

 Open access • Proceedings Article • DOI:10.1109/ICASSP.2013.6638879

Particle metropolis hastings using Langevin dynamics — Source link

Johan Dahlin, Fredrik Lindsten, Thomas B. Schön

Institutions: Linköping University

Published on: 26 May 2013 - International Conference on Acoustics, Speech, and Signal Processing

Topics: Markov chain Monte Carlo, Hybrid Monte Carlo, Monte Carlo method in statistical physics, Monte Carlo integration and Monte Carlo molecular modeling

Related papers:

- [Particle Markov chain Monte Carlo methods](#)
- [Filtering via Simulation: Auxiliary Particle Filters](#)
- [Monte Carlo Sampling Methods Using Markov Chains and Their Applications](#)
- [Feynman-Kac formulae : genealogical and interacting particle systems with applications](#)
- [Riemann manifold Langevin and Hamiltonian Monte Carlo methods](#)

Share this paper:    

View more about this paper here: <https://typeset.io/papers/particle-metropolis-hastings-using-langevin-dynamics-1qajmi0a75>

PARTICLE METROPOLIS HASTINGS USING LANGEVIN DYNAMICS

Johan Dahlin, Fredrik Lindsten, Thomas B. Schön

Division of Automatic Control, Linköping University, Linköping, Sweden.

E-mail: {johan.dahlin, lindsten, schon}@isy.liu.se.

ABSTRACT

Particle Markov Chain Monte Carlo (PMCMC) samplers allow for routine inference of parameters and states in challenging nonlinear problems. A common choice for the parameter proposal is a simple random walk sampler, which can scale poorly with the number of parameters. In this paper, we propose to use log-likelihood gradients, i.e. the score, in the construction of the proposal, akin to the Langevin Monte Carlo method, but adapted to the PMCMC framework. This can be thought of as a way to guide a random walk proposal by using drift terms that are proportional to the score function. The method is successfully applied to a stochastic volatility model and the drift term exhibits intuitive behaviour.

Index Terms— Bayesian inference, Sequential Monte Carlo, Particle Markov Chain Monte Carlo, Langevin Monte Carlo.

1. INTRODUCTION

Particle Markov Chain Monte Carlo (PMCMC) is a relatively new method for simultaneous Bayesian parameter and state inference in general nonlinear state-space models,

$$x_{t+1}|x_t \sim f_\theta(x_{t+1}|x_t), \quad (1a)$$

$$y_t|x_t \sim h_\theta(y_t|x_t), \quad (1b)$$

with latent state $x_t \in \mathcal{X}$, observation $y_t \in \mathcal{Y}$ and unknown static parameter $\theta \in \Theta$. We model θ as a random variable with prior density $\pi(\theta)$. Furthermore, $f_\theta(\cdot)$ denotes the transition kernel of the latent process, $h_\theta(\cdot)$ denotes the observation kernel and the initial state is distributed according to a density $\mu_\theta(\cdot)$.

PMCMC is one way to use sequential Monte Carlo (SMC), i.e. a particle filter, as a proposal mechanism within MCMC. This family of algorithms were derived and analysed in [1]. Similar ideas have appeared in previous work on pseudo-marginal Monte Carlo [2, 3]. Recent work in PMCMC has to a large extent been focused on improving the efficiency of the inherent SMC samplers. Some examples of this is the introduction of backward simulation [4, 5] and ancestor sampling [6] in the Particle Gibbs sampler and the use of fully adapted auxiliary particle filters within PMCMC [7].

In this work, we focus on the MCMC part of PMCMC. We provide an extension to the Particle Marginal Metropolis-Hastings (PMMH) algorithm, using well-known results from MCMC. This extension aims to enable the use of more of the available information when proposing new parameter values, than what is possible in the PMMH algorithm. PMMH is often viewed as an *exact approximation* of a marginal Metropolis-Hastings (MMH) sampler in the

parameter space Θ . SMC is then only used to provide an unbiased estimate of the intractable likelihood, which is used in computing the MMH acceptance probability. However, PMMH does in fact target a distribution on an extended space, i.e. a large space containing Θ , and there is more useful information available from the particle filter than just the likelihood estimate. To exploit this, we propose an extension to PMMH to allow for more general proposals, resulting in what we refer to as the Particle Metropolis-Hastings (PMH) algorithm. The possibility for such an extension was first mentioned in the discussions following the original PMCMC paper [8], but to our knowledge, it has not been further exploited for constructing efficient proposal kernels.

