

PARTICLE GIBBS WITH REFRESHED BACKWARD SIMULATION

Pete Bunch Fredrik Lindsten Sumeetpal Singh

Signal Processing and Communications Laboratory
Department of Engineering, University of Cambridge
Trumpington Street, Cambridge, UK, CB2 1PZ

ABSTRACT

The particle Gibbs algorithm can be used for Bayesian parameter estimation in Markovian state space models. Sometimes the resulting Markov chains mix slowly when the component particle filter suffers from degeneracy. This effect can be somewhat alleviated using backward simulation. In this paper we show how a simple modification to this scheme, which we refer to as refreshed backward simulation, can further improve the mixing. This works by sampling new state values simultaneously with the corresponding ancestor indexes. Although the necessary conditional distributions cannot be sampled directly, we provide suitable Markov kernels which target them. The efficacy of this new scheme is demonstrated with a simulation example.

Index Terms— Sequential Monte Carlo, particle Markov chain Monte Carlo, Gibbs sampling, backward simulation

1. INTRODUCTION

Particle Markov chain Monte Carlo (PMCMC) algorithms [1, 2, 3, 4] provide an elegant and effective solution for Bayesian parameter learning with Markovian state space models. They are based on the formulation of an extended target distribution over the system of random variables comprising a particle filter, which has the desired posterior distribution as a marginal. A Markov chain for which this extended distribution is invariant may be constructed using sequential Monte Carlo (SMC) components. The particle system is composed of a set of states for each time step and corresponding ancestor index variables, which define a set of state trajectories.

In this paper we consider in particular the *particle Gibbs* (PG) algorithm, introduced by [1]. This samples in turn new values for the unknown parameters, the particle system, and an index variable indicating one reference trajectory. Sampling the particle system is equivalent to running a modified particle filter. Like any Gibbs sampler, this has the advantage over Metropolis-Hastings of not requiring an accept/reject stage. However, the resulting chains are still liable to mix slowly if the particle filter suffers from path-space degeneracy.

It is possible to reduce degeneracy, and thus improve the mixing of the PG Markov chain, by incorporating additional sampling steps, either during the filtering stage, known as particle Gibbs with ancestor sampling (PG-AS) [4], or in an additional backward sweep, known as particle Gibbs with backward simulation (PG-BS) [5, 6]. The improvement arises from sampling new values for individual ancestor indexes, and thus allowing the reference trajectory to be updated gradually rather than changing it all at once.

For near-degenerate models, mixing may be slow even when using PG-BS or PG-AS. Specifically, when the model transition distribution is highly informative, the probability of sampling any change

in the particle ancestry is low. Intuitively, the problem is that the only state history consistent with a particular future is that from which the future was originally generated. We can mitigate this effect by using a modified procedure which we call *refreshed backward simulation* [7, 8]. When sampling an ancestor index for the reference trajectory, we simultaneously sample a new value for the associated state. This allows us some leeway to steer the potential state histories towards the fixed future, consequently increasing the probability of changing the ancestry and thus improving the mixing of the Markov chain.

A recent paper [9] presents ideas which have some overlap with our work. Specifically, they use Markov chain Monte Carlo during a backward sweep to sample new state values. However, they introduce a different extended target distribution in order to justify this addition, which necessitates changes to the conditional particle filter, whereas we only use the standard PMCMC distribution.

1.1. State Space Modelling and the Particle Filter

We consider a standard Markovian state space model with a sequence of latent states $x_t \in \mathcal{X} : t = 1, \dots, T$, and a corresponding sequence of observations $y_t \in \mathcal{Y} : t = 1, \dots, T$. We assume that the transition and observation distributions have associated densities with respect to some appropriate measure (e.g. Lebesgue),

$$x_t | x_{t-1} \sim f_{\theta,t}(x_t | x_{t-1}) \quad y_t | x_t \sim g_{\theta,t}(y_t | x_t).$$

We use the convention that $f_{\theta,1}(x_1 | x_0) = f_{\theta,1}(x_1)$ is the prior density of the first state. The variable $\theta \in \Theta$ is a collection of unknown model parameters upon which $f_{\theta,t}$ and $g_{\theta,t}$ depend, which has a prior density $p(\theta)$.

