1:T}^k)}{\gamma_t(x_{1:t}^{b_t})}. \tag{8}$$

Hence, to sample a new ancestor index for the conditioned path at time $t+1$, we proceed as follows. Given $x'_{t+1:T} (= x_{t+1:T}^{b_{t+1:T}})$ we compute the backward sampling weights,

$$w_{t|T}^m = w_t^m \frac{\gamma_T(\{x_{1:t}^m, x'_{t+1:T}\})}{\gamma_t(x_{1:t}^m)}, \tag{9}$$

for $m = 1, \ldots, N$. We then set $b_t = m$ with probability proportional to $w_{t|T}^m$.

It follows that the proposed CSMC with ancestor sampling (Step $1'$), conditioned on $\{x'_{1:T}, b_{1:T}\}$, can be realized as in Algorithm 1. The difference between this algorithm and the CSMC sampler derived in [1] lies in the ancestor sampling step 2(b) (where instead, they set $a_t^{b_t} = b_{t-1}$). By introducing the ancestor sampling, we break the strong dependence between the generated particle trajectories and the path on which we condition. We call the resulting method, defined by Steps $1'$ and $2'$ above, *PG with ancestor sampling* (PG-AS).

Algorithm 1 CSMC with ancestor sampling, conditioned on $\{x'_{1:T}, b_{1:T}\}$

---

1. **Initialize ($t = 1$):**
   (a) Draw $x_1^m \sim R_1(x_1)$ for $m \neq b_1$ and set $x_1^{b_1} = x'_1$.
   (b) Set $w_1^m = W_1(x_1^m)$ for $m = 1, \ldots, N$.
2. **for** $t = 2, \ldots, T$**:**
   (a) Draw $\{a_t^m, x_t^m\} \sim M_t(a_t, x_t)$ for $m \neq b_t$ and set $x_t^{b_t} = x'_t$.
   (b) Draw $a_t^{b_t}$ with $P(a_t^{b_t} = m) \propto w_{t-1|T}^m$.
   (c) Set $x_{1:t}^m = \{x_{1:t-1}^{a_t^m}, x_t^m\}$ and $w_t^m = W_t(x_{1:t}^m)$ for $m = 1, \ldots, N$.

The idea of including the variables $b_{1:T-1}$ in the PG sampler has previously been suggested by Whiteley [7] and further explored in [2, 3]. This previous work, however, accomplishes this with a explicit backward simulation pass, which, as we discuss in the following section, is problematic for our applications to non-Markovian SSMs. In the PG-AS sampler, instead of requiring distinct forward and backward sequences of Gibbs steps as in PG with backward simulation (PG-BS), we obtain a similar effect via a single forward sweep.

4 Truncation for non-Markovian state-space models

We return to the problem of inference in non-Markovian SSMs of the form shown in (1). To employ backward sampling, we need to evaluate the ratio

$$\frac{\gamma_T(x_{1:T})}{\gamma_t(x_{1:t})} = \frac{p(x_{1:T}, y_{1:T})}{p(x_{1:t}, y_{1:t})} = \prod_{s=t+1}^{T} g(y_s \mid x_{1:s})f(x_s \mid x_{1:s-1}). \tag{10}$$

In general, the computational cost of computing the backward sampling weights will thus be $O(T)$. This implies that the cost of generating a full backward trajectory is $O(T^2)$. It is therefore computationally prohibitive to employ backward simulation type of particle smoothers, as well as the PG samplers discussed above, for general non-Markovian models.