In particular, we make use of Fisher’s identity and the particle system generated by the particle filter to compute an estimate of the score function (i.e. the log-likelihood gradient). Methods for score estimation using particle filters have previous been proposed and used for maximum likelihood inference in e.g. [9, 10, 11, 12, 13].

In this contribution, we propose a new algorithm called Langevin Particle Metropolis-Hastings (LPMH). We combine the PMH algorithm with a forward smoother for score estimation and ideas from Langevin Monte Carlo (LMC) methods. LMC is a type of Hamiltonian Markov Chain Monte Carlo (HMC) method, with its roots in statistical physics. HMC was first introduced in [14] under the name Hybrid Monte Carlo, and has proved to be a very useful tool for proposal construction in general MCMC samplers, see e.g. [15, 16, 17, 18, 19].

2. SEQUENTIAL MONTE CARLO

Sequential Monte Carlo (SMC) samplers are a family of simulation methods for sequentially approximating a sequence of target distributions, see e.g. [20, 21]. For example, SMC can be used for state inference in nonlinear non-Gaussian state-space models (1). We introduce SMC in terms of the auxiliary particle filter (APF) [22]. The APF consists of two steps: (i) resampling and mutating particles and (ii) calculating the importance weights. Let $\{x_{1:t-1}^i, w_{t-1}^i\}_{i=1}^N$ be a weighted particle system targeting the joint smoothing density at time $t-1$, i.e. $p_\theta(x_{1:t-1} | y_{1:t-1})$. Then, the particle system defines an empirical distribution,

$$\hat{p}_\theta(dx_{1:t-1} | y_{1:t-1}) \triangleq \sum_{i=1}^N \frac{w_{t-1}^i}{\sum_{l=1}^N w_{t-1}^l} \delta_{x_{1:t-1}^i}(dx_{1:t-1}),$$

which approximates the target distribution. Here $\delta_z(dx)$ refers to a Dirac point-mass at z . The resampling and mutation step propagates the particles to time t by sampling from a proposal kernel,

$$(a_t^i, x_t^i) \sim M_{\theta,t}(a_t, x_t) \triangleq \frac{w_{t-1}^{a_t}}{\sum_{l=1}^N w_{t-1}^l} R_{\theta,t}(x_t | x_{t-1}^{a_t}), \quad (2)$$

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

The variable a_t is the index to an *ancestor particle* $x_{t-1}^{a_t}$ and $R_{\theta,t}$ is a proposal kernel which proposes a new particle at time t given this ancestor. In this formulation, the resampling is implicit and corresponds to sampling these ancestor indices.

In the weighting step, new importance weights are computed according to $w_t^i = W_{\theta,t}(x_t^i, x_{t-1}^{a_t^i})$, with

$$W_{\theta,t}(x_t, x_{t-1}) \triangleq \frac{h_\theta(y_t|x_t)f_\theta(x_t|x_{t-1})}{R_{\theta,t}(x_t|x_{t-1})}. \quad (3)$$

Here, $f_\theta(\cdot)$ and $h_\theta(\cdot)$ are given by the model in (1) and y_t denotes the observation at time t . This results in a new weighted particle system $\{x_{1:t}^i, w_t^i\}_{i=1}^N$, targeting the joint smoothing density at time t . The SMC sampler thus iterates between propagating particles with high weights forward in time and computing new importance weights given the measurements. The method is initialised by sampling from a proposal density $x_1^i \sim R_{\theta,1}(x_1)$ and assigning weights $w_1^i = W_{\theta,1}(x_1^i)$ where the weight function is given by $W_{\theta,1}(x_1) \triangleq h_\theta(y_1 | x_1)\mu_\theta(x_1)/R_1^\theta(x_1)$.

3. LANGEVIN PARTICLE METROPOLIS-HASTINGS

In this section, we introduce a general Particle Metropolis-Hastings (PMH) algorithm, which allows the entire particle system to be used for constructing a proposal in the parameter space. In particular, we use the particle system to estimate the score function, which is then used in an LMC proposal.