Our objective is to approximate the joint posterior density over all the unknown variables,

$$p(\theta, x_{1:T} | y_{1:T}) = \frac{1}{Z} p(\theta) \prod_{t=1}^T g_{\theta,t}(y_t | x_t) f_{\theta,t}(x_t | x_{t-1}), \quad (1)$$

where Z is a normalising constant.

The particle filter is a sequential Monte Carlo algorithm which recursively approximates the sequence of densities $p(x_{1:t} | \theta, y_{1:t}) : t = 1, \dots, T$. This is achieved by propagating forwards a collection of N particles $\{x_{1:t}^{(i)} : i = 1, \dots, N\}$, each of which is a realisation of the state sequence, along with a set of associated weights $\{w_t^{(i)} : i = 1, \dots, N\}$. This procedure is well established. See e.g. [10, 11, 1, 6] for details of particle filters and their use as a component in PMCMC schemes.

Particle filters exhibit a significant deficiency known as path-space degeneracy. Only a subset of the particles at each time instant are used in the construction of those at the next time instant. This means that the number of unique states appearing in the trajectories

decreases as we look back in time. If T is sufficiently large, then there will be a time step before which every particle is the same [12].

2. PARTICLE GIBBS

An ideal Gibbs sampler for targeting (1) might alternately sample from the state and parameter conditional distributions, $p(x_{1:T}|\theta, y_{1:T})$ and $p(\theta|x_{1:T}, y_{1:T})$. We assume here that the parameter conditional is straightforward to sample from, either directly using conjugate priors or with Metropolis-Hastings (MH). Sampling from the state conditional is the more challenging step. This can rarely be achieved directly. A particle filter could be used to return an approximately distributed sample, but the resulting algorithm will not have the correct target distribution because of this approximation. The approach used by particle Gibbs is to construct an extended distribution over all the random variables comprising a particle filter. This may be targeted without approximation, and admits the desired posterior as a marginal.

The PMCMC extended target distribution is constructed over the space of an entire particle system. This comprises states and ancestor indexes,

$$\begin{aligned} \mathbf{x}_t &= \{x_t^{(i)} : i = 1, \dots, N\} \quad t = 1, \dots, T \\ \mathbf{a}_t &= \{a_t^{(i)} : i = 1, \dots, N\} \quad t = 2, \dots, T. \end{aligned}$$

The ancestor index $a_t^{(i)} \in \{1, \dots, N\}$ indicates the $(t-1)$ parent state from which $x_t^{(i)}$ follows. Hence, each particle state trajectory is constructed recursively by tracing the lineage described by these ancestor indexes,

$$x_{1:t}^{(i)} = x_{1:t-1}^{(a_t^{(i)})} \cup x_t^{(i)}.$$

Furthermore, let $K \in \{1, \dots, N\}$ be the index of one particular reference trajectory, and indicate the ancestry of this particle by,

$$b_t = \begin{cases} K & t = T \\ a_{t+1}^{(b_{t+1})} & t = T-1, \dots, 1. \end{cases}$$

For this reference trajectory write,

$$x_{1:T}^{(b_{1:T})} = \{x_t^{(b_t)} : t = 1, \dots, T\},$$

and for the remaining states,

$$x_{1:T}^{(-b_{1:T})} = \mathbf{x}_{1:T} \setminus x_{1:T}^{(b_{1:T})}.$$

The extended target distribution may now be written as,

$$\begin{aligned} \pi(\theta, \mathbf{a}_{2:T}, \mathbf{x}_{1:T}, K) &= \frac{1}{N^T} p(\theta, x_{1:T}^{(b_{1:T})} | y_{1:T}) \\ &\times \prod_{i \neq b_1} q_1(x_1^{(i)}) \prod_{t=2}^T \left[\prod_{i \neq b_t} \frac{w_{t-1}^{(a_t^{(i)})}}{\sum_j w_{t-1}^{(j)}} q_t(x_t^{(i)} | x_{t-1}^{(a_t^{(i)})}) \right], \quad (2) \end{aligned}$$

in which the unnormalised importance weights are,

$$w_t^{(i)} = f_{\theta,t}(x_t^{(i)} | x_{t-1}^{(a_t^{(i)})}) g_{\theta,t}(y_t | x_t^{(i)}) / q_t(x_t^{(i)} | x_{t-1}^{(a_t^{(i)})}) \quad (3)$$

and $\{q_t\}$ are importance densities. These may depend on the observation sequence $y_{1:T}$, and the same convention regarding q_1 is used as for the $f_{\theta,1}$. By construction the extended target distribution has the desired posterior as a marginal.