4
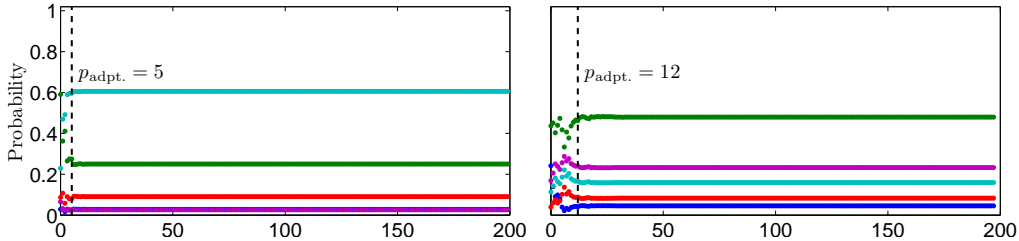
Figure 1: Probability under $\widetilde{P}_p$ as a function of the truncation level p for two different systems; one 5 dimensional (left) and one 20 dimensional (right). The $N = 5$ dotted lines correspond to $\widetilde{P}_p(m)$ for $m \in \{1, \ldots, N\}$, respectively (N.B. two of the lines overlap in the left figure). The dashed vertical lines show the value of the truncation level $p_{\text{adpt.}}$, resulting from the adaption scheme with $\gamma = 0.1$ and $\tau = 10^{-2}$. See Section 6.2 for details on the experiments.

To make progress, we consider non-Markovian models in which there is a decay in the influence of the past on the present, akin to that in Markovian models but without the strong Markovian assumption. Hence, it is possible to obtain a useful approximation when the product in (10) is truncated to a smaller number of factors, say p. We then replace (9) with the approximation,

$$\widetilde{w}_{t|T}^{p,m} = w_t^m \frac{\gamma_{t+p}(\{x_{1:t}^m, x'_{t+1:t+p}\})}{\gamma_t(x_{1:t}^m)}. \tag{11}$$

The following proposition formalizes our assumption.

Proposition 1. *Let P and $\widetilde{P}_p$ be the probability distributions on $\{1, \ldots, N\}$, defined by the backward sampling weight (9) and the truncated backward sampling weights (11), respectively. Let $h_s(k) = g(y_{t+s} \mid x_{1:t}^k, x'_{t+1:t+s}) f(x'_{t+s} \mid x_{1:t}^k, x'_{t+1:t-s})$ and assume that $\max_{k,l} (h_s(k)/h_s(l) - 1) \leq A \exp(-cs)$, for some constants A and $c > 0$. Then, $D_{\mathsf{KLD}}(P\|\widetilde{P}_p) \leq C \exp(-cp)$ for some constant C, where D_{KLD} is the Kullback-Leibler divergence (KLD).*

Proof. Provided in the supplemental material. □

From (11), we see that we can compute the backward weights in constant time under the truncation within the PG-AS framework. The resulting approximation can be quite useful; indeed, in our experiments we have seen that even $p = 1$ can lead to very accurate inferential results. In general, however, it will not be known a priori how to set the truncation level p for any given problem. To address this problem, we propose to use an adaption of the truncation level. Since the approximative weights (11) can be evaluated sequentially, the idea is to start with $p = 1$ and then increase p until the weights have, in some sense, converged. In particular, in our experimental work, we have used the following simple approach.

Let $\widetilde{P}_p$ be the discrete probability measure defined by (11). Let $\varepsilon_p = D_{\mathsf{TV}}(\widetilde{P}_p, \widetilde{P}_{p-1})$ be the total variation (TV) distance between the distributions for two consecutive truncation levels. We then compute the exponentially decaying moving average of the sequence ε_p, with forgetting factor $\gamma \in [0, 1]$, and stop when this falls below some threshold $\tau \in [0, 1]$. This adaption scheme removes the requirement to specify p directly, but instead introduces the design parameters γ and τ. However, these parameters are much easier to reason about – a small value for γ gives a rapid response to changes in ε_p whereas a large value gives a more conservative stopping rule, improving the accuracy of the approximation at the cost of higher computational complexity. A similar trade off holds for the threshold τ as well. Most importantly, we have found that the same values for γ and τ can be used for a wide range of models, with very different mixing properties.

To illustrate the effect of the adaption rule, and how the distribution $\widetilde{P}_p$ typically evolves as we increase p, we provide two examples in Figure 1. These examples are taken from the simulation study provided in Section 6.2. Note that the untruncated distribution P is given for the maximal value of p, i.e., furthest to the right in the figures. By using the adaptive truncation, we can stop the evaluation of the weights at a much earlier stage, and still obtain an accurate approximation of P.