3.1. Particle Metropolis-Hastings

The PMMH algorithm [1] can be seen as an *exact approximation* of an idealised MMH sampler. More precisely, the method is designed as a standard Metropolis-Hastings sampler on the Θ -space, with some proposal density $q(\theta | \theta')$. A standard choice is to use a Gaussian random walk proposal, i.e.

$$q(\theta|\theta') = \mathcal{N}(\theta; \theta', \Sigma_\theta), \quad (4)$$

where Σ_θ denotes the random walk covariance matrix, θ' denotes the last accepted parameter and θ denotes a new proposed parameter. The target density is the posterior $p(\theta | y_{1:T}) \propto p_\theta(y_{1:T})\pi(\theta)$. For this target density, the acceptance probability will depend on the likelihood $p_\theta(y_{1:T})$, which in general is intractable for the model (1). To deal with this issue, PMMH uses an SMC sampler to compute an unbiased estimate of the likelihood,

$$\widehat{p}_\theta(y_{1:T}) \triangleq \prod_{t=1}^T \left(\frac{1}{N} \sum_{i=1}^N w_t^i \right). \quad (5)$$

This estimate is then used in place of the true likelihood when evaluating the acceptance probability. The phrase *exact approximation* refers to the fact that this seemingly approximate method, in fact admits the exact posterior $p(\theta | y_{1:T})$ as stationary distribution.

The way in which this property is established in [1], is to interpret the PMMH sampler as a standard MCMC sampler on an extended space. To formalise this, let $\mathbf{x}_t \triangleq \{x_t^1, \dots, x_t^N\} \in \mathcal{X}^N$ and $\mathbf{a}_t \triangleq \{a_t^1, \dots, a_t^N\} \in \{1, \dots, N\}^N$ be the collections of particles and ancestor indices, respectively, generated at time t of the SMC sampler. Define the space $\Omega \triangleq \mathcal{X}^{NT} \times \{1, \dots, N\}^{N(T-1)}$. It then follows that a complete pass of the SMC sampler for times $t = 1, \dots, T$, generates a collection of random variables $\{\mathbf{x}_{1:T}, \mathbf{a}_{2:T}\} \in \Omega$. This is exploited in PMMH, which is interpreted

as a Metropolis-Hastings sampler on the extended space $\Theta \times \Omega$, for which SMC is used as part of the proposal mechanism.

This interpretation also suggests that we have more freedom in designing the proposal for θ . At each iteration of PMMH, the state of the Markov chain is given by some point $\{\theta', \mathbf{x}'_{1:T}, \mathbf{a}'_{1:T}\} \in \Theta \times \Omega$. Consequently, we can allow the proposal for θ to depend on all these variables, and not only on θ' , as is done in PMMH. That is, we choose some proposal kernel according to,

$$q(\theta|\theta', \mathbf{x}'_{1:T}, \mathbf{a}'_{2:T}, y_{1:T}), \quad (6)$$

resulting in what we refer to as the PMH algorithm. Clearly, PMH contains PMMH as a special case. The use of a more general proposal kernel as in (6) allows us to make use of valuable information available in the particle system, which is otherwise neglected. In the discussion following the seminal PMCMC paper, it is indeed mentioned by [8] that this information can be useful in constructing better parameter proposal densities.

The validity of PMH can be assessed analogously to that of PMMH (see [1]), as the state trajectory proposal and the extended target remains the same in both algorithms. The only affected quantities are the proposal density (6) and the acceptance probability, which in PMH is given by,

$$\alpha(\theta, \theta') = 1 \wedge \frac{\pi(\theta)}{\pi(\theta')} \frac{\widehat{p}_\theta(y_{1:T})}{\widehat{p}_{\theta'}(y_{1:T})} \frac{q(\theta'|\theta, \mathbf{x}_{1:T}, \mathbf{a}_{2:T}, y_{1:T})}{q(\theta|\theta', \mathbf{x}'_{1:T}, \mathbf{a}'_{2:T}, y_{1:T})}. \quad (7)$$

Here $\pi(\theta)$ is the parameter prior density, $\widehat{p}_\theta(y_{1:T})$ is the likelihood estimate given in (5) and $a \wedge b \triangleq \min(a, b)$.