During each step of the particle Gibbs algorithm new values are sampled in turn from appropriate conditional posterior distributions for θ , then $\{\mathbf{a}_{2:T}^{(-b_{2:T})}, \mathbf{x}_{1:T}^{(-b_{1:T})}\}$, and finally K .

The conditional for the non-reference particles may, by construction, be sampled sequentially forwards in time. This procedure is known as a *conditional particle filter*, since it consists of the same operations as a standard particle filter, but for one ancestor-state pair at each time step which is set deterministically to be equal to that of the reference trajectory.

The conditional for the reference trajectory index is [1],

$$\pi(K|\theta, \mathbf{a}_{2:T}, \mathbf{x}_{1:T}) = \frac{w_T^{(K)}}{\sum_j w_T^{(j)}}.$$

Thus, an index is sampled by normalising the final particle filter weights and then drawing once from the resulting categorical distribution.

3. PARTICLE GIBBS WITH BACKWARD SIMULATION

Mixing of the particle Gibbs algorithm can be very slow. This can be seen as a failing of the conditional particle filter. The reference trajectory is guaranteed to appear in the final particle system. If the system suffers from path-space degeneracy then the old and new reference trajectories are likely to have a near-identical ancestry, with differences only appearing towards the end of the sequence. The Markov chain required for the early states to converge will be impractically long.

3.1. Standard Backward Simulation

As suggested by [5], this problem may be mitigated by including an additional sampling stage in each step of the PG algorithm. Sweeping backwards, for each time step a new ancestor index is drawn from,

$$\begin{aligned} \pi(a_t^{(b_t)} | \theta, \mathbf{a}_{2:t-1}, \mathbf{x}_{1:t-1}, a_{t+1:T}^{(b_{t+1:T})}, x_{t:T}^{(b_{t:T})}, K) \\ = \frac{w_{t-1}^{(a_t^{(b_t)})} f_{\theta,t}(x_t^{(b_t)} | x_{t-1}^{(a_t^{(b_t)})})}{\sum_j w_{t-1}^{(j)} f_{\theta,t}(x_t^{(b_t)} | x_{t-1}^{(j)})}. \quad (4) \end{aligned}$$

Note that each of these operations is a *collapsed* Gibbs move, meaning that some of the variables are marginalised before conditioning. Specifically, we have marginalised the future states and ancestors other than those in the reference trajectory. Marginalisation is commonly used for Gibbs samplers to improve the mixing and to simplify the implementation, and if correctly formulated will not alter the stationary distribution; see [13] for details.

Algorithmically, this additional stage corresponds to backward simulation [14]. The sampler sweeps backwards through time, sampling a new value for each ancestor index $a_t^{(b_t)}$ from a set of smoothing weights proportional to $w_t^{(i)} f_{\theta,t}(x_t^{(b_t)} | x_{t-1}^{(i)})$. Backward simulation within PG was suggested by [5], and explored by [6], although in the latter case using a modified extended target distribution.

3.2. Refreshed Backward Simulation

Backward simulation allows the sampler to change the ancestry of the reference trajectory even when the conditional particle filter suffers from degeneracy. However, if the model transition density is

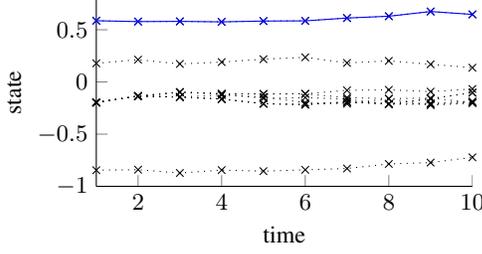


Fig. 1. An illustration of ineffective backward simulation on a toy linear-Gaussian problem. Crosses indicate particle filter states, and dotted lines the ancestries. The reference particle is shown with a solid line. A backward simulation sweep does not result in any changes to the reference particle ancestry.

tightly concentrated in one area of the state space (e.g. if the variance $\mathbb{V}[x_t|x_{t-1}]$ is small) then the probability of changing the ancestor indexes may be very low. See Fig 1. If this situation arises, then the ability of backward simulation to mitigate the problems of particle degeneracy and accelerate the mixing of PG can be limited.