5

5 Application areas

In this section we present examples of problem classes involving non-Markovian SSMs for which the proposed PG-AS sampler can be applied. Numerical illustrations are provided in Section 6.

5.1 Rao-Blackwellized particle smoothing

One popular approach to increase the efficiency of SMC samplers for SSMs is to marginalize over one component of the state, and apply an SMC sampler in the lower-dimensional marginal space. This leads to what is known as the Rao-Blackwellized particle filter (RBPF) [8–10]. The same approach has also been applied to state smoothing [11,12], but it turns out that Rao-Blackwellization is less straightforward in this case, since the marginal state-process will be non-Markovian. As an example, a mixed linear/nonlinear Gaussian SSM (see, e.g., [10]) with "nonlinear state" x_t and "linear state" z_t, can be reduced to $x_t \sim p(x_t \mid x_{1:t-1}, y_{1:t-1})$ and $y_t \sim p(y_t \mid x_{1:t}, y_{1:t-1})$. These conditional densities are Gaussian and can be evaluated for any fixed marginal state trajectory $x_{1:t-1}$ by running a conditional Kalman filter to marginalize the z_t-process.

In order to apply a backward-simulation-based method (e.g., a particle smoother) for this model, we need to evaluate the backward sampling weights (9). In a straightforward implementation[2], we thus need to run N Kalman filters for $T - t$ time steps, for each $t = 1, \ldots, T - 1$. The computational complexity of this calculation can be reduced by employing the truncation proposed in Section 4.

5.2 Particle smoothing for degenerate state-space models

Many dynamical systems are most naturally modelled as degenerate in the sense that the transition kernel of the state process does not admit any dominating measure. For instance, consider a nonlinear system with additive noise of the form,

$$\xi_t = f(\xi_{t-1}) + G\omega_{t-1}, \qquad\qquad y_t = g(\xi_t) + e_t, \qquad\qquad (12)$$

where G is a tall matrix, and consequently $\operatorname{rank}(G) < \dim(\xi_t)$. That is, the process noise covariance matrix is singular. SMC samplers can straightforwardly be applied to this type of models, but it is more problematic to address the smoothing problem using particle methods. The reason is that the backward kernel also will be degenerate and it cannot be approximated in a natural way by the forward filter particles, as is normally done in backward-simulation-based particle smoothers.

A possible remedy for this issue is to recast the degenerate SSM as a non-Markovian model in a lower-dimensional space. Let $G = U \begin{bmatrix} \Sigma & 0 \end{bmatrix}^{\mathsf{T}} V^{\mathsf{T}}$ with unitary U and V be a singular value decomposition of G and let,

$$\begin{bmatrix} x_t \\ z_t \end{bmatrix} \triangleq U^{\mathsf{T}} \xi_t = U^{\mathsf{T}} f(UU^{\mathsf{T}} \xi_{t-1}) + \begin{bmatrix} \Sigma V^{\mathsf{T}} \omega_{t-1} \\ 0 \end{bmatrix}. \qquad\qquad (13)$$

For simplicity we assume that z_1 is known. If this is not the case, it can be included in the system state or seen as a static parameter of the model. Hence, the sequence $z_{1:t}$ is $\sigma(x_{1:t-1})$-measurable and we can write $z_t = z_t(x_{1:t-1})$. With $v_t \triangleq \Sigma V^{\mathsf{T}} \omega_t$ and by appropriate definitions of the functions f_x and h, the model (12) can thus be rewritten as, $x_t = f_x(x_{1:t-1}) + v_{t-1}$ and $y_t = h(x_{1:t}) + e_t$, which is a non-degenerate, non-Markovian SSM. By exploiting the truncation proposed in Section 4 we can thus apply PG-AS to do inference in this model.