The PMH proposal suggested in (6) allows for using the observations and the entire particle system generated by the particle filter to propose new parameters. For instance, if a conjugate prior is used for θ , we can let (6) be the posterior of θ given $x_{1:T}$ and $y_{1:T}$ to mimic a Gibbs move. Whether or not this approach can be a useful alternative to the Particle Gibbs sampler when the conditional particle filter degenerates and a backward kernel is not available for smoothing, is a question which requires further investigation. In the next section, we focus on another useful piece of information, namely the score function.

3.2. Proposal using Langevin dynamics

Let $S_T(\theta) = \nabla \log p_\theta(y_{1:T})$ denote the score function and let $\mathcal{L}(\theta) = \nabla \log \pi(\theta)$ be the gradient of the log-prior. It follows that $\nabla \log p(\theta | y_{1:T}) = S_T(\theta) + \mathcal{L}(\theta)$. A Langevin diffusion with stationary distribution $p(\theta | y_{1:T})$ is thus given by the stochastic differential equation (SDE),

$$d\theta(\tau) = [S_T(\theta(\tau)) + \mathcal{L}(\theta(\tau))] \frac{d\tau}{2} + dB(\tau), \quad (8)$$

where B denotes Brownian motion. Hence, in theory, it is possible to draw samples from $p(\theta | y_{1:T})$ by simulating this SDE to stationarity. This idea underlies Langevin Monte Carlo (LMC), which uses a first order Euler discretisation of (8),

$$\theta_{\tau+1} = \theta_\tau + \frac{(\Delta\tau)^2}{2} [S_T(\theta_\tau) + \mathcal{L}(\theta_\tau)] + (\Delta\tau)z_\tau, \quad (9)$$

where $z_\tau \sim \mathcal{N}(0, I)$ and $\Delta\tau$ is the discretisation step size. To account for the discretisation error and ensure that $p(\theta | y_{1:T})$ is the stationary distribution of the process, a Metropolis-Hastings accept/reject decision is made after each simulation step.

Algorithm 1 Langevin Particle Metropolis-Hastings (LPMH)

Assume that a prior density $\pi(\theta)$ is specified. Assume that the parameter θ' , the score estimate $\widehat{S}_T(\theta')$ and the likelihood estimate $\widehat{p}_{\theta'}(y_{1:T})$ are available from the previous MCMC iteration. Then, the next iteration is as follows:

1. Sample θ according to (11) (using the previous parameter value and score estimate).
 2. Run an SMC sampler, targeting $p_{\theta}(x_{1:T} | y_{1:T})$ to obtain the particle system $\{\mathbf{x}_{1:T}, \mathbf{a}_{2:T}\}$ and an estimate of the likelihood $\widehat{p}_{\theta}(y_{1:T})$.
 3. Compute the score estimate $\widehat{S}_T(\theta)$ using the particle system $\{\mathbf{x}_{1:T}, \mathbf{a}_{2:T}\}$ and (10).
 4. Calculate the acceptance probability $\alpha(\theta, \theta')$ as in (7) using the contribution from the proposal density as given in (12).
 5. Sample a random number $u \sim \mathcal{U}[0, 1]$.
 6. If $u < \alpha(\theta, \theta')$, accept $\{\theta, \widehat{p}_{\theta}(y_{1:T}), \widehat{S}_T(\theta)\}$, otherwise retain $\{\theta', \widehat{p}_{\theta'}(y_{1:T}), \widehat{S}_T(\theta')\}$.
-

A thorough and accessible introduction to LMC is given in [23]. The information contained in the score function gives useful guidance for the parameter process. It is well-known [24, 25] that the LMC proposal scales better than a random-walk proposal, as the dimension of the parameter space increases.

