Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

Second-order particle MCMC for Bayesian parameter inference ⋆

Johan Dahlin * **Fredrik Lindsten** ** **Thomas B. Schön** ***

* *Division of Automatic Control, Linköping University, Sweden.*
E-mail: johan.dahlin@liu.se
** *Dept. of Engineering, University of Cambridge, United Kingdom.*
E-mail: fredrik.lindsten@eng.cam.ac.uk
*** *Division of Systems and Control, Uppsala University, Sweden.*
E-mail: thomas.schon@it.uu.se

Abstract: We propose an improved proposal distribution in the Particle Metropolis-Hastings (PMH) algorithm for Bayesian parameter inference in nonlinear state space models. This proposal incorporates second-order information about the parameter posterior distribution, which can be extracted from the particle filter already used within the PMH algorithm. The added information makes the proposal scale-invariant, simpler to tune and can possibly also shorten the burn-in phase. The proposed algorithm has a computational cost which is proportional to the number of particles, i.e. the same as the original marginal PMH algorithm. Finally, we provide two numerical examples that illustrates some of the possible benefits of adding the second-order information.

Keywords: Bayesian inference, Particle Markov Chain Monte Carlo, Sequential Monte Carlo.

1. INTRODUCTION

We are interested in Bayesian parameter inference in nonlinear state space models (SSM). An SSM with latent states $x_{0:T} \triangleq \{x_t\}_{t=0}^T$ and measurements $y_{1:T} \triangleq \{y_t\}_{t=1}^T$ is defined as

$$x_t|x_{t-1} \sim f_\theta(x_t|x_{t-1}), \tag{1a}$$
$$y_t|x_t \sim g_\theta(y_t|x_t), \tag{1b}$$

where $f_\theta(\cdot)$ and $g_\theta(\cdot)$ denote known distributions parametrised by the unknown static parameter vector $\theta \in \Theta \subseteq \mathbb{R}^d$. We also assume that the initial state is distributed according to $x_0 \sim \mu(x_0)$. In Bayesian inference, we are interested in computing the parameter posterior,

$$p(\theta|y_{1:T}) = \frac{p_\theta(y_{1:T})p(\theta)}{p(y_{1:T})}, \tag{2}$$

where $p(\theta)$ denotes the prior distribution of the parameter. Here, the likelihood function can be expressed as

$$p_\theta(y_{1:T}) = p(y_{1:T}|\theta) = \prod_{t=1}^T p_\theta(y_t|y_{1:t-1}). \tag{3}$$

For nonlinear and/or non-Gaussian models, the one-step predictive distribution $p_\theta(y_t|y_{1:t-1})$ is intractable and therefore the parameter posterior is also intractable. However, these quantities can be estimated e.g. using Sequential Monte Carlo (SMC) [Doucet and Johansen, 2011], Markov chain Monte Carlo (MCMC) [Robert and Casella, 1999] or a combination of the two. The latter solution is referred to as particle MCMC (PMCMC) [Andrieu and Roberts, 2009, Andrieu et al., 2010] and enables routine Bayesian parameter inference in general SSMs (1).

⋆ Supported by the project Probabilistic modeling of dynamical systems (Contract number: 621-2013-5524) funded by the Swedish Research Council.

Earlier work in the area of Bayesian parameter inference includes e.g. Cappé et al. [2005], Ninness and Henriksen [2010] and Peterka [1981]. PMCMC has earlier been used for nonlinear inference in e.g. finance [Pitt et al., 2012], social network analysis [Everitt, 2012] and system identification [Dahlin et al., 2013]. In the latter, we propose a method using Particle Metropolis-Hastings (PMH) with a proposal based on first-order information about the posterior.

In this work, we improve the performance of the PMH algorithm by also incorporating second-order information into the proposal. This draws upon results presented by Girolami and Calderhead [2011] for the Metropolis-Hastings (MH) algorithm and can be seen as a particle analogue to the manifold Metropolis Adjusted Langevin Algorithm (mMALA).

By including the Hessian, the proposal is given the ability to automatically adjust the step length during the run. This has the benefit of shortening the burn-in period and simplifies the tedious tuning, as the proposal is scale-invariant. Note, that this is similar to a Newton-based optimisation algorithm, which also enjoys the same invariance.

Another improvement is the use of particle smoothers with linear complexity for estimating the first-order and second-order information. This greatly decreases the computational cost of the algorithm compared to our earlier work, which has a quadratic complexity in the number of particles. The proposed method is illustrated on two SSMs, which shows some of the possible benefits of using the second-order proposal.

2. CONSTRUCTING SECOND-ORDER PROPOSALS

As previously stated, direct computation of the parameter posterior distribution (2) is often intractable. Instead, we make use of the Metropolis-Hastings (MH) algorithm [Metropolis et al., 1953, Hastings, 1970, Robert and Casella, 1999] to sample from the posterior by the use of a Markov chain with certain properties. The chain is constructed so that its stationary distribution is the posterior $p(\theta|y_{1:T})$, from which we would like to sample.

