Original citation:
Johansen, Adam M. (2015) On block, tempering and particle MCMC for systems identification. In: 17th IFAC Symposium on System Identification, Beijing, China, 19-21 Oct 2015. Published in: Proceedings of 17th IFAC Symposium on System Identification

Permanent WRAP url:
http://wrap.warwick.ac.uk/74067

On Blocks, Tempering and Particle MCMC for Systems Identification

Adam M. Johansen

Department of Statistics, University of Warwick, Coventry, CV4 7AL, UK (e-mail: a.m.johansen@warwick.ac.uk)

Abstract: The widespread use of particle methods for addressing the filtering and smoothing problems in state-space models has, in recent years, been complemented by the development of particle Markov Chain Monte Carlo (PMCMC) methods. PMCMC uses particle filters within offline systems-identification settings. We develop a modified particle filter, based around block sampling and tempering, intended to improve their exploration of the state space and the associated estimation of the marginal likelihood. The aim is to develop particle methods with improved robustness properties, particularly for parameter values which are *not* able to explain observed data well, for use within PMCMC algorithms. The proposed strategies do not require a substantial analytic understanding of the model structure, unlike most techniques for improving particle-filter performance.

Keywords: Dynamic Systems; Monte Carlo Method; Optimal Filtering; Target Tracking

1. INTRODUCTION

We consider the problem of identification of an indirectly-observed dynamical system in which the functional form of the dynamics, and of the measurement system, is known up to a collection of unknown parameters. In particular, let $\{(X_t)\}_{t\geq 1} \in \mathcal{X}^{\mathbb{N}}$ be an unobserved homogeneous Markov process characterized by its initial density

$$X_1 \sim \mu_\theta(\cdot) \tag{1}$$

and transition probability density

$$X_t|(X_{t-1} = x_{t-1}) \sim f_\theta(\cdot|x_{t-1}) \tag{2}$$

w.r.t to a dominating measure, e.g. Lebesgue, denoted abusively dx_t. The observations $\{Y_t\}_{t\geq 1} \in \mathcal{Y}^{\mathbb{N}}$ are assumed to be conditionally independent given $\{(X_t)\}_{t\geq 1}$, with common conditional probability density

$$Y_t|\left(X_t = x_t\right) \sim g_\theta\left(\cdot|\, x_t\right) \tag{3}$$

w.r.t. to a dominating measure denoted dy_t.

In settings in which the parameter vector, θ, is known *a priori* the remaining inferential problem is one of filtering and smoothing: deducing the conditional distribution of the latent state sequences given the observations received. Such inference typically proceeds via the conditional distribution of the latent sequence given the parameter value and the observed data: $p^\theta(x_{1:t}|y_{1:t})$. These and related inferential tasks are computationally challenging, and except for a small number of systems with convenient analytical structure, it is necessary to resort to numerical methods. Particle filters are iterative Monte Carlo algorithms which combine importance sampling and resampling techniques, propagating a weighted sample from one iteration to the next so as to provide an empirical approximation of each distribution in turn. Such methods date back to Gordon et al. (1993); see Doucet and Johansen (2011) for an overview of the numerous developments since then.

Within this parametric framework, the problem of *systems identification* is one of parameter inference: within the specified model class, which parameter vector θ provides the best description of the system, given the observations available thus far? A recent overview of methods available to address this problem is provided by Kantas et al. (2015). Here we focus upon the particular approach of Particle Markov chain Monte Carlo (PMCMC), developed by Andrieu et al. (2010)

PMCMC employs particle methods within MCMC to update the latent state sequence (either using Metropolis-Hastings or Gibbs-Sampler type updates). For this strategy to work reliably, it is necessary for the particle filter to behave well — providing good estimates of the marginal likelihood in particular — for all values of the parameter vector within the support of the posterior distribution. The algorithm proposed below aims to provide a particle filter robust enough to perform well in a wide variety of situations with minimal application-specific tuning.

Particle methods can “lose track” of the current state when applied to data sequences which are unlikely under the assumed model. Such a scenario can arise when the model imperfectly describes the real world system under study (perhaps a step change in the latent state occurs in reality, but cannot be described by the model, or perhaps an outlying observation which would be extremely improbable under the model is obtained), and even if the model is perfect, if the wrong parameter value is assumed such a situation can arise — if, e.g., the variance of the observation noise is dramatically underestimated.

Although the stability properties of the filter can be such that a particle filter can recover from this problem (see, e.g., Del Moral (2004) for a thorough exposition of these forgetting properties and their use in controlling the approximation error of particle filters), this phenomenon can have a substantial influence on the approximation of the smoothing distribution and, indeed, upon the approximation of the marginal likelihood.