We can increase the chances of altering the ancestry, and thus further improve mixing of the Markov chain, if the backward simulation algorithm is modified to simultaneously sample a new value for each state along with the corresponding ancestor index. At each time instant we now sample from,

$$\begin{aligned} \pi(a_t^{(b_t)}, x_t^{(b_t)} | \theta, \mathbf{a}_{2:t-1}, \mathbf{x}_{1:t-1}, a_{t+1:T}^{(b_{t+1:T})}, x_{t+1:T}^{(b_{t+1:T})}, K) \\ = \frac{w_{t-1}^{(a_t^{(b_t)})} f_{\theta,t}(x_t^{(b_t)} | x_{t-1}^{(a_{t-1}^{(b_t)})}) g_{\theta,t}(y_t | x_t^{(b_t)}) f_{\theta,t+1}(x_{t+1}^{(b_{t+1})} | x_t^{(b_t)})}{\sum_j w_{t-1}^{(j)} \int f_{\theta,t}(x_t | x_{t-1}^{(j)}) g_{\theta,t}(y_t | x_t) f_{\theta,t+1}(x_{t+1}^{(b_{t+1})} | x_t) dx} \end{aligned} \quad (5)$$

As before, this is a collapsed Gibbs move.

In standard backward simulation, the conditional for each ancestor index is a categorical distribution (4), which can be sampled directly by evaluating the weight associated with each possible value, or by rejection sampling [15, 16]. It is also possible to use an MH kernel targeting this distribution [7].

In contrast, the joint conditional for state-ancestor pairs is a mixed continuous-discrete distribution (5). Since it will not in general be possible to sample from this distribution directly, we consider two possible Markov kernels which can be used instead. To clarify the following explanations, we write the one-step target distribution in a simplified form, omitting superfluous indexes and conditioning,

$$\begin{aligned} \pi(a_t, x_t | \mathbf{x}_{t-1}) &= \frac{w_{t-1}^{(a_t)} \rho_t(x_t | x_{t-1}^{(a_t)})}{\sum_j w_{t-1}^{(j)} \int \rho_t(x | x_{t-1}^{(j)}) dx} \\ \rho_t(x_t | x_{t-1}^{(a_t)}) &= f_{\theta,t}(x_t | x_{t-1}^{(a_t)}) g_{\theta,t}(y_t | x_t) f_{\theta,t+1}(x_{t+1}^{(b_{t+1})} | x_t). \end{aligned} \quad (6)$$

3.2.1. Metropolis-Hastings

We can target (6) using MH. From current values a_t^* and x_t^* , we can propose new values a_t' and x_t' by drawing from,

$$\frac{v_{t-1}^{(a_t')}}{\sum_j v_{t-1}^{(j)}} \phi_t(x_t' | x_{t-1}^{(a_t')}, x_t^*), \quad (7)$$

in which $\{v_{t-1}^{(i)} : i = 1, \dots, N\}$ are a set of proposal weights for the ancestor index and ϕ_t is a new proposal density. The resulting

acceptance probability is then,

$$\begin{aligned} \alpha(\{a_t^*, x_t^*\} \rightarrow \{a_t', x_t'\}) \\ = \min \left\{ 1, \frac{w_{t-1}^{(a_t')} \rho_t(x_t' | x_{t-1}^{(a_t')})}{w_{t-1}^{(a_t^*)} \rho_t(x_t^* | x_{t-1}^{(a_t^*)})} \frac{v_{t-1}^{(a_t^*)} \phi_t(x_t^* | x_{t-1}^{(a_t^*)}, x_t')}{v_{t-1}^{(a_t')} \phi_t(x_t' | x_{t-1}^{(a_t')}, x_t^*)} \right\} \end{aligned} \quad (8)$$

This is the scheme suggested in [7] for state smoothing.