The (ideal) MH algorithm is an iterative procedure where two steps are carried out during each iteration: (i) sample parameters from a *proposal distribution*, $\theta'' \sim q(\theta''|\theta')$, where θ' denotes the parameters from the previous state of the Markov chain, and (ii) accept or reject the new parameters with the *acceptance probability*,

$$\alpha(\theta'', \theta') = 1 \wedge \frac{p(\theta'')}{p(\theta')} \frac{p_{\theta''}(y_{1:T})}{p_{\theta'}(y_{1:T})} \frac{q(\theta'|\theta'')}{q(\theta''|\theta')}, \tag{4}$$

where we introduce the operator $a \wedge b = \min\{a, b\}$.

Recall that the likelihood $p_\theta(y_{1:T})$ is intractable for the general SSM (1). In Section 4, we discuss how to solve this particular problem, while still making sure that the Markov chain converges to the parameter posterior. This is done by replacing the intractable likelihood with an unbiased estimate resulting in an *exact approximation* of the MH algorithm [Andrieu et al., 2010].

In this section, we construct a proposal that makes use of the first-order and second-order information about the posterior. After this, we discuss how to construct estimators for the required intractable quantities using SMC methods.

2.1 Laplace approximation of the log-posterior distribution

A proposal distribution can be constructed by using a Laplace approximation [Robert and Casella, 1999] of the log-posterior distribution. Consider a second-order Taylor expansion of $\log p(\theta''|y_{1:T})$ around θ',

$$\begin{aligned}\log p(\theta''|y_{1:T}) \approx \log p(\theta'|y_{1:T}) \\ + (\theta'' - \theta')^\top \nabla \log p(\theta|y_{1:T})\Big|_{\theta=\theta'} \\ + \frac{1}{2}(\theta'' - \theta')^\top \nabla^2 \log p(\theta|y_{1:T})\Big|_{\theta=\theta'} (\theta'' - \theta').\end{aligned}$$

By taking the exponential of both sides and completing the square, we obtain

$$\begin{aligned} p(\theta''|y_{1:T}) &= \mathcal{N}(\theta''; \theta' + \mathcal{G}_T(\theta'), \mathcal{W}_T(\theta')), \text{ with} \\ \mathcal{W}_T^{-1}(\theta') &\triangleq \mathcal{I}_T(\theta') - \nabla^2 \log \pi(\theta)\big|_{\theta=\theta'}, \\ \mathcal{G}_T(\theta') &\triangleq \mathcal{W}_T(\theta') \left[\mathcal{S}_T(\theta') + \nabla \log \pi(\theta)\big|_{\theta=\theta'}\right], \end{aligned}$$

which is discussed in e.g. Robert and Casella [1999]. Here, we introduced the notation $\mathcal{S}_T(\theta') \triangleq \nabla \log p_\theta(y_{1:T})|_{\theta=\theta'}$ and $\mathcal{I}_T(\theta') \triangleq -\nabla^2 \log p_\theta(y_{1:T})|_{\theta=\theta'}$ for the gradient and the negative Hessian of the log-likelihood, respectively.

In Robert and Casella [1999], the authors discard the second-order information $\mathcal{W}_T(\theta)$ from the expression by replacing it with a constant diagonal $d \times d$-matrix. Here, we instead keep the second-order information and guided by the Laplace approximation, suggest the use of the proposal,

$$\begin{aligned} q\Big(\theta''|\theta', \mathcal{S}_T(\theta'), \mathcal{I}_T(\theta')\Big) \\ = \mathcal{N}\left(\theta''; \theta' + \frac{\Gamma^2}{2}\mathcal{G}_T(\theta'), \Gamma^2 \mathcal{W}_T(\theta')\right), \end{aligned} \tag{5}$$

where $\Gamma = \mathsf{diag}(\gamma)$ denotes a diagonal matrix with γ being a scalar or a d-vector with step-length(s). We use the former in the second-order proposal because of its scale-invariance property. In the zeroth-order and first-order proposals (introduced below) a vector is often needed to use different step-lengths for each parameter.

2.2 Properties of the proposal distribution

We refer to the expression in (5) as the *second-order proposal*, since it makes use of both the gradient and the Hessian in proposing new parameters. If the Hessian of the log-posterior is replaced with a $d \times d$-identity matrix, $\mathcal{W}_T(\theta) \equiv \mathbf{I}_d$, a *first-order proposal* is obtained. Lastly, if the gradient is removed as well, $\mathcal{G}_T(\theta) \equiv 0$, a *zeroth-order proposal* is obtained. This proposal distribution is equivalent to a Gaussian random walk proposal, which is a common standard choice when using the MH algorithm.

We note in the passing that the second-order proposal has a statistical and geometrical interpretation. The gradient and the negative Hessian of the log-likelihood are often referred to as the *score function* and the *Fisher information matrix*, respectively. From such a perspective, the proposal in (5) is shown in Girolami and Calderhead [2011] to be a random walk on a Riemann manifold with constant curvature using the information matrix as the metric.