A natural strategy is to incorporate information from subsequent observations into the proposal distribution. The first step towards this strategy is the use of a good approximation of the locally-optimal proposal distribution $p(x_t|y_t, x_{t-1})$ within the particle filter (Doucet et al., 2000), followed by the adoption of the auxiliary particle filter approach (Pitt and Shephard, 1999). These methods can improve the behaviour of particle filters, sometimes substantially, but don't allow information from observations beyond time t to influence the proposal at time t. In settings in which some slight lag between observation and estimation is permissible, it is possible to make use of *block-sampling* strategies (Doucet et al., 2006) or the closely-related *lookahead* methods (Lin et al., 2013).

The use of *tempering* to allow difficult inference problems in which the prior and posterior are substantially different within sequential Monte Carlo is common (Neal, 2001; Del Moral et al., 2006b). Its use in particle filtering settings, however, has to the best of our knowledge been restricted to gradually introducing individual observations (Godsill and Clapp, 2001) — and the approach advocated by (Liu, 2001, p73–74) in which resampling is done using a transformation of the importance weights with residual weights retained after resampling is of a similar character.

The approach developed here employs tempering strategies to emulate the block-sampling particle filter *without* recourse to carefully designed proposal distributions.

2. METHODOLOGY

2.1 Block-Tempered Particle Filters

We seek to develop particle filters which are robust to parameter misspecification: we wish to be able to apply the algorithm to obtain good approximations of quantities associated with the model for all parameter values, regardless of whether these values are well supported by the data. In addition to the obvious desirability of such robustness, it is necessary in the PMCMC context to obtain such estimates. Our motivation is to develop particle "filters" suitable for incorporation within a PMCMC algorithm and as such our main objective is to provide the best possible approximation of the normalising constant (and hence marginal likelihood; noting that $p(y|\theta) = \int p(y|x,\theta)p(x|\theta)dx$) for given parameter values for a data sequence over some fixed (discrete) time interval $[\![1,T]\!] := \{1,\ldots,T\}$.

In order to provide greater robustness — without requiring the construction of near-optimal proposal distributions, perhaps in a block-sampling context, which is realistically possible in only a small proportion of HMMs — we consider a generic strategy which allows information about the model which is available to be employed but requires only that we can evaluate the transition density and likelihood pointwise up to a normalizing constant.

The general idea is to use the two degrees of freedom provided by the sequential importance resampling framework within which particle methods exist: to produce a sequence of distributions which is more regular and varies more slowly than the natural sequence provided by the filtering distributions, $\{p^\theta(x_{1:t}|y_{1:t})\}_{t=1}^T$, and to use proposal distributions which allow us to properly explore the support of each of these distributions in turn. Motivated by the good theoretical properties of block-sampling (Doucet et al., 2006), we propose to use tempered MCMC transitions in order to update blocks of variables gradually. In contrast to applying these MCMC transitions within a simple particle filter, the proposed strategy introduces the influence of each observation gradually, allowing the filter to adapt to improbable observations *before* fully incorporating their contribution to the likelihood within the importance weights. In doing this, we aim to ensure that the successive distributions from which the algorithm samples are always *close enough* to one another (in χ^2-distance, say) to allow for good approximation of their relative normalising constants.

For definiteness, we consider a collection of distributions with densities of the form (with $t \wedge T := \min(t,T)$):

$$\pi^\theta_{t,r}(x_{1:t\wedge T}) = \mu_\theta(x_1)^{\beta^1_{(t,r)}} g_\theta(y_1|x_1)^{\gamma^1_{(t,r)}} \cdot \prod_{s=2}^{T\wedge t} f_\theta(x_s|x_{s-1})^{\beta^s_{(t,r)}} g_\theta(y_s|x_s)^{\gamma^s_{(t,r)}}, \quad (4)$$

where $\{\beta^s_{(t,r)}\}$ and $\{\gamma^s_{(t,r)}\}$ are a collection of $[0,1]$-valued variables defined for all $s \in [\![1,T]\!]$, $r \in [\![1,R]\!]$ and $t \in [\![1,T']\!]$, with $T' = T + L$ for some $R, L \in \mathbb{N}$. These are essentially sequences of tempering parameters. The algorithm proceeds for T' iterations. During iteration t it iterates through R steps to approximate the distribution $\pi^\theta_{t,r}$ over $x_{1:t\wedge T}$. $\gamma^s_{(t,r)}$ determines the influence that the time s observation has during iteration t, step r of the algorithm and $\beta^s_{(t,r)}$ plays a similar rôle for the density of the transition from time $s-1$ to s during that algorithmic iteration. $\beta^s_{(t,r)}$ and $\gamma^s_{(t,r)}$ are each non-decreasing, both in r for every valid s,t and in t for every valid r,s. Typically, as one wishes to gradually introduce the influence of observations in the order in which they arrive, one would expect the collections to be non-increasing in s and subject to the constraints that both collections take the value 1 for all r when $s < t - L$ and the value 0 for any r when $t < s$. Note that L and R are each design parameters, with the lag parameter, L, specifying the size of blocks over which the influence of each observation is introduced and the second parameter, R, the number of intermediate steps of the algorithm performed for each observation.