To be able to make use of LMC, we need to compute the score function $S_T(\theta)$, which is intractable for the models under study. To make progress, we suggest to use an SMC estimate of the score function within the PMH algorithm, similarly to how an SMC estimate of the likelihood is used in PMMH. To estimate the score function, we use Fisher's identity (see e.g. [10]),

$$\nabla \log p_{\theta}(y_{1:T}) = \int \nabla \log p_{\theta}(x_{1:T}, y_{1:T}) p_{\theta}(x_{1:T} | y_{1:T}) dx_{1:T},$$

where $\log p_{\theta}(x_{1:T}, y_{1:T})$ is readily available from (1). Hence, computing the score function equates to solving a smoothing problem. We have access to the complete particle system $\{\mathbf{x}_{1:T}, \mathbf{a}_{1:T}\}$, which allows us to address this problem as part of the proposal construction in (6). This can be done in a range of different ways. The simplest is probably to make use of the filter/smoothers by [26], which is attractive due to its linear complexity in N . However, this method is known to suffer from path degeneracy and the variance of the score estimate grows at least quadratically with T [12].

Another option is to use the forward filter/backward smoother [10], or its forward-only version [9]. The smoothing estimate of the score function is here computed according to the latter alternative by the following recursion,

$$T_t^i(\theta) = \left[\sum_{k=1}^N W_{t-1}^k f_{\theta}(x_t^i | x_{t-1}^k) \right]^{-1} \sum_{j=1}^N W_{t-1}^j f_{\theta}(x_t^i | x_{t-1}^j) \\ \times \left[T_{t-1}^j(\theta) + \nabla \log g_{\theta}(y_t | x_t^j) + \nabla \log f_{\theta}(x_t^i | x_{t-1}^j) \right], \quad (10a)$$

$$\widehat{S}_t^i(\theta) = \sum_{i=1}^N W_t^i T_t^i(\theta), \quad (10b)$$

where \widehat{S}_t denotes the estimated score at time t . Note that this estimator is biased, which is true for both smoothers mentioned above, but

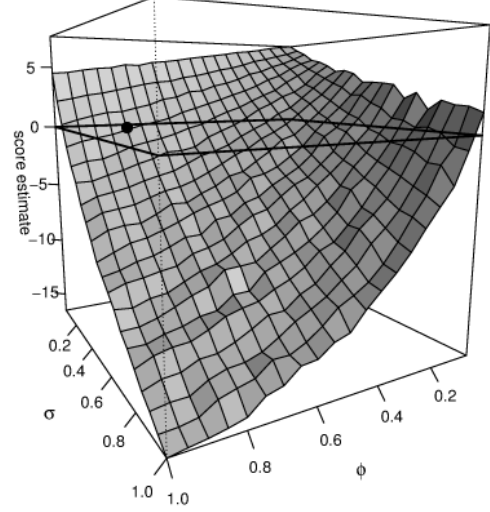


Fig. 1. The estimated score function for ϕ with $\mu = \mu^*$. The dot indicates the true parameters and the thick lines the zero level.

the bias decreases as N^{-1} . The computational complexity of this method scales as N^2 , but the variance only grows linearly with T . As mentioned above, other smoothers can also be used in PMH, e.g. the forward filter/backward simulator [27] and its variants [28, 29]. Note that this requires some further modifications due to the additional level of stochasticity in these algorithms.

Using an estimate $\widehat{S}_T(\theta) \approx S_T(\theta)$ according to (10), we construct a proposal kernel similarly to (9),

$$\theta = \theta' + \epsilon \left[\widehat{S}_T(\theta') + \mathcal{L}(\theta') \right] + z', \quad (11)$$

with $z' \sim \mathcal{N}(0, \Sigma_{\theta})$. The parameter ϵ denotes a step-length, scaling the score estimate and Σ_{θ} denotes a covariance matrix, possibly different from the one used in (4). The intuition behind this proposal is that the score function indicates a direction in which the likelihood is increasing. This information is useful as we are trying to explore regions of high posterior probability. The contribution to the acceptance probability (7) from this choice of proposal distribution is given by,

$$\frac{q(\theta' | \theta, \mathbf{x}_{1:T}, \mathbf{a}_{2:T}, y_{1:T})}{q(\theta | \theta', \mathbf{x}'_{1:T}, \mathbf{a}'_{2:T}, y_{1:T})} = \frac{\mathcal{N}(\theta'; \theta + \epsilon [\widehat{S}_T(\theta) + \mathcal{L}(\theta)], \Sigma_{\theta})}{\mathcal{N}(\theta; \theta' + \epsilon [\widehat{S}_T(\theta') + \mathcal{L}(\theta')], \Sigma_{\theta})}. \quad (12)$$