The convergence of the first-order proposal is analysed by Roberts and Rosenthal [1998] and under certain assumptions it require $\mathcal{O}(d^{-1/3})$ steps to converge to the stationary distribution. This is compared with $\mathcal{O}(d)$ steps for the zeroth-order proposal. Therefore the first-order proposal is more efficient as the number of parameters d increases. To the best of the authors' knowledge, no analysis has been published for the second-order proposal. However, numerical comparisons are presented in Section 5 which could support that the properties of the first-order proposal also carries over the the second-order proposal.

Note that, the MH algorithm with the second-order proposal depends on the likelihood, gradient and negative Hessian, which for the general SSM (1) are intractable. Therefore, we now continue with discussing SMC methods which can be used to solve this problem.

3. ESTIMATING SECOND-ORDER PROPOSALS

SMC is a family of algorithms used to sample from a sequence of probability distributions. A typical application of SMC methods is to sample from the filtering and smoothing distribution in SSMs. In this setting, we refer to SMC methods as *particle filters* and *particle smoothers*, respectively. Here, we limit ourselves to the auxiliary particle filter (APF) [Pitt and Shephard, 1999] and the fixed-lag (FL) particle smoother [Kitagawa and Sato, 2001]. For more information regarding SMC, see e.g. Doucet and Johansen [2011] and Del Moral et al. [2006].

3.1 Auxiliary particle filter

We use the APF to compute an estimate of the likelihood and the latent states of the SSM (1). An APF targeting the smoothing distribution $p_\theta(x_{1:t}|y_{1:t})$ generates a particle system using N particles $\{x_{1:t}^{(i)}, w_t^{(i)}\}_{i=1}^N$. This can be used to estimate the smoothing distribution,

$$\widehat{p}_\theta(\mathrm{d}x_{1:t}|y_{1:t}) \triangleq \sum_{i=1}^N \frac{w_t^{(i)}}{\sum_{k=1}^N w_t^{(k)}} \delta_{x_{1:t}^{(i)}}(\mathrm{d}x_{1:t}), \tag{6}$$

where $w_t^{(i)}$ and $x_{1:t}^{(i)}$ denote the unnormalised weight and the state trajectory of particle i from time 1 to t, respectively. Here, $\delta_z(\mathrm{d}x_{1:t})$ denotes the Dirac measure in the point z. The particle system is generated sequentially by the APF in two steps: (i) the sampling/propagation step, and (ii) the weighting step.

In the first step, the particle system from the previous time step $t-1$ is resampled and propagated to generate an unweighted particle system at time t. This can be seen as sampling from a proposal kernel,

$$\{a_t^{(i)}, x_t^{(i)}\} \sim \frac{w_{t-1}}{\sum_{k=1}^N w_{t-1}^{(k)}} R_\theta(x_t|x_{t-1}^{a_t}, y_t), \tag{7}$$

where we append the sampled particle to the trajectory by $x_{1:t}^{(i)} = \{x_{1:t-1}^{a_t^{(i)}}, x_t^{(i)}\}$. Here, $a_t^{(i)}$ denotes the *ancestor index*, i.e. the index of the particle at time $t-1$, from which $x_t^{(i)}$ originates. Furthermore, $R_\theta(x_t|x_{t-1}^{a_t}, y_t)$ denotes some propagation kernel from which we can sample a new particle at time t given the ancestor particle at time $t-1$.

In the second step, the particle weights are computed as

$$w_t^{(i)} = W_\theta(x_t^{(i)}, x_{t-1}^{a_t^{(i)}}) \triangleq \frac{g_\theta(y_t|x_t^{(i)}) f_\theta(x_t^{(i)}|x_{t-1}^{a_t^{(i)}})}{R_\theta\left(x_t^{(i)}|x_{t-1}^{a_t^{(i)}}, y_t\right)}. \tag{8}$$

Hence, the particle system at time t can be estimated recursively using the two steps in the APF.

3.2 Estimation of the likelihood

The likelihood for the general SSM (1) can be estimated using the particle systems obtained from the APF. This is done by first writing the one-step predictive density as

$$\begin{aligned} p_\theta(y_t|y_{1:t-1}) &= \int g_\theta(y_t|x_t) f_\theta(x_t|x_{t-1}) p_\theta(x_{t-1}|y_{1:t-1}) \,\mathrm{d}x_{t-1:t} \\ &= \int W_\theta(x_t, x_{t-1}) R_\theta(x_t|x_{t-1}, y_t) p_\theta(x_{t-1}|y_{1:t-1}) \,\mathrm{d}x_{t-1:t}, \end{aligned}$$

where we have multiplied and divided with the propagation kernel $R_\theta(\cdot)$. To approximate the integral, we note that the (unweighted) particle pairs $\{x_{t-1}^{a_t}, x_t\}$ are approximately drawn from $R_\theta(x_t|x_{t-1}, y_t) p_\theta(x_{t-1}|y_{1:t-1})$. Consequently, we obtain the Monte Carlo approximation

$$p_\theta(y_t|y_{1:t-1}) \approx \frac{1}{N} \sum_{i=1}^N W_\theta(x_t^{(i)}, x_{t-1}^{a_t^{(i)}}) = \frac{1}{N} \sum_{i=1}^N w_t^{(i)}.$$

By inserting this approximation into (3) we obtain the particle estimate of the likelihood,

$$p_\theta(y_{1:T}) = \prod_{t=1}^T \left(\frac{1}{N} \sum_{i=1}^N w_t^{(i)} \right). \tag{9}$$

This likelihood estimator has been studied extensively in the SMC literature. The estimator is consistent and unbiased, see e.g. Pitt et al. [2012] and Proposition 7.4.1 in Del Moral [2004]. Remember, that the unbiasedness is an essential property for the exact approximation of the MH algorithm and therefore also for our algorithm.