Some simple possibilities for the tempering strategies include tempering both transition and observation densities smoothly over a window of length L, setting

$$\beta^s_{(t,r)} = \gamma^s_{(t,r)} = \left(1 \wedge \frac{R(t-s)+r}{RL}\right) \vee 0 \quad (5)$$

or tempering only the observation density over such a window, using

$$\beta^s_{(t,r)} = \mathbb{I}\{s \le t\} \quad \gamma^s_{(t,r)} = \left(1 \wedge \frac{R(t-s)+r}{RL}\right) \vee 0, \quad (6)$$

where $\mathbb{I}\{s \le t\}$ takes the value 1 if $\{s \le t\}$ and 0 otherwise. Note that setting $\beta^s_{(t,r)} = \mathbb{I}\{s \le t\}$ and $\gamma^s_{(t,r)} = \mathbb{I}\{s < t\} + r\mathbb{I}\{s = t\}/R$ recovers the simple likelihood-tempered particle filter, with the influence of observation t introduced over R MCMC steps during iteration t.

It should be noted that many variations of this basic structure are possible and might find application in particular problems. A more general treatment leads to substantial increases in notational complexity, but requires essentially the same argument as that presented here and so we restrict ourselves to this case in the interests of parsimony.

Algorithm 1 provides a statement of a simple generic algorithm which employs tempering within SMC, a *block-tempered particle filter (BTPF)*. It can be viewed as a sequence of importance reweighting steps followed by the application of MCMC kernels which preserve the distribution targeted at each step of the algorithm in the spirt of *generalised importance sampling* (Robert and Casella, 2004), but a different interpretation will be provided below. Although our motivation is controlling the fluctuations of the normalising constant estimate with PMCMC in mind, it would be possible to use this algorithm as an alternative particle filter in many settings.

Algorithm 1 Block-Tempered Particle Filter

```
Initialization: Set $t \leftarrow 1; r \leftarrow 1$
  (1) Sample $X^i_{(1,1)} \sim q_1$
  (2) Weight $W^i_{(1,1)} \propto \pi_{(1,1)}(X^i_{(1,1)})/q_1(X^i_{(1,1)})$
  (3) For $r = 2 : R$:
      (a) Resample from $\{W^i_{(1,r-1)}, X^i_{(1,r-1)}\}_{i=1}^N$ to obtain $\{1/N, \widetilde{X}^i_{(1,r)}\}_{i=1}^N$
      (b) Weight $W^i_{(1,r)} \propto \pi_{(1,r)}(\widetilde{X}^i_{(1,r)})/\pi_{(1,r-1)}(\widetilde{X}^i_{(1,r)})$
      (c) Sample $X^i_{(1,r)} \sim K_{(1,r)}(\widetilde{X}^i_{(1,r)}, \cdot)$
  (4) Set $\mathbf{X}^i_{(1,R)} = X^i_{(1,R)}$.
Iteration: Set $t \leftarrow t + 1; r \leftarrow 1$
  (1) Resample from $\{W^i_{(t-1,R)}, \mathbf{X}^i_{(t-1,R)}\}_{i=1}^N$to obtain $\{1/N, \widetilde{\mathbf{X}}^i_{(t,1)}\}_{i=1}^N$
  (2) If $t \leq T$:
      (a) Sample $X^i_{(t,1),t} \sim q_t(\cdot; \widetilde{\mathbf{X}}^i_{(t,1)})$
      (b) Set $\mathbf{X}^i_{(t,1)} = \left(\widetilde{\mathbf{X}}^i_{(t,1)}, X^i_{(t,1),t}\right)$
  (3) Weight
      $$W^i_{(t,1)} \propto \frac{\pi_{(t,1)}(\mathbf{X}^i_{(t,r)})}{\pi_{(t-1,R)}(\widetilde{\mathbf{X}}^i_{(t,r)})q_t(X^i_{(t,1)}; \widetilde{\mathbf{X}}^i_{(t,1)})}$$
  (4) For $r = 2 : R$:
      (a) Resample from $\{W^i_{(t,r-1)}, \mathbf{X}^i_{(t,r-1)}\}_{i=1}^N$ to obtain $\{1/N, \widetilde{\mathbf{X}}^i_{(t,r)}\}_{i=1}^N$
      (b) Weight $W^i_{(t,r)} \propto \pi_{(t,r)}(\widetilde{\mathbf{X}}^i_{(t,r)})/\pi_{(t,r-1)}(\widetilde{\mathbf{X}}^i_{(t,r)})$
      (c) Sample $\mathbf{X}^i_{(t,r)} \sim K_{(t,r)}(\widetilde{\mathbf{X}}^i_{(t,r)}, \cdot)$
Notes:
 (1) The index $i$ is a dummy index identifying a particular particle; each line must be carried out for every $i \in [\![1, N]\!]$.
 (2) $q_t$ are importance sampling proposal distributions.
 (3) $K_{(t,r)}$ are $\pi_{(t,r)}$-invariant Markov kernels (which need not be ergodic and which are typically at least partially degenerate, updating only states $x_{t-L:t}$).
```