3.3. Final algorithm

The PMH algorithm can be formulated using the PMMH algorithm with the proposal density (6) and the acceptance probability (7). The Langevin Particle Metropolis-Hastings (LPMH) method in Algorithm 1, is a PMH algorithm with the special choice of proposal given by (11). Note that the PMMH sampler is obtained as a special case, by the choice $\epsilon = 0$. Hence, the introduction of the drift results in more parameters for the user to choose. However, guided by (9), we can set the drift coefficient to half the noise variance as a rule-of-thumb. We emphasize that LPMH admits the exact posterior $p(\theta | y_{1:T})$ as stationary distribution despite the fact that the proposal is based on an SMC approximation.

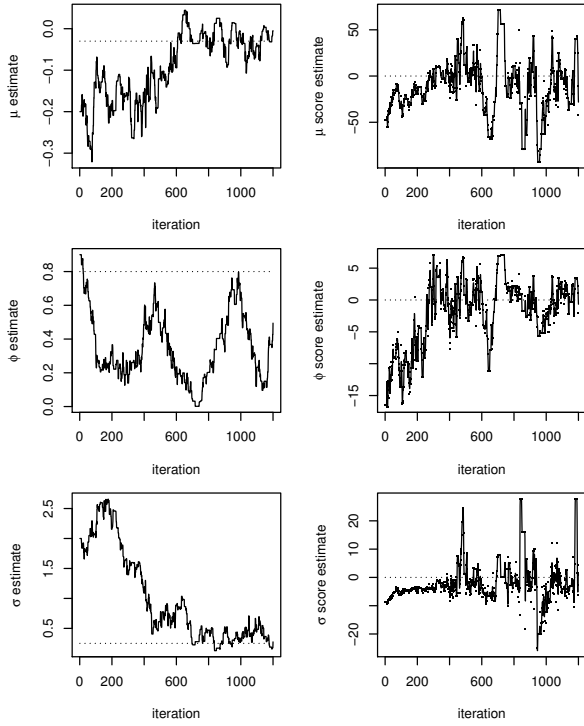


Fig. 2. Left: Trace plots of the three parameter with dotted lines indicating the true parameters. Right: Score estimates at different iterations for the three parameters shown as kernel regression estimates.

4. NUMERICAL ILLUSTRATIONS

We consider a stochastic volatility model to illustrate the behaviour of the score estimate and the proposed LPMH algorithm. The model used is the reparametrized Cox-Ingersoll-Ross model [30] discussed in [31], which is expressed as the following state-space model

$$x_{t+1} = \mu + x_t + \phi \exp(-x_t) + \exp\left(-\frac{x_t}{2}\right) v_t,$$

$$y_t = \sigma \exp\left(\frac{x_t}{2}\right) e_t,$$

where v_t and e_t denote two independent standard normal processes, i.e. $\mathcal{N}(0, 1)$. The problem is to infer the parameters $\theta = \{\mu, \phi, \sigma\}$ given a set of observations $\{y_{1:T}\}$. The covariance matrix of the noise term in (11) was chosen as a diagonal matrix with the following three diagonal elements $\{0.02^2, 0.04^2, 0.08^2\}$. The scaling of the score function was chosen as $\epsilon = \{0.02^2, 0.04^2, 0.08^2\}/2$, which follows from the Euler discretisation as previously discussed.

The system was simulated for $T = 100$ time steps with true parameters $\theta^* = \{-0.03, 0.8, 0.2\}$. In Figure 1, we have estimated the score function for ϕ by fixing $\mu = \mu^*$. The dot indicates the points (σ^*, ϕ^*) and the thick lines indicate the zero score level. The score landscape exhibits an intuitive behaviour, i.e. being negative if the parameters are larger than their true value and positive if smaller.