3.3 Estimation of the log-likelihood gradient

To estimate the gradient of the log-likelihood $\mathcal{S}_T(\theta)$ using SMC methods, we employ *Fisher's identity* [Fisher, 1925, Cappé et al., 2005, Ninness et al., 2010],

$$\nabla \log p_\theta(y_{1:T}) = \mathbb{E}_\theta \left[\nabla \log p_\theta(x_{1:T}, y_{1:T}) \middle| y_{1:T} \right]. \tag{10}$$

For the general SSM (1), we have

$$p_\theta(x_{1:T}, y_{1:T}) = \mu(x_0) \prod_{t=1}^T f_\theta(x_t|x_{t-1}) g_\theta(y_t|x_t), \tag{11}$$

which inserted into (10) results in

$$\begin{aligned} \nabla \log p_\theta(y_{1:T}) &= \sum_{t=1}^T \int \xi_\theta(x_{t-1:t}) p_\theta(x_{t-1:t}|y_{1:T}) \,\mathrm{d}x_{t:t-1}, \\ \xi_\theta(x_{t-1:t}) &= \nabla \log f_\theta(x_t|x_{t-1}) + \nabla \log g_\theta(y_t|x_t). \end{aligned}$$

Hence, $\nabla \log p_\theta(y_{1:T})$ depends on the intractable two-step $p_\theta(x_{t-1:t}|y_{1:T})$ smoothing distribution.

In Poyiadjis et al. [2011], this quantity is computed using the APF directly or by using a forward smoother (FS) [Del Moral et al., 2010]. The drawback of the first approach is poor accuracy due to particle degeneracy. The second approach is computationally costly as the FS algorithm has a computational complexity of $\mathcal{O}(N^2T)$ compared to $\mathcal{O}(NT)$ for the APF.

In this paper, we instead make use of the FL-smoother [Kitagawa and Sato, 2001, Olsson et al., 2008] which has the same computational cost as the APF, but better accuracy. This follows from that the FL-smoother experience less problems with particle degeneracy compared to the APF. The FL-smoother relies on the assumption that the SSM (1) is mixing fast. That is, we can use the approximation $p_\theta(x_t|y_{1:T}) \approx p_\theta(x_t|y_{1:\kappa_t})$, with $\kappa_t = \min\{t+\Delta, T\}$ and where Δ denotes some lag. Hence, the smoothing distribution of x_t is not strongly influenced by measurements obtained after some time κ_t.

By marginalisation of (6) over $x_{1:t-2}$ and $x_{t+1:\kappa_t}$, we obtain the *empirical two-step smoothing distribution* as

$$\widehat{p}_\theta(\mathrm{d}x_{t-1:t}|y_{1:\kappa_t}) \triangleq \sum_{i=1}^N w_{\kappa_t}^{(i)} \delta_{\tilde{x}_{\kappa_t,t-1:t}^{(i)}}(\,\mathrm{d}x_{t-1:t}), \tag{12}$$

where we use the notation $\tilde{x}_{\kappa_t,t}^{(i)} = x_t^{a_{\kappa_t,t}^{(i)}}$. Here, we let $a_{\kappa_t,t}^{(i)}$ denote the ancestor index of particle $x_{\kappa_t}^{(i)}$ at time t. Inserting (11) and (12) into (10) gives the estimate of the gradient

$$\widehat{\mathcal{S}}_T(\theta) = \sum_{t=1}^T \sum_{i=1}^N w_{\kappa_t}^{(i)} \xi_\theta(\tilde{x}_{\kappa_t,t}^{(i)}, \tilde{x}_{\kappa_t,t-1}^{(i)}). \tag{13}$$

In Olsson et al. [2008], the statistical properties of the FL-smoother are analysed. It is shown that the lag $\Delta^\star \propto \log T$ minimises the mean squared error of the state estimates.

It is also shown that the resulting estimates are biased and this could be a significant problem in many applications. However in our setting, the bias is later compensated for by the accept/reject-procedure in the MH algorithm and the invariance property is retained.