As the distributions targeted by this class of SMC algorithms do not (in general) coincide with the sequence of filtering distributions, a little extra work is required in order to obtain online filtering estimates. At the end of an iteration of the algorithm (i.e. following step 4 of Algorithm 1), $\{W^i_{(t,R)}, \mathbf{X}^i_{(t,R)}\}_{i=1}^N$ is a sample weighted to target $\pi_{(t,R)}$ and it is reasonable to use the associated weighted empirical measure to approximate this distribution. The filtering distribution itself can be approximated with the reweighted empirical measure:

$$\widehat{p}^N(dx_{1:t\wedge T}|y_{1:t\wedge T}) = \sum_{i=1}^N \widetilde{W}_t \delta_{\mathbf{X}^i_{(t,R)}}(dx_{1:t\wedge T})$$

where the weightings are given by

$$\widetilde{W}^i_t \propto W^i_{(t,R)} \frac{p(\mathbf{X}^i_{(t,R)}|y_{1:t})}{\pi_{(t,R)}(\mathbf{X}^i_{(t,R)})}, \qquad \sum_{i=1}^N \widetilde{W}^i_t = 1.$$

Note that these importance weights are readily calculated when using the sequence of distributions described by Equation 4, and in particular:

$$\frac{p(x_{1:t\wedge T}|y_{1:t})}{\pi_{(t,R)}(x_{1:t\wedge T})} \propto \mu_\theta(x_1)^{1-\beta^1_{(t,R)}} g_\theta(y_1|x_1)^{1-\gamma^1_{(t,R)}} \cdot \prod_{s=2}^{T\wedge t} f_\theta(x_s|x_{s-1})^{1-\beta^s_{(t,R)}} g_\theta(y_s|x_s)^{1-\gamma^s_{(t,R)}},$$

where we'd ordinarily expect many of the $\beta^s_{(t,R)}$ and $\gamma^s_{(t,R)}$ values to be identically 1, perhaps for all $s < t - L$ as discussed above, leading to further simplification. Expectations with respect to the filtering/smoothing distributions can then be approximated by the expectation with respect to this weighted random measure.

We can justify Algorithm 1 a little more formally as an instance of an SMC sampler in the sense of Del Moral et al. (2006b). Such samplers provide, iteratively, collections of weighted samples from a sequence of distributions $\{\pi_n\}_{n=1}^{n_{\max}}$ over essentially any random variables on some measurable spaces $(E_n, \mathcal{E}_n)$, by constructing a sequence of auxiliary distributions $\{\widetilde{\pi}_n\}_{n=1}^{n_{\max}}$ on spaces of increasing dimensions,

$$\widetilde{\pi}_n(dx_{1:n}) = \pi_n(dx_n) \prod_{s=1}^{n-1} L_s(x_{s+1}, dx_s), \qquad (7)$$

where the sequence of Markov kernels $\{L_s\}_{s=1}^{n-1}$, termed auxiliary kernels, is formally arbitrary but critically influences the estimator variance. Standard sequential importance resampling algorithms can then be applied to the sequence of synthetic distributions, $\{\widetilde{\pi}_n\}_{n=1}^{n_{\max}}$.