In Figure 2, we present the trace plots and the contribution of the drift term for the initial 1200 iterations from one run of the LPMH algorithm using $N = 100$ particles. We initialize using $\theta_0 = \{-0.2, 0.9, 2\}$ and the priors $\pi(\mu) = \mathcal{U}[-1, 1]$, $\pi(\phi) = \mathcal{U}[-1, 1]$ and $\pi(\sigma) = \mathcal{U}[0, \infty]$. To visualize the impact of the drift, an esti-

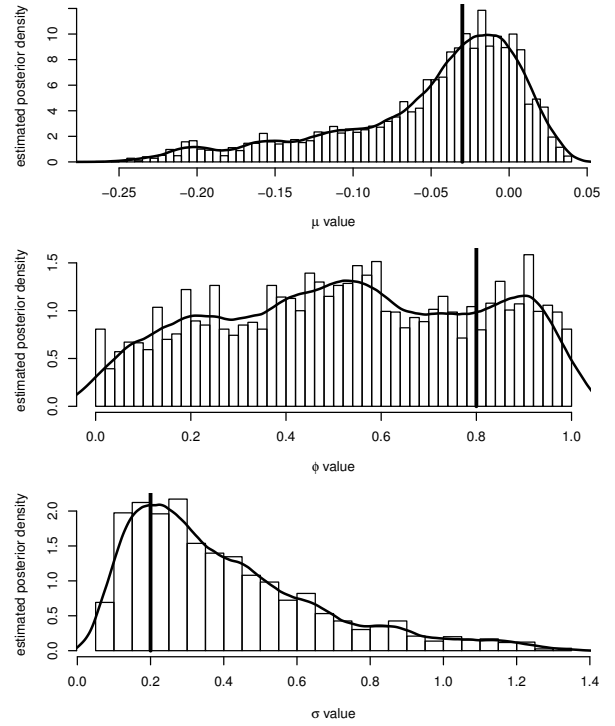


Fig. 3. The posterior density estimates as histograms and kernel density estimates with the vertical lines indicating the true parameter values.

mate of the drift has been created using a non-parametric regression with an Epanechnikov kernel [32].

The impact of the drift is noticeable in the beginning as the drift to a large extent have the same sign, i.e. pulling the parameter towards the true value. Once the Markov chain has reached the true value, the influence is somewhat increased and resembles some additional zero-mean white noise. This suggests that a smaller noise variance can be used compared with the random walk sampler in PMMH.

In Figure 3, we present the resulting posterior density estimates for $M = 10\,000$ iterations (discarding the initial 3000 iterations as burn-in). The histogram is overlaid with a kernel density estimate using an Epanechnikov kernel.

5. CONCLUSION

In this paper, we have explored the potential of using score information to guide the random walks through the parameter space. The LPMH algorithm is based on a generalised version of the PMMH algorithm, called PMH, allowing us for exploiting more of the information available in the generated particle systems. Combining the PMH sampler with forward smoothing for score estimation gives us the LPMH algorithm.

Future work includes applying the LPMH algorithm to high-dimensional problems to investigate if the high-dimensional properties of the LMC sampler carries over to PMCMC. Using the particle system for estimating other quantities than the score is also of interest for designing more efficient proposal kernels. Other smoothers are also of interest for decreasing the complexity of the score estimation, making the algorithms faster and more efficient.