3.4 Estimation of the negative log-likelihood Hessian

The negative Hessian $\mathcal{I}_T(\theta)$ of the log-likelihood can be estimated using SMC methods in combination with *Louis' identity* [Louis, 1982, Cappé et al., 2005],

$$\begin{aligned}-\nabla^2 \log p_\theta(y_{1:T}) = [\nabla \log p_\theta(y_{1:T})]^2 \\ - \mathbb{E}_\theta \left[[\nabla \log p_\theta(x_{1:T}, y_{1:T})]^2 | y_{1:T} \right] \\ - \mathbb{E}_\theta \left[\nabla^2 \log p_\theta(x_{1:T}, y_{1:T}) | y_{1:T} \right], \quad (14)\end{aligned}$$

where we introduce $v^2 = vv^\top$ for some vector v. Here, we make use of the APF based smoother proposed in Poyiadjis et al. [2011] for estimating $\mathcal{I}_T(\theta)$. Here, the FL-smoother cannot be readily used for this problem as it cannot be used to estimate the required distributions. Instead, we can compute the negative Hessian using a recursive scheme from $t = 1$ to T of the form

$$\widehat{\beta}_\theta(x_t^{(i)}) = \widehat{\beta}_\theta(\tilde{x}_{t,t-1}^{(i)}) + \xi_\theta(x_t^{(i)}, \tilde{x}_{t,t-1}^{(i)}), \quad (15a)$$

$$\widehat{\eta}_\theta(x_t^{(i)}) = \widehat{\eta}_\theta(\tilde{x}_{t,t-1}^{(i)}) + \zeta_\theta(x_t^{(i)}, \tilde{x}_{t,t-1}^{(i)}), \quad (15b)$$

where we introduce the quantity

$$\zeta_\theta(x_{t-1:t}) = \nabla^2 \left[\log f_\theta(x_t|x_{t-1}) + \log g_\theta(y_t|x_t) \right].$$

The estimate of the negative Hessian is given by

$$\widehat{\mathcal{I}}_T(\theta) = \left[\widehat{\mathcal{S}}_T(\theta) \right]^2 - \sum_{i=1}^N w_t^{(i)} \left[\widehat{\beta}_\theta(x_t^{(i)})^2 + \widehat{\eta}_\theta(x_t^{(i)}) \right]. \quad (16)$$

3.5 SMC algorithm

In Algorithm 1, we present the complete algorithm that combines the APF and the FL-smoother to compute estimates of the gradient and negative Hessian. The primary outputs from this algorithm are the estimates of the likelihood, the gradient and the negative Hessian given a parameter θ.

In our experience, the off-diagonal elements in the information matrix are often difficult to estimate with good accuracy. Therefore, we only use the diagonal elements of the information matrix in the remainder of this work. This retains the property that the second-order proposal is scale-invariant, but without taking the curvature into account. Also, this does not allow for any covariation in the parameters proposed in the algorithm. That is, the parameters are assumed to be independent, which could lead to poor exploration of non-isotropic posteriors.

4. PARTICLE METROPOLIS-HASTINGS

From the previous development, we know how to estimate the various quantities needed for using the MH algorithm with the second-order proposal. Recall, that the exact approximation of the MH algorithm guarantees that the stationary distribution of the Markov chain remains the parameter posterior, see Andrieu et al. [2010]. This result only requires that the log-likelihood estimate is unbiased.

In fact, we are allowed to use the entire particle system in the proposal, see Dahlin et al. [2013]. This opens up for using the second-order proposal, since we have demonstrated that the gradient and Hessian information can be computed using the particle system. Note, that these estimates are biased, but this does not affect the invariance property as this is compensated for by the accept/reject mechanism.

Algorithm 1 Sequential Monte Carlo for estimation of the gradient and Hessian of the log-likelihood

INPUTS: SSM (1), $y_{1:T}$ (observations), $R_\theta(\cdot)$ (particle proposal), N (no. particles) and Δ (lag).
OUTPUTS: $\widehat{p}_\theta(y_{1:T})$, $\widehat{\mathcal{S}}_T(\theta)$ and $\widehat{\mathcal{I}}_T(\theta)$ (est. of likelihood, gradient and negative Hessian).

1: Initialise the particles $x_0^{(i)}$ for $i = 1, \ldots, N$.
2: **for** $t = 1$ to T **do**
3: Sample (7) for $i = 1, \ldots, N$.
4: Compute (8) for $i = 1, \ldots, N$.
5: **end for**
6: Compute (9), (13) and (16) to obtain $\widehat{p}_\theta(y_{1:T})$, $\widehat{\mathcal{S}}_T(\theta)$ and $\widehat{\mathcal{I}}_T(\theta)$.

Algorithm 2 Second-order Particle Metropolis-Hastings for Bayesian parameter inference in nonlinear SSMs

INPUTS: Algorithm 1, M (no. PMH iterations), θ_0 (initial parameter), γ (proposal step length).
OUTPUT: $\theta = \{\theta_1, \ldots, \theta_M\}$ (samples from the parameter posterior).