For given values of R and T', we define the sequence of extended target distributions to be:

$$\widetilde{\pi}^\theta_n(d\mathbf{x}_{1:n}) := \pi^\theta_{(t_n, r_n)}(d\mathbf{x}_n) \prod_{s=1}^{n-1} L_s(\mathbf{x}_{s+1}, d\mathbf{x_s}),$$

where the sequences r_n and t_n are defined as:

$$t_n = \left\lfloor \frac{n-1}{R} \right\rfloor + 1$$
$$r_n = (n - 1 \mod R) + 1,$$

we identify $\mathbf{x}_n$ in this sequence with $\mathbf{x}_{(t_n, r_n)}$ in the multi-index formulation, the forward kernels are the same proposal distribution / MCMC kernels as specified in Algorithm 1:

$$K_n(\mathbf{x}, d\mathbf{x}') = \begin{cases} \delta_{\mathbf{x}}(d\mathbf{x}'_{1:t_{n-1}}) q_{t_n}(d\mathbf{x}'{}_{t_n}; \mathbf{x}) & \begin{array}{l} r_n = 1, \\ t_n \leq T \end{array} \\ \\ K_{(t_n, r_n)}(\mathbf{x}, d\mathbf{x}') & \text{otherwise} \end{cases}$$

and the auxiliary (or backward) kernels are:

$$L_n(\mathbf{x}, d\mathbf{x}') = \begin{cases} \delta_{\mathbf{x}_{1:t_n-1}}(d\mathbf{x}') & \begin{array}{l} r_n = 1, \\ t_n \leq T \end{array} \\ K_n^{TR}(\mathbf{x}, d\mathbf{x}') & \text{otherwise} \end{cases},$$

where K_n^{TR} denotes the time-reversal of K_n with respect to its own invariant distribution, $\pi_{(t_n, r_n)}$. Note that this coincides with carrying out a standard (state-extending) sequential importance resampling step whenever a new observation is introduced and employing the time-reversal kernel associated with the MCMC kernel used in the forward direction at other steps. See Del Moral et al. (2006b) for further details.

Remark 1. It should be understood that much of the apparent degeneracy induced by this representation can be eliminated at the expense of slightly more cumbersome notation (and doing so can be essential if one wishes to employ backward (Whiteley, 2010) or ancestor (Lindsten et al., 2012) sampling within a particle Gibbs framework; see Finke et al. (2014) for an example in a slightly more complicated scenario). However, in the current setting the main barrier to overcome before such an approach could be employed is the partial degeneracy resulting from the MCMC transitions: Metropolis-Hastings type moves will inherently have positive probability of rejection for at least some state values and consequently introduce non-trivial dependence between the value taken by a particle during iteration t and $t + 1$. In this context other strategies to improve behaviour may be more readily applicable.

2.2 Block-Tempered Particle Filters for PMCMC

Although it is well known that the PMCMC approach is not limited to parameter estimation for HMMs and that quite general SMC algorithms can be embedded within MCMC algorithms using this framework, the vast majority of the PMCMC literature, particularly that for parameter estimation in HMMs, employs relatively straightforward particle filters. A notable exception is the case in which the likelihood can take the value zero in which developments of the Alive particle filter have proved fruitful (Jasra et al., 2013; Persing and Jasra, 2014). There has also been some work using other SMC algorithms in different contexts (e.g. Finke et al. (2014); Jasra et al. (2014)), and some effort has gone into improving the MCMC component of PMCMC (e.g. Dahlin et al. (2014)) but when dealing with inference for HMMs standard particle filters are generally employed. A recursive variant of PMCMC in which the MCMC algorithm is replaced with a corresponding SMC algorithm was also introduced by Chopin et al. (2013).

It has been reported that vanilla particle filtering algorithms can experience difficulties when some *a priori* plausible parameter values are inconsistent with the observed data, e.g. Owen and Gillespie (2015, Section 4.2.1). However, although it is known that more sophisticated particle filtering strategies can be incorporated within the PMCMC framework by representing them as simple SIR algorithms e.g., Andrieu et al. (2010, Section 4) who suggest employing the representations of Doucet and Johansen (2011), little use appears to have been made of these strategies in practice. The simplest such extension is to the auxiliary particle filter Pitt and Shephard (1999), which can be viewed as a standard particle filter which targets a sequence of auxiliary distributions (Johansen and Doucet, 2008), as exploited in a PMCMC context by Dahlin et al. (2014) allowing an additional degree of design freedom.

The use of the more general formulation given in (Andrieu et al., 2010, Section 4), rather than the form specialized to the natural HMM implementation earlier in that paper, demonstrates that essentially any SIR algorithm can be used within a PMCMC context. The SMC sampler formulation of the algorithm described in the previous section demonstrates that it is a SIR algorithm with final target distribution $\widetilde{\pi}^{\theta}_{R(T+L)}$. Thus we can employ Algorithm 1 within a PMCMC algorithm, with parameter prior $p(\theta)$, knowing that the invariant distribution admits a marginal proportional to $p(\theta)\widetilde{\pi}^{\theta}_{R(T+L)}(\mathbf{x}_{1:R(T+L)})$, which by construction can be further marginalized to obtain a distribution proportional to $p(\theta)\pi^{\theta}_{(T+L,R)}(\mathbf{x}_{R(T+L)})$, i.e. the Bayesian posterior distribution of interest.