5.3 Additional problem classes

There are many more problem classes in which non-Markovian models arise and in which backward-simulation-based methods can be of interest. For instance, the Dirichlet process mixture model (DPMM, see, e.g., [13]) is a popular nonparametric Bayesian model for mixtures with an unknown number of components. Using a Polya urn representation, the mixture labels are given by a non-Markovian stochastic process, and the DPMM can thus be seen as a non-Markovian SSM. SMC has

---

[2]For the specific problem of Rao-Blackwellized smoothing in conditionally Gaussian models, a backward simulator which can be implemented in $O(T)$ computational complexity has recently been proposed in [11]. This is based on the idea of propagating information backward in time as the backward samples are generated.

6
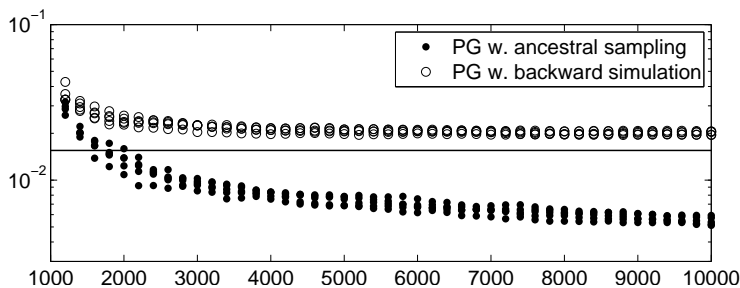
Figure 2: Rao-Blackwellized state smoothing using PG. Running RMSEs for five independent runs of PG-AS (●) and PG-BS (○), respectively. The truncation level is set to $p = 1$. The solid line corresponds to a run of an untruncated FF-BS.

previously been used for inference in DPMMs [14, 15]. An interesting venue for future work is to use the PG-AS sampler for these models. A second example in Bayesian nonparametrics is Gaussian process (GP) regression and classification (see, e.g., [16]). The sample path of the GP can be seen as the state-process in a non-Markovian SSM. We can thus employ PMCMC, and in particular PG-AS, to address these inference problems.

An application in genetics, for which SMC has been been successfully applied, is reconstruction of phylogenetic trees [17]. A phylogenetic tree is a binary tree with observation at the leaf nodes. SMC is used to construct the tree in a bottom up fashion. A similar approach has also been used for Bayesian agglomerative clustering, in which SMC is used to construct a binary clustering tree based on Kingman's coalescent [18]. The generative models for the trees used in [17, 18] are in fact Markovian, but the observations give rise to a conditional dependence which destroys the Markov property. To employ backward simulation to these models, we are thus faced with problems of a similar nature as those discussed in Section 4.

6 Numerical evaluation

This section contains a numerical evaluation of the proposed method. We consider linear Gaussian systems, which is instructive since the exact smoothing density then is available, e.g., by running a modified Bryson-Frazier (MBF) smoother [19]. For more details on the experiments, and for additional (nonlinear) examples, see [20].

6.1 RBPS: Linear Gaussian state-space model

As a first example, we consider Rao-Blackwellized particle smoothing (RBPS) in a single-output 4th-order linear Gaussian SSM. We generate $T = 100$ samples from the system and run PG-AS and PG-BS, marginalizing three out of the four states using an RBPF, i.e., $\dim(x_t) = 1$. Both methods are run for $R = 10000$ iterations using $N = 5$ particles. The truncation level is set to $p = 1$, leading to a coarse approximation. We discard the first 1000 iterations and then compute running means of the state trajectory $x_{1:T}$. From these, we then compute the running root mean squared errors (RMSEs) ϵ_r *relative to the true posterior means* (computed with an MBF smoother). Hence, if no approximation would have been made, we would expect $\epsilon_r \to 0$, so any static error can be seen as the effect of the truncation. The results for five independent runs from both PG samplers are shown in Figure 2. First, we note that both methods give accurate results. Still, the error for PG-AS is close to an order of magnitude less than for PG-BS. Furthermore, it appears as if the error for PG-AS would decrease further, given more iterations, suggesting that the bias caused by the truncation is dominated by the Monte Carlo variance, even after $R = 10000$ iterations.