1: Run Algorithm 1 to obtain $\widehat{p}_{\theta_0}(y_{1:T})$, $\widehat{\mathcal{S}}_T(\theta_0)$ and $\widehat{\mathcal{I}}_T(\theta_0)$.
2: **for** $k = 1$ to M **do**
3: Sample $\theta' \sim q(\theta'|\theta_{k-1}, \widehat{\mathcal{S}}_T(\theta_{k-1}), \widehat{\mathcal{I}}_T(\theta_{k-1}))$ using (5).
4: Run Algorithm 1 to obtain $\widehat{p}_{\theta'}(y_{1:T})$, $\widehat{\mathcal{S}}_T(\theta')$ and $\widehat{\mathcal{I}}_T(\theta')$.
5: Sample $u_k \sim \mathcal{U}[0, 1]$.
6: Compute (17) to obtain $\alpha(\theta', \theta_{k-1})$.
7: **if** $u_k < \alpha(\theta', \theta_{k-1})$ **then**
8: {Accept the proposed parameter}
9: $\theta_k \leftarrow \theta'$ and $\widehat{p}_{\theta_k}(y_{1:T}) \leftarrow \widehat{p}_{\theta'}(y_{1:T})$.
10: $\widehat{\mathcal{S}}_T(\theta_k) \leftarrow \widehat{\mathcal{S}}_T(\theta')$ and $\widehat{\mathcal{I}}_T(\theta_k) \leftarrow \widehat{\mathcal{I}}_T(\theta')$.
11: **else**
12: {Reject the proposed parameter}
13: $\theta_k \leftarrow \theta_{k-1}$ and $\widehat{p}_{\theta_k}(y_{1:T}) \leftarrow \widehat{p}_{\theta_{k-1}}(y_{1:T})$.
14: $\widehat{\mathcal{S}}_T(\theta_k) \leftarrow \widehat{\mathcal{S}}_T(\theta_{k-1})$ and $\widehat{\mathcal{I}}_T(\theta_k) \leftarrow \widehat{\mathcal{I}}_T(\theta_{k-1})$.
15: **end if**
16: **end for**

Hence, we can use the MH algorithm together with Algorithm 1 to form the final method in Algorithm 2. The acceptance probability follows from (4) as

$$\alpha(\theta'', \theta') = 1 \wedge \frac{\widehat{p}_{\theta''}(y_{1:T})}{\widehat{p}_{\theta'}(y_{1:T})} \frac{p(\theta'')}{p(\theta')} \frac{q\big(\theta'|\theta'', \widehat{\mathcal{S}}_T(\theta''), \widehat{\mathcal{I}}_T(\theta'')\big)}{q\big(\theta''|\theta', \widehat{\mathcal{S}}_T(\theta'), \widehat{\mathcal{I}}_T(\theta')\big)}. \quad (17)$$

This is the full PMH procedure that uses the second-order proposal. The complexity of the algorithm is linear in the number of particles N and in the number of iterations M. The user-choices include the particle proposal kernel $R_\theta(\cdot)$, the lag Δ, the number of particles N and the number of iterations M. Also, the step-sizes γ needs to be tuned for each model, this is further discussed in the subsequent section.

5. NUMERICAL ILLUSTRATIONS

We continue by illustrating the method proposed in Algorithm 2 for parameter estimation in nonlinear SSMs. First, we consider a linear Gaussian state space (LGSS)

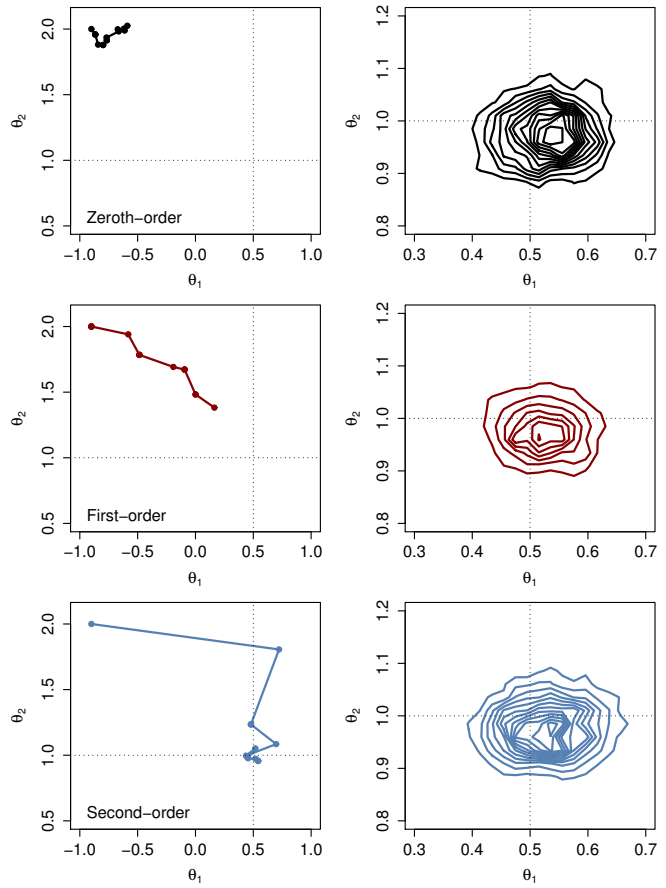

Fig. 1. The trace plots (left) of the first 15 iterations and contour plots of the parameter posterior estimates (right) from the three proposals used in Algorithm 2 on the LGSS model in (18). The dotted lines corresponds to the *true* parameters from which the data were generated.