6. 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, no. 3, pp. 269–342, 2010.
- [2] M. A. Beaumont, "Estimation of population growth or decline in genetically monitored populations," *Genetics*, vol. 164, no. 3, pp. 1139–1160, 2003.
- [3] C. Andrieu and G. O. Roberts, "The pseudo-marginal approach for efficient Monte Carlo computations," *The Annals of Statistics*, vol. 37, no. 2, pp. 697–725, 2009.
- [4] N. Whiteley, C. Andrieu, and A. Doucet, "Efficient Bayesian inference for switching state-space models using discrete particle Markov chain Monte Carlo methods," Tech. Rep., Bristol Statistics Research Report 10:04, 2010.
- [5] F. Lindsten and T. B. Schön, "On the use of backward simulation in the particle Gibbs sampler," in *Proceedings of the 37th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Kyoto, Japan, March 2012, pp. 3845–3848.
- [6] F. Lindsten, M. I. Jordan, and T. B. Schön, "Ancestor sampling for particle Gibbs," in *Proceedings of the 2012 Conference on Neural Information Processing Systems (NIPS)*, Lake Tahoe, NV, USA, Dec. 2012.
- [7] 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*, vol. 171, pp. 134–151, 2012.
- [8] C. Andrieu, A. Doucet, and R. Holenstein, "Discussion on Particle Markov chain Monte Carlo methods," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 72, no. 3, pp. 333–339, 2010.
- [9] P. Del Moral, A. Doucet, and S. Singh, "Forward smoothing using sequential Monte Carlo," *Pre-print*, 2010, arXiv:1012.5390v1.
- [10] O. Cappé, E. Moulines, and T. Rydén, *Inference in Hidden Markov Models*, Springer, 2005.
- [11] 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.
- [12] 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*, vol. 98, no. 1, pp. 65–80, 2011.
- [13] G. Poyiadjis, A. Doucet, and S. S. Singh, "Maximum likelihood parameter estimation in general state-space models using particle methods," in *Proceedings of the American Statistical Association*, 2005.
- [14] S. Duane, A. D. Kennedy, B. J. Pendleton, and D. Roweth, "Hybrid Monte Carlo," *Physics letters B*, vol. 195, no. 2, pp. 216–222, 1987.
- [15] M. Girolami and B. Calderhead, "Riemann manifold Langevin and Hamiltonian Monte Carlo methods," *Journal of the Royal Statistical Society: Series B*, vol. 73, no. 2, pp. 1–37, 2011.
- [16] M. C. Higgs, *Approximate inference for state-space models*, Ph.D. thesis, University College London (UCL), 2011.
- [17] H. Ishwaran, "Applications of Hybrid Monte Carlo to Bayesian generalized linear models: Quasicomplete separation and neural networks," *Journal of Computational and Graphical Statistics*, vol. 8, no. 4, pp. 779–799, 1999.
- [18] M. N. Schmidt, "Function factorization using warped Gaussian processes," in *Proceedings of the 26th Annual International Conference on Machine Learning*, Montreal, Canada, June 2009, ACM, pp. 921–928.
- [19] J. S. Liu, *Monte Carlo strategies in scientific computing*, Springer, 2008.
- [20] A. Doucet and A. Johansen, "A tutorial on particle filtering and smoothing: Fifteen years later," in *The Oxford Handbook of Nonlinear Filtering*, D. Crisan and B. Rozovsky, Eds. Oxford University Press, 2011.
- [21] F. Gustafsson, "Particle filter theory and practice with positioning applications," *IEEE Aerospace and Electronic Systems Magazine*, vol. 25, no. 7, pp. 53–82, 2010.
- [22] M. K. Pitt and N. Shephard, "Filtering via simulation: Auxiliary particle filters," *Journal of the American Statistical Association*, vol. 94, no. 446, pp. 590–599, 1999.
- [23] R. M. Neal, "MCMC using Hamiltonian dynamics," in *Handbook of Markov Chain Monte Carlo*, B. Steve, G. Andrew, J. Galin, and M. Xiao-Li, Eds. Chapman & Hall/ CRC Press, June 2010.
- [24] M. Creutz, "Global Monte Carlo algorithms for many-fermion systems," *Physical Review D*, vol. 38, no. 4, pp. 1228–1238, 1988.
- [25] A. D. Kennedy, "The theory of hybrid stochastic algorithms," *Probabilistic Methods in Quantum Field Theory and Quantum Gravity*, vol. B224, pp. 209–223, 1990.
- [26] G. Kitagawa, "Monte Carlo filter and smoother for non-Gaussian nonlinear state space models," *Journal of Computational and Graphical Statistics*, vol. 5, no. 1, pp. 1–25, 1996.
- [27] S. J. Godsill, A. Doucet, and M. West, "Monte Carlo smoothing for nonlinear time series," *Journal of the American Statistical Association*, vol. 99, no. 465, pp. 156–168, Mar. 2004.
- [28] 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.
- [29] R. Douc, A. Garivier, E. Moulines, and J. Olsson, "Sequential Monte Carlo smoothing for general state space hidden Markov models," *Annals of Applied Probability*, vol. 21, no. 6, pp. 2109–2145, 2011.
- [30] J. Hull, *Options, futures, and other derivatives*, Pearson, 7 edition, 2009.
- [31] S. Chib, M. K. Pitt, and N. Shephard, "Likelihood based inference for diffusion driven state space models," Tech. Rep., Nuffield Collage, Oxford, UK, 2006.
- [32] J. Friedman, T. Hastie, and R. Tibshirani, *The elements of statistical learning*, Springer Series in Statistics, 2 edition, 2009.