For further comparison, we also run an untruncated forward filter/backward simulator (FF-BS) particle smoother [21], using $N = 5000$ forward filter particles and $M = 500$ backward trajectories (with a computational complexity of $O(NMT^2)$). The resulting RMSE value is shown as a solid line in Figure 2. These results suggest that PMCMC samplers, such as the PG-AS, indeed can be serious competitors to more "standard" particle smoothers. Even with $p = 1$, PG-AS outperforms

7
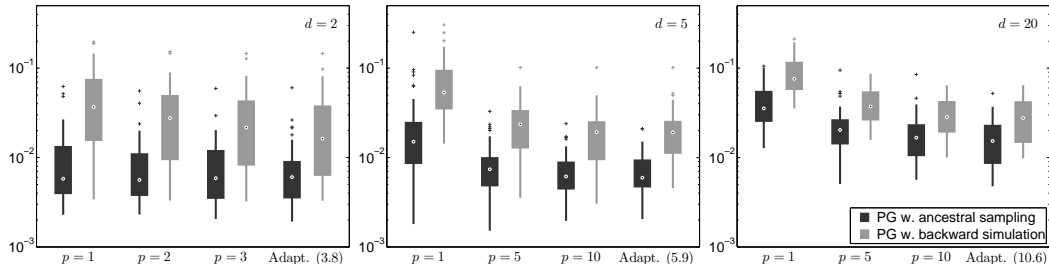
Figure 3: Box plots of the RMSE errors for PG-AS (black) and PG-BS (gray), for 150 random systems of different dimensions d (left, $d = 2$; middle, $d = 5$; right, $d = 20$). Different values for the truncation level p are considered. The rightmost boxes correspond to an adaptive threshold and the values in parentheses are the average over all systems and MCMC iterations (the same for both methods). The dots within the boxes show the median errors.

FF-BS in terms of accuracy and, due to the fact that the ancestor sampling allows us to use as few as $N = 5$ particles at each iteration, at a lower computational cost.

6.2 Random linear Gaussian systems with rank deficient process noise covariances

To see how the PG samplers are affected by the choice of truncation level p and by the mixing properties of the system, we evaluate them on random linear Gaussian SSMs of different orders. We generate 150 random systems, using the MATLAB function `drss` from the Control Systems Toolbox, with model orders 2, 5 and 20 (50 systems for each model order). The number of outputs are taken as 1, 2 and 4 for the different model orders, respectively. The systems are then simulated for $T = 200$ time steps, driven by Gaussian process noise entering only on the first state component. Hence, the rank of the process noise covariance is 1 for all systems.

We run the PG-AS and PG-BS samplers for 10000 iterations using $N = 5$ particles. We consider different fixed truncation levels, as well as an adaptive level with $\gamma = 0.1$ and $\tau = 10^{-2}$. Again, we compute running posterior means (discarding 1000 samples) and RMSE values relative the true posterior mean. Box plots are shown in Figure 3. Since the process noise only enters on one of the state components, the mixing tends to deteriorate as we increase the model order. Figure 1 shows how the probability distributions on $\{1, \ldots, N\}$ change as we increase the truncation level, in two representative cases for a 5th and a 20th order system, respectively. By using an adapted level, we can obtain accurate results for systems of different dimensions, without having to change any settings between the runs.

7 Discussion

PG-AS is a novel approach to PMCMC that makes use of backward simulation ideas without needing an explicit backward pass. Compared to PG-BS, a conceptually similar method that does require an explicit backward pass, PG-AS has advantages, most notably for inference in the non-Markovian SSMs that have been our focus here. When using the proposed truncation of the backward weights, we have found PG-AS to be more robust to the approximation error than PG-BS. Furthermore, for non-Markovian models, PG-AS is easier to implement than PG-BS, since it requires less bookkeeping. It can also be more memory efficient, since it does not require us to store intermediate quantities that are needed for a separate backward simulation pass, as is done in PG-BS. Finally, we note that PG-AS can be used as an alternative to PG-BS for other inference problems to which PMCMC can be applied, and we believe that it will prove attractive in problems beyond the non-Markovian SSMs that we have discussed here.