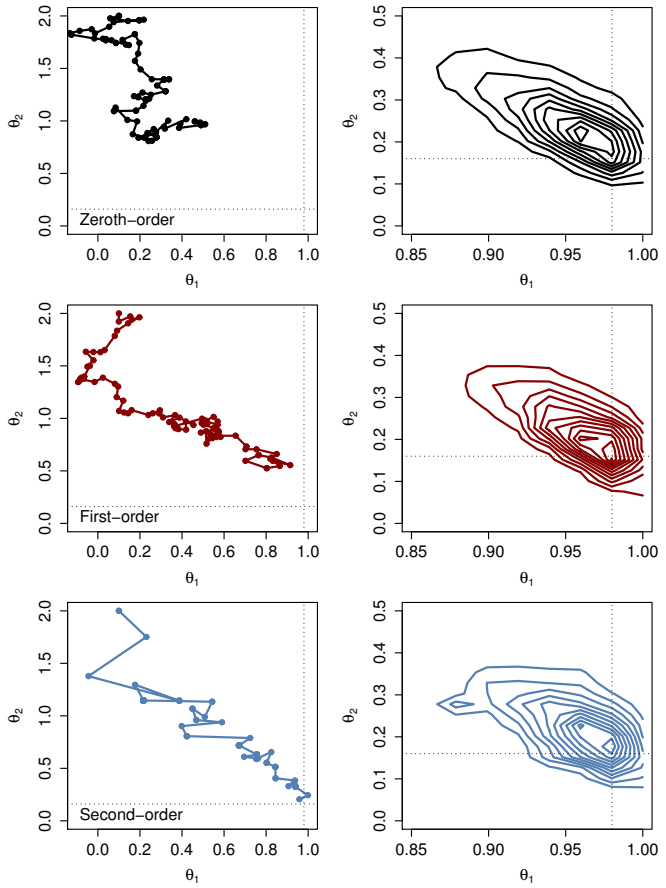

Fig. 2. The trace plots (left) of the first 90 iterations and contour plots of the parameter posterior estimates (right) from the three proposals used in Algorithm 2 on the stochastic volatility model in (19). The dotted lines corresponds to the *true* parameters from which the data were generated.

model and then a popular stochastic volatility model with a nonlinear observation process.

We compare the three different variations of the proposal in (5), i.e. zeroth-order, first-order and second-order. The step length γ is selected individually for each method such that the acceptance rate is about 40%. Also, we use the same step length for all the parameters to simplify calibration, i.e. γ is selected as a scalar.

5.1 Linear Gaussian state space model

Consider the LGSS model,

$$x_{t+1}|x_t \sim \mathcal{N}\left(x_{t+1}; \theta_1 x_t, \theta_2^2\right), \tag{18a}$$

$$y_t|x_t \sim \mathcal{N}\left(y_t; x_t, 0.1^2\right), \tag{18b}$$

with parameters $\theta^\star = \{\theta_1^\star, \theta_2^\star\} = \{0.5, 1.0\}$. We use $T = 250$ time steps, $N = 5\,000$ particles, $M = 10\,000$ (discarding the first $5\,000$ iterations as burn-in) and the bootstrap APF with $R_\theta(\cdot) = f_\theta(\cdot)$ and systematic resampling. The fixed-lag is chosen as $\Delta = 12$. Here, we use improper priors for the parameters, i.e. $p(\theta_1) = \mathcal{U}[-1, 1]$ and $p(\theta_2) = \mathcal{U}[0, \infty]$. The step lengths are tuned as $\gamma^{(0)} = 0.04$, $\gamma^{(1)} = 0.065$, $\gamma^{(2)} = 1.50$, for the zeroth-order, first-order and second-order proposals respectively.

In the left part of Figure 1, we present the trace plots of the burn-in phase of the algorithms. We clearly see the advantage of using the second-order proposal, as it adjusts its step size quickly to reach the neighbourhood of the true parameters. The contour plots of the estimated parameter posteriors are shown in the right part of Figure 1, where we see that all proposals give similar parameter posterior estimates.

5.2 Nonlinear stochastic volatility model

Consider the Hull-White stochastic volatility model [Hull and White, 1987],

$$x_{t+1}|x_t \sim \mathcal{N}\left(x_{t+1}; \theta_1 x_t, \theta_2^2\right), \tag{19a}$$

$$y_t|x_t \sim \mathcal{N}\left(y_t; 0, 0.65^2 \exp(x_t)\right), \tag{19b}$$

with parameters $\theta^\star = \{\theta_1^\star, \theta_2^\star\} = \{0.98, 0.16\}$. We use the same settings and priors as for the LGSS example. The step lengths are tuned as $\gamma^{(0)} = 0.05$, $\gamma^{(1)} = 0.045$, $\gamma^{(2)} = 1.70$, respectively.

In Figure 2, we present the burn-in trace plots and the parameter posterior distributions for the three proposals. The behaviours of the proposals are similar to the LGSS example and using the second-order proposal again shortens the burn-in, but keeps a similar parameter posterior estimate.