3.2.2. Conditional Importance Sampling

The marginal conditional distribution for the ancestor indexes is,

$$\pi(a_t | \mathbf{x}_{t-1}) = \int \pi(x, a_t | \mathbf{x}_{t-1}) dx = \frac{w_{t-1}^{(a_t)} \int \rho_t(x | x_{t-1}^{(a_t)}) dx}{\sum_j w_{t-1}^{(j)} \int \rho_t(x | x_{t-1}^{(j)}) dx}.$$

If this distribution is dominated by a small number of ancestors with high probability, then an MH kernel will be inefficient. It may take a large number of steps before one of these is proposed. In such circumstances it may be advantageous to use the following method based on *conditional importance sampling* (CIS) instead.

CIS uses the same principle as the conditional particle filter [1], but applied to a single time step. Suppose we have the current values a_t^* and x_t^* , then a Markov kernel may be constructed with (6) as its invariant distribution by following algorithm 1, in which $\psi_t(x_t | x_{t-1})$ is an appropriate importance density.

Algorithm 1 Conditional importance sampling for the joint ancestor-state conditional distributions.

Require: Preceding particle states \mathbf{x}_{t-1} , current values a_t^* and x_t^* .

- 1: Sample an index uniformly $c^* \in \{1, \dots, N\}$.
 - 2: Set $a_t^{(c^*)} = a_t^*$. Set $x_t^{(c^*)} = x_t^*$.
 - 3: **for all** $i \in \{1, \dots, N\} \setminus c^*$ **do**
 - 4: Sample $a_t^{(i)} \sim \frac{v_{t-1}^{(a_t^*)}}{\sum_j v_{t-1}^{(j)}}$. Sample $x_t^{(i)} \sim \psi_t(x_t | x_{t-1}^{(a_t^{(i)})})$.
 - 5: **end for**
 - 6: Sample $c' \sim \frac{u_t^{(c)}}{\sum_j u_t^{(j)}}$, where $u_t = \frac{w_{t-1}^{(a_t^{(i)})} \rho_t(x_t^{(i)} | x_{t-1}^{(a_t^{(i)})})}{v_{t-1}^{(a_t^{(i)})} \psi_t(x_t^{(i)} | x_{t-1}^{(a_t^{(i)})})}$.
 - 7: Set $a_t' = a_t^{(c')}$.
 - 8: Set $x_t' = x_t^{(c')}$.
 - 9: **return** New values a_t' and x_t' .
-

To justify that this is a correct Markov kernel, we construct another extended target distribution over these particles. (Note that this set of particles is separate to that of the primary Gibbs sampler.)

$$\eta(\mathbf{a}_t, \mathbf{x}_t, c) = \frac{1}{N} \pi(x_t^{(c)}, a_t^{(c)} | \mathbf{x}_{t-1}) \prod_{i \neq c} \frac{v_{t-1}^{(a_t^{(i)})}}{\sum_j v_{t-1}^{(j)}} \psi_t(x_t^{(i)} | x_{t-1}^{(a_t^{(i)})}).$$

The first part of algorithm 1 by construction corresponds to sampling from the conditional distribution, $\eta(\mathbf{a}_t^{(-c)}, \mathbf{x}_t^{(-c)} | a_t^{(c)}, x_t^{(c)}, c)$, and the final part to sampling from $\eta(c | \mathbf{a}_t, \mathbf{x}_t)$. Hence, if the starting values are distributed according to the desired posterior, then the final values must also be, and so the procedure is a Markov kernel with the desired invariant distribution.

4. EXTENSIONS AND VARIATIONS

4.1. Multiple Time Steps

In extreme cases, even with refreshed backward simulation the update rate of the earliest states may be low. If this occurs, it may be beneficial to extend the method to sample states at multiple steps, thus giving us yet more leeway to match the sampled future to the possible particle histories. At each time instant we now sample from,

$$\begin{aligned} & \pi(a_t^{(b_t)}, x_{t:t+L-1}^{(b_t)} | \theta, \mathbf{x}_{1:t-1}, \mathbf{a}_{2:t-1}, x_{t+L:T}^{(b_{t+L:T})}, a_{t+1:T}^{(b_{t+1:T})}, K) \\ & \propto w_{t-1}^{(a_t^{(b_t)})} f_{\theta, t+L}(x_{t+L}^{(b_{t+L})} | x_{t+L-1}^{(b_{t+L-1})}) \\ & \quad \times \prod_{k=t}^{t+L-1} f_{\theta, k}(x_k^{(b_k)} | x_{k-1}^{(a_k^{(b_k)})}) g_{\theta, k}(y_k | x_k^{(b_k)}). \end{aligned}$$

As before, this is a collapsed Gibbs move. Implementing such a scheme will require careful design of proposal or importance densities for the joining bridge of states $x_{t:t+L-1}^{(b_{t:t+L-1})}$.