Acknowledgments

This work was supported by: the project Calibrating Nonlinear Dynamical Models (Contract number: 621-2010-5876) funded by the Swedish Research Council and CADICS, a Linneaus Center also funded by the Swedish Research Council.

8

References

[1] C. Andrieu, A. Doucet, and R. Holenstein, "Particle Markov chain Monte Carlo methods," *Journal of the Royal Statistical Society: Series B*, vol. 72, no. 3, pp. 269–342, 2010.

[2] N. Whiteley, C. Andrieu, and A. Doucet, "Efficient Bayesian inference for switching state-space models using discrete particle Markov chain Monte Carlo methods," Bristol Statistics Research Report 10:04, Tech. Rep., 2010.

[3] F. Lindsten and T. B. Schön, "On the use of backward simulation in the particle Gibbs sampler," in *Proceedings of the 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Kyoto, Japan, Mar. 2012.

[4] A. Doucet and A. Johansen, "A tutorial on particle filtering and smoothing: Fifteen years later," in *The Oxford Handbook of Nonlinear Filtering*, D. Crisan and B. Rozovsky, Eds. Oxford University Press, 2011.

[5] M. K. Pitt and N. Shephard, "Filtering via simulation: Auxiliary particle filters," *Journal of the American Statistical Association*, vol. 94, no. 446, pp. 590–599, 1999.

[6] D. A. V. Dyk and T. Park, "Partially collapsed Gibbs samplers: Theory and methods," *Journal of the American Statistical Association*, vol. 103, no. 482, pp. 790–796, 2008.

[7] N. Whiteley, "Discussion on Particle Markov chain Monte Carlo methods," Journal of the Royal Statistical Society: Series B, 72(3), p 306–307, 2010.

[8] R. Chen and J. S. Liu, "Mixture Kalman filters," *Journal of the Royal Statistical Society: Series B*, vol. 62, no. 3, pp. 493–508, 2000.

[9] A. Doucet, S. J. Godsill, and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," *Statistics and Computing*, vol. 10, no. 3, pp. 197–208, 2000.

[10] T. Schön, F. Gustafsson, and P.-J. Nordlund, "Marginalized particle filters for mixed linear/nonlinear state-space models," *IEEE Transactions on Signal Processing*, vol. 53, no. 7, pp. 2279–2289, Jul. 2005.

[11] S. Särkkä, P. Bunch, and S. 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, Jul. 2012.

[12] W. Fong, S. J. Godsill, A. Doucet, and M. West, "Monte Carlo smoothing with application to audio signal enhancement," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 438–449, Feb. 2002.

[13] N. L. Hjort, C. Holmes, P. Mller, and S. G. Walker, Eds., *Bayesian Nonparametrics*. Cambridge University Press, 2010.

[14] S. N. MacEachern, M. Clyde, and J. S. Liu, "Sequential importance sampling for nonparametric Bayes models: The next generation," *The Canadian Journal of Statistics*, vol. 27, no. 2, pp. 251–267, 1999.

[15] P. Fearnhead, "Particle filters for mixture models with an unknown number of components," *Statistics and Computing*, vol. 14, pp. 11–21, 2004.

[16] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[17] A. Bouchard-Côté, S. Sankararaman, and M. I. Jordan, "Phylogenetic inference via sequential Monte Carlo," *Systematic Biology*, vol. 61, no. 4, pp. 579–593, 2012.

[18] Y. W. Teh, H. Daumé III, and D. Roy, "Bayesian agglomerative clustering with coalescents," *Advances in Neural Information Processing*, pp. 1473–1480, 2008.

[19] G. J. Bierman, "Fixed interval smoothing with discrete measurements," *International Journal of Control*, vol. 18, no. 1, pp. 65–75, 1973.

[20] F. Lindsten, M. I. Jordan, and T. B. Schön, "Ancestor sampling for particle Gibbs," arXiv.org, arXiv:1210.6911, Oct. 2012.

[21] S. J. Godsill, A. Doucet, and M. West, "Monte Carlo smoothing for nonlinear time series," *Journal of the American Statistical Association*, vol. 99, no. 465, pp. 156–168, Mar. 2004.

9