2.3 Extensions and Outstanding Questions

Numerous extensions of the technique described above are clearly possible. Indeed, most of the improvements to standard particle filters can also be employed within the setting described, although some care is required to ensure the validity of the resulting PMCMC algorithm.

We have focussed on particle variants of Metropolis-Hastings type algorithms. It would also be of interest to investigate the potential of these ideas in the Gibbs sampling context noting the need to develop efficient conditional SMC algorithms in this context.

One obvious question is how best to tune L and R. In a block-sampling setting the optimal lag is closely related to the timescale on which the system forgets and one can use that characteristic timescale to specify L. R is then set to the smallest number which allows adequate mixing based upon a pilot study. However, in the present setting these mixing properties will vary with the values of the parameter, θ. In the simplest case, in which L and R are state-independent constants, this leads to a compromise between values large enough to lead to adequate performance in difficult regions of the parameter space but not so large that gains in this region are overwhelmed by increased computational costs elsewhere. It would be still more interesting to develop "adaptive" versions of the algorithm, in the spirit of Jasra et al. (2014), in which the degree of tempering adapts to the level required to obtain acceptable performance.

Similarly, the choice and specification of annealing schedule in this algorithm provides a considerable degree of flexibility, but developing good generic (or, better, adaptive) approaches to this specification will be essential if such a strategy is to be widely applicable.

3. NUMERICAL ILLUSTRATION

Space does not permit an extensive numerical study and we content ourselves with demonstrating certain key features of the proposed algorithm. We do not consider its incorporation into PMCMC but instead demonstrate the quality of the marginal likelihood estimates provided by the core particle filtering algorithm. We illustrate the method on a simple linear Gaussian model for which it is possible to compute analytically the filtering and smoothing distributions, together with the marginal likelihood of any given parameter vector for an observed data sequence, allowing an absolute assessment of performance.

We consider the case in which $\theta = (\sigma_f, \sigma_g)$, setting

$$f_\theta(x'|x) = \mathcal{N}(x'; x, \sigma_f^2); \qquad g_\theta(y|x) = \mathcal{N}(y; x, \sigma_g^2). \qquad (8)$$

To compare the behaviour of bootstrap and block-tempered particle filters when there is a substantial mismatch between the model and the observed data (encapsulating the key feature of both data with outlying observations, and the case of model misspecification, including the use of parameters inconsistent with the data), we consider the data sequence shown in Figure 1a. Values of $\sigma_f = \sigma_g = 1$ were used for all computations. Exact filtering and smoothing distributions were obtained via Kalman-filtering and smoothing techniques and are shown in Figure 1b — illustrating the clear mismatch between the two at time 74 in particular.

Algorithm 1 was implemented using variable-at-a-time random walk Metropolis with unit proposal variance, scanning back L steps from the final state value currently present (i.e. running from time $T \wedge t$ to $T \wedge t - L + 1$). Results are summarized in Figure 2 which shows in panel (a) the empirical mean and standard deviation of estimates obtained from 50 independent replicates of various combinations of N, R and L and in panel (b) the corresponding estimated relative (normalized by dividing by the true value) root mean squared error (RMSE) of these estimates.

A value of $R = 0$ indicates that no MCMC moves were used (and corresponds to the standard bootstrap particle filter when $L = 1$). The ordinal axis shows a proxy for computational cost, $N(1 + RL)$. Throughout the range of costs considered, the bootstrap particle filter and resample move variant ($L = 1$, $R = 1$) are dominated by a particle filter with tempered transitions ($L = 1$, $R > 1$) and, *a fortiori*, the BTPF ($L > 1$) in this setting: the only *red* ($L = 1$) points in Figure 2b not to have at least one *blue* ($L = 3$) or *purple* ($L = 5$) point below it and to their left have relative errors greater than 2.5.

4. CONCLUSIONS

The BTPF is an approach to particle filtering that aims to be robust to model misspecification. It intended for use within PMCMC algorithms and might also find applications in settings in which certain observations are highly informative and carry significant information about the state sequence several time steps in advance. It is our intention to stress that algorithms more sophisticated than simple bootstrap particle filters are applicable in a PMCMC context and that, when poor mixing is observed and it is not feasible to increase the number of particles sufficiently to address it, more sophisticated filtering strategies such as the BTPF should be considered.

Numerous questions about, and possible extensions to, the BTPF emerge, as outlined in Section 2.3, suggesting several avenues for further work.

ACKNOWLEDGEMENTS

We thank Axel Finke and three anonymous referees for helpful comments on earlier versions of this manuscript.