6. CONCLUSIONS

We have proposed a novel algorithm based on PMH and particle smoothing for Bayesian parameter inference in nonlinear SSMs. The algorithm uses first-order and second-order information in the proposal to improve the performance of the *vanilla* PMH algorithm. The complexity of the proposed algorithm is linear in the number of particles, which makes it a practical alternative to other smoothing-based inference algorithms.

We have seen examples illustrating that using the second-order proposals shortens the burn-in phase. Also, the second-order proposal is simpler to tune as it is scale-invariant and automatically rescales the step length in each direction. In the MH algorithm, it is known that adding first-order information into the proposal improves the performance in high dimensional problems. Hopefully, similar results can be found for the second-order proposal in the PMH framework.

Future work includes theoretical analysis of the convergence rate and scaling properties of the algorithm. Also, it would be interesting to explore the use of Hamiltonian MCMC [Neal, 2010, Girolami and Calderhead, 2011] ideas in this setting. This would potentially improve the mixing of the Markov chain and could open up for the possibility of solving problems with hundreds of parameters .

At `http://users.isy.liu.se/en/rt/johda87/`, we provide code and that can be used to reproduce some of the numerical illustrations in this paper.

REFERENCES

C. Andrieu and G. O. Roberts. The pseudo-marginal approach for efficient Monte Carlo computations. *The Annals of Statistics*, 37 (2):697–725, 2009.

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

O. Cappé, E. Moulines, and T. Rydén. *Inference in Hidden Markov Models*. Springer, 2005.

J. Dahlin, F. Lindsten, and T. B. Schön. Particle Metropolis Hastings using Langevin dynamics. In *Proceedings of the 38th International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Vancouver, Canada, May 2013.

P. Del Moral. *Feynman-Kac Formulae - Genealogical and Interacting Particle Systems with Applications*. Probability and its Applications. Springer, 2004.

P. Del Moral, A. Doucet, and A. Jasra. Sequential Monte Carlo samplers. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(3):411–436, 2006.

P. Del Moral, A. Doucet, and S. Singh. Forward smoothing using sequential Monte Carlo. *Pre-print*, 2010. arXiv:1012.5390v1.

A. Doucet and A. Johansen. A tutorial on particle filtering and smoothing: Fifteen years later. In D. Crisan and B. Rozovsky, editors, *The Oxford Handbook of Nonlinear Filtering*. Oxford University Press, 2011.

R. G. Everitt. Bayesian parameter estimation for latent Markov random fields and social networks. *Journal of Computational and Graphical Statistics*, 21(4):940–960, 2012.

R. A. Fisher. Theory of statistical estimation. *Mathematical Proceedings of the Cambridge Philosophical Society*, 22(05):700–725, 1925.

M. Girolami and B. Calderhead. Riemann manifold Langevin and Hamiltonian Monte Carlo methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(2):1–37, 2011.

W. K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

J. Hull and A. White. The pricing of options on assets with stochastic volatilities. *The Journal of Finance*, 42(2):281–300, 1987.

G. Kitagawa and S. Sato. Monte carlo smoothing and self-organising state-space model. In A. Doucet, N. de Fretias, and N. Gordon, editors, *Sequential Monte Carlo methods in practice*, pages 177–195. Springer, 2001.

T. A. Louis. Finding the observed information matrix when using the EM algorithm. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 44(02):226–233, 1982.

N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092, 1953.

R. M. Neal. MCMC using Hamiltonian dynamics. In S. Brooks, A. Gelman, G. Jones, and X-L. Meng, editors, *Handbook of Markov Chain Monte Carlo*. Chapman & Hall/ CRC Press, June 2010.

B. Ninness and S. Henriksen. Bayesian system identification via Markov chain Monte Carlo techniques. *Automatica*, 46(1):40–51, 2010.

B. Ninness, A. Wills, and T. B. Schön. Estimation of general nonlinear state-space systems. In *Proceedings of the 49th IEEE Conference on Decision and Control (CDC)*, Atlanta, USA, December 2010.

J. Olsson, O. Cappé, R. Douc, and E. Moulines. Sequential Monte Carlo smoothing with application to parameter estimation in nonlinear state space models. *Bernoulli*, 14(1):155–179, 2008.

V. Peterka. Bayesian system identification. *Automatica*, 17(1):41–53, 1981.

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

M. K. Pitt, R. S. Silva, P. Giordani, and R. Kohn. On some properties of Markov chain Monte Carlo simulation methods based on the particle filter. *Journal of Econometrics*, 171(2):134–151, 2012.

G. Poyiadjis, A. Doucet, and S. S. Singh. Particle approximations of the score and observed information matrix in state space models with application to parameter estimation. *Biometrika*, 98(1):65–80, 2011.

C. P. Robert and G. Casella. *Monte Carlo Statistical Methods*. Springer, 1 edition, 1999.

G. O. Roberts and J. S. Rosenthal. Optimal Scaling of Discrete Approximations to Langevin Diffusions. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 60(1):255–268, 1998.