4.2. Ancestor Sampling

Rather than conducting the complete forward sweep (i.e. the conditional particle filter) followed by a backward simulation sweep, the steps of the two may be interleaved. This is the basis of particle Gibbs with ancestor sampling (PG-AS) [4]. At time step t , we would first sample from,

$$\pi(a_t^{(-b_t)}, x_t^{(-b_t)} | \theta, \mathbf{a}_{2:t-1}, \mathbf{x}_{1:t-1}, a_{t:T}^{(b_{t:T})}, x_{t:T}^{(b_{t:T})}, K),$$

and then from,

$$\pi(a_t^{(b_t)}, x_t^{(b_t)} | \theta, \mathbf{a}_{2:t-1}, \mathbf{x}_{1:t-1}, a_{t+1:T}^{(b_{t+1:T})}, x_{t+1:T}^{(b_{t+1:T})}, K).$$

5. SIMULATIONS

5.1. The Model

Particle Gibbs (PG), Particle Gibbs with Backward Simulation (PG-BS) and Particle Gibbs with Refreshed Backward Simulation (PG-RBS) were tested on a tracking model. The transition model is 3D near constant velocity motion [17], and the observation model is noisy measurements of bearing, elevation and range. The parameter to be learned is the scale factor on the transition covariance matrix σ^2 , which characterises the target manoeuvrability. Data sets are simulated from the model, each with 100 time steps. An uninformative conjugate prior is used for σ^2 .

5.2. Algorithm Settings

The algorithms were each run on 5 different simulated data sets. Each algorithm was run for 5000 iterations, with a burn in of 1000. PG and PG-BS were run twice, with 100 and 200 particles each. PG-RBS was run with 100 particles. PG-RBS with 100 particles takes roughly the same time as PG-BS with 200 particles.

The particle filter uses an extended Kalman filter approximation to the optimal importance density,

$$q_t(x_t | x_{t-1}) \approx p(x_t | x_{t-1}, y_t).$$

PG-RBS is implemented with the CIS method using a similar Gaussian approximation for the importance density,

$$\psi_t(x_t | x_{t-1}) \approx p(x_t | x_{t-1}, x_{t+1}, y_t).$$

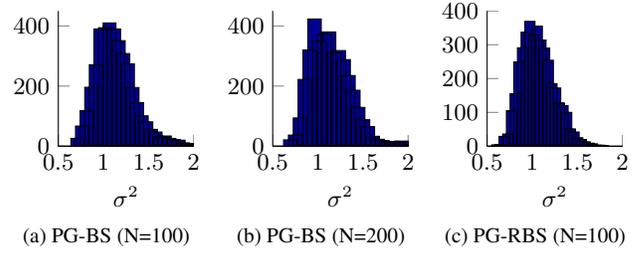


Fig. 2. Posterior sample histograms.

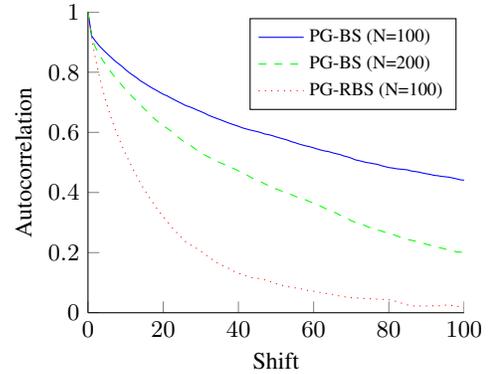


Fig. 3. Mean autocorrelation function for PG-BS and PG-RBS.

For the proposal weights, we use $v_t^{(i)} = w_t^{(i)}$.