REFERENCES

Andrieu, C., Doucet, A., and Holenstein, R. (2010). Particle Markov chain Monte Carlo. *Journal of the Royal Statistical Society B*, 72(3), 269–342.

Chopin, N., Jacob, P., and Papaspiliopoulos, O. (2013). SMC^2: an efficient algorithm for sequential analysis of state space models. *Journal of the Royal Statistical Society B*, 75(3), 397–426.

Dahlin, J., Lindsten, F., and Schön, T. (2014). Second-order particle MCMC for Bayesian parameter inference. In *Proceedings of the 18th World Congress of the International Federation of Automatic Control (IFAC)*. Cape Town, South Africa.

Del Moral, P. (2004). *Feynman-Kac formulae: genealogical and interacting particle systems with applications.* Springer, New York.

Del Moral, P., Doucet, A., and Jasra, A. (2006). Sequential Monte Carlo samplers. *Journal of the Royal Statistical Society B*, 63(3), 411–436.

Doucet, A., Briers, M., and Sénécal, S. (2006). Efficient block sampling strategies for sequential Monte Carlo methods. *Journal of Computational and Graphical Statistics*, 15(3), 693–711.

Doucet, A., Godsill, S., and Andrieu, C. (2000). On sequential simulation-based methods for Bayesian filtering. *Statistics and Computing*, 10(3), 197–208.

Doucet, A. and Johansen, A.M. (2011). A tutorial on particle filtering and smoothing: Fiteen years later. In D. Crisan and B. Rozovsky (eds.), *The Oxford Handbook of Nonlinear Filtering*, 656–704. Oxford University Press.

Finke, A., Johansen, A.M., and Spanò, D. (2014). Static-parameter estimation in piecewise deterministic processes using particle Gibbs samplers. *Annals of the Institute of Statistical Mathematics (Tokyo)*, 66(3), 577–609.

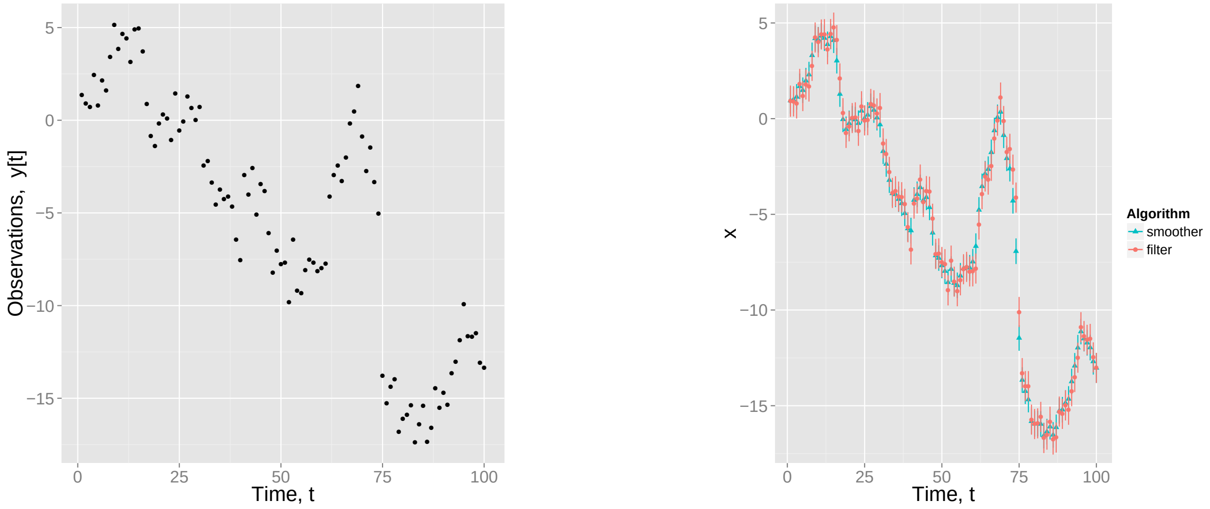
Godsill, S. and Clapp, T. (2001). Improvement strategies for Monte Carlo particle filters. In A. Doucet, N. de Freitas, and N. Gordon (eds.), *Sequential Monte Carlo Methods in Practice*, 139–158. Springer, New York.

Gordon, N.J., Salmond, S.J., and Smith, A.F.M. (1993). Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings-F*, 140(2), 107–113.

Jasra, A., Kantas, N., and Persing, A. (2014). Bayesian parameter inference for partially observed stopped processes. *Statistics and Computing*, 24(1), 1–20.

Jasra, A., Lee, A., Yau, C., and Zhang, X. (2013). The alive particle filter. Mathematics ePrint 1304.0151, ArXiv.