5.3. Results

PG does not mix well at all. Parameter estimates do not approach the true value, even after 5000 iterations. PG-BS and PG-RBS do converge. Figure 2 shows the posterior histograms and figure 3 the mean autocorrelation function over the 5 simulations. The latter indicates faster mixing from PG-RBS with both equal-time and equal-particle equivalents.

6. DISCUSSION AND CONCLUSIONS

We have introduced a simple but effective modification to particle Gibbs which can significantly improve the mixing of the Markov chains. This uses a sweep of refreshed backward simulation after each iteration of the conditional particle filter, in which new values of the states of the reference trajectory are sampled along with their corresponding ancestor indexes. Direct sampling of the appropriate conditional distributions is not generally possible, however suitable Markov kernels are available instead. As well as basic Metropolis-Hastings, we have presented a kernel based on conditional importance sampling. The benefit of using refreshed backward simulation will depend on the efficiency of this Markov kernel. Simulations on a simple tracking model have demonstrated substantial improvements in Markov chain autocorrelation.

7. REFERENCES

- [1] C Andrieu, A Doucet, and R Holenstein, “Particle Markov chain Monte Carlo methods,” *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 72, pp. 269–342, 2010.
- [2] J Olsson and T Rydén, “Rao-Blackwellization of particle Markov chain Monte Carlo methods using forward filtering backward sampling,” *IEEE Transactions on Signal Processing*, vol. 59, no. 10, pp. 4606–4619, 2011.
- [3] N Chopin and S Singh, “On particle Gibbs sampling,” *Bernoulli (forthcoming)*, 2014.
- [4] F Lindsten, M I Jordan, and T B Schön, “Particle Gibbs with ancestor sampling,” *Journal of Machine Learning Research*, vol. 15, pp. 2145–2184, 2014.
- [5] N Whiteley, “Discussion on “Particle Markov chain Monte Carlo methods” by Andrieu et al.,” *Journal of the Royal Statistical Society: Series B*, vol. 72, no. 3, pp. 306–307, 2010.
- [6] F Lindsten and T B Schön, “On the use of backward simulation in the particle Gibbs sampler,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012, pp. 3845–3848.
- [7] P Bunch and S Godsill, “Improved particle approximations to the joint smoothing distribution using Markov chain Monte Carlo,” *IEEE Transactions on Signal Processing*, vol. 61, no. 4, pp. 956–963, 2013.
- [8] P Bunch, *Particle Filtering and Smoothing for Challenging Time Series Models*, Ph.D. thesis, University of Cambridge, 2014.
- [9] C K Carter, E F Mendes, and R Kohn, “An extended space approach for particle Markov chain Monte Carlo methods,” *arXiv preprint arXiv:1406.5795*, 2014.
- [10] O. Cappé, S.J. Godsill, and E. Moulines, “An overview of existing methods and recent advances in sequential Monte Carlo,” *Proceedings of the IEEE*, vol. 95, pp. 899–924, 2007.
- [11] A Doucet and A M Johansen, “A tutorial on particle filtering and smoothing: Fifteen years later,” in *The Oxford Handbook of Nonlinear Filtering*, D Crisan and B Rozovsky, Eds. Oxford University Press, 2009.
- [12] P E. Jacob, L M. Murray, and S Rubenthaler, “Path storage in the particle filter,” *Statistics and Computing*, pp. 1–10, 2013.
- [13] D A van Dyk and T Park, “Partially collapsed Gibbs samplers,” *Journal of the American Statistical Association*, vol. 103, no. 482, pp. 790–796, 2008.
- [14] S J Godsill, A Doucet, and M West, “Monte Carlo smoothing for nonlinear time series,” *Journal of the American Statistical Association*, vol. 99, no. 465, pp. 156–168, 2004.
- [15] R Douc, A Garivier, E Moulines, and J Olsson, “Sequential Monte Carlo smoothing for general state space hidden Markov models,” *The Annals of Applied Probability*, vol. 21, no. 6, pp. 2109–2145, 2011.
- [16] E Taghavi, F Lindsten, L Svensson, and T B Schön, “Adaptive stopping for fast particle smoothing,” in *38th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013.
- [17] X R Li and V P Jilkov, “Survey of maneuvering target tracking. part I: Dynamic models,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1333–1364, 2003.