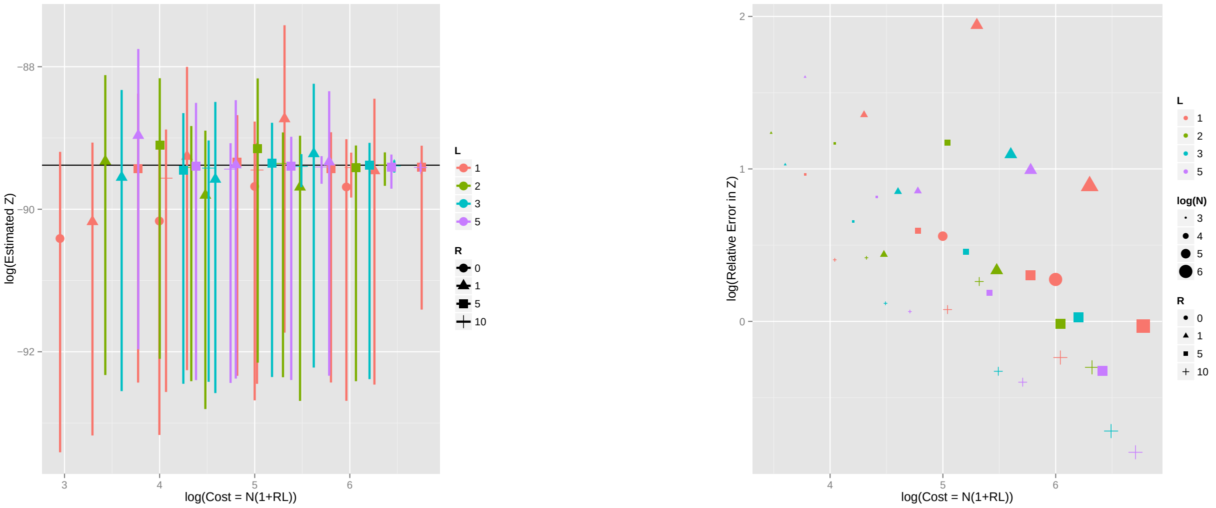
Johansen, A.M. and Doucet, A. (2008). A note on the auxiliary particle filter. *Statistics and Probability Letters*, 78(12), 1498–1504.



(a) Data sequence: generated from the model described in (8) with $\sigma_f = \sigma_g = 1$ and perturbed by subtracting 10 from all observations received at or after $t = 75$.

(b) Exact filter and smoother: points show the mean, vertical bars the extent of ±1 standard deviation intervals.

Fig. 1. Data sequence and exact filtering and smoothing distributions.



(a) Variability of marginal likelihood estimates (empirical mean ±1 standard deviation) obtained from 500 runs of various particle filtering algorithms shown against computational cost. The horizontal line shows the true value.

(b) Empirical relative RMSE of the estimated marginal likelihood obtained from 500 runs with each configuration (algorithms for which the mean estimate of Z was less than one fifth of the true value are not plotted). All logarithms are base 10.

Fig. 2. Illustration of the performance of marginal likelihood estimation based on a variety of particle filters.

Kantas, N., Doucet, A., Singh, S.S., Maciejowski, J.M., and Chopin, N. (2015). On particle methods for parameter estimation in general state-space models. *Statistical Science*. In press.

Lin, M., Chen, R., and Liu, J.S. (2013). Lookahead strategies for sequential Monte Carlo. *Statistical Science*, 28(1), 69–94.

Lindsten, F., Jordan, M.I., and Schön, T.B. (2012). Ancestor sampling for particle Gibbs. In *Proceedings of the 2012 Conference on Neural Information Processing Systems (NIPS)*. Lake Tahoe, NV, USA.

Liu, J.S. (2001). *Monte Carlo Strategies in Scientific Computing*. Springer, New York.

Neal, R.M. (2001). Annealed importance sampling. *Statistics and Computing*, 11, 125–139.

Owen, J.and Wilkinson, D.J. and Gillespie, C.S. (2015). Likelihood free inference for markov processes: a comparison. *Statistical Applications in Genetics and Molecular Biology*. In press.

Persing, A. and Jasra, A. (2014). Twisting the alive particle filter. *Methodology and Computing in Applied Probability*. In press. DOI 10.1007/s11009-014-9422-7.

Pitt, M.K. and Shephard, N. (1999). Filtering via simulation: Auxiliary particle filters. *Journal of the American Statistical Association*, 94(446), 590–599.

Robert, C.P. and Casella, G. (2004). *Monte Carlo Statistical Methods*. Springer, New York, second edition.

Whiteley, N. (2010). Contribution to the discussion of Andrieu, C., Doucet, A., and Holenstein, R (2010).