Discussion Paper Series

RIEB

Kobe University

DP2016-36

A Multiple-Try Extension of the Particle Marginal Metropolis-Hastings (PMMH) Algorithm with an Independent Proposal

Takashi KAMIHIGASHI

Hiroyuki WATANABE

December 7, 2016

Research Institute for Economics and Business Administration

Kobe University

2-1 Rokkodai, Nada, Kobe 657-8501 JAPAN

A Multiple-Try Extension of the Particle Marginal Metropolis-Hastings (PMMH) Algorithm with an Independent Proposal*

Takashi Kamihigashi† Hiroyuki Watanabe‡

December 6, 2016

Abstract

In this paper we propose a multiple-try extension of the PMMH algorithm with an independent proposal. In our algorithm, $I \in \mathbb{N}$ parameter particles are sampled from the independent proposal. For each of them, a particle filter with $K \in \mathbb{N}$ state particles is run. We show that the algorithm has the following properties: (i) the distribution of the Markov chain generated by the algorithm converges to the posterior of interest in total variation; (ii) as I increases to ∞, the acceptance probability at each iteration converges to 1 with probability 1; and (iii) as I increases to ∞, the autocorrelation of any order for any parameter with bounded support converges to 0. These results indicate that the algorithm generates almost i.i.d. samples from the posterior for sufficiently large I. Our numerical experiments suggest that one can visibly improve mixing by increasing I from 1 to a small number. This does not significantly increase computation time if a computer with at least the same number of threads is used.

Keywords: multiple-try method; particle marginal Metropolis-Hastings; Markov chain Monte Carlo; mixing; state space models

*Financial support from the Japan Society for the Promotion of Science (JSPS KAKENHI Grant Number 15H05729) is gratefully acknowledged.

†Research Institute for Economics and Business Administration (RIEB), Kobe University, Rokkodai, Nada, Kobe 657-8501 JAPAN. Email: tkamihig@rieb.kobe-u.ac.jp.

‡Research Institute for Economics and Business Administration (RIEB), Kobe University, Rokkodai, Nada, Kobe 657-8501 JAPAN. Email: hwatanabe@rieb.kobe-u.ac.jp.

1 Introduction

Over the past few decades, Markov chain Monte Carlo (MCMC) algorithms have become an indispensable tool for Bayesian inference. A standard MCMC algorithm such as Metropolis-Hastings (MH) generates a Markov chain that converges in distribution to the posterior of interest as the number of iterations increases to ∞ (e.g., Tierney, 1994; Roberts and Rosenthal, 2004). In this case the posterior can be approximated by the empirical distribution provided that the number of iterations is large enough. In theory this holds quite generally, but in practice the Markov chain generated by an MCMC algorithm often suffers from poor mixing and fails to converge within a reasonable time frame. The problem is particularly severe for state space models, in which a large number of state (or hidden) variables are often treated as unknown parameters.

As far as state space models are concerned, the problem can be mitigated by using the particle marginal Metropolis-Hastings (PMMH) algorithm (Andrieu et al., 2009, 2010), which embeds a particle filter, or a sequential Monte Carlo method (e.g., Doucet et al., 2000), within an MH algorithm. PMMH belongs to the class of MCMC algorithms known as particle MCMC algorithms (Andrieu et al., 2010), which build on earlier work by Beaumont (2003) and Andrieu and Roberts (2009).

In the PMMH algorithm, given a sample of the static parameter vector from a proposal, the marginal likelihood of observed data is sequentially approximated by Monte Carlo integration using (what we call) "state particles." The state of the chain is updated with a probability based on the ratio of marginal likelihood estimates. While PMMH has been applied and extended in a wide variety of contexts (e.g., Yuan et al., 2012; Kokkala and Särkkä, 2015; Golightly et al., 2015; Fearnhead and Meligkotsidou, 2015), it leaves room for improvement in terms of static parameters, which are only updated using a standard MH algorithm.

Earlier efforts to improve the mixing of parameters for static models include the multiple-try Metropolis (MTM) method (Liu et al., 2000). In this method, multiple "parameter particles" are drawn from a proposal. One of them is resampled according to importance weights, and is accepted to update the state of the chain with a probability computed using additional reference samples. This method is closely related to PMCMC algorithms—in particular, the particle MH (PMH) algorithm—as discussed by Martino et al. (2016). Pandolfi et al. (2010) provide a generalization of MTM, and

Bédard et al. (2012) present a scaling analysis of MTM methods. Martino and Louzada (2016) discuss mixing properties of various MTM algorithms.

While MTM and its extensions have been applied to state space models (e.g., So, 2006; Wang, 2009), studies in the literature that discuss both PMMH and MTM are surprisingly limited. Exceptions include Andrieu et al. (2009, p. 57), who differentiate PMMH from MTM, and Storvik (2011), who considers acceptance probabilities for a class of MCMC algorithms including PMMH and MTM. Related studies that do not explicitly mention PMMH include Martino et al. (2016), who examine the relations between PMH and MTM schemes, and Tran et al. (2016), who discuss extensions of their adaptive sampler to MTM and a class of MCMC algorithms including PMCMC.

In this paper we propose a new algorithm for state space models that combines PMMH and MTM under the assumption that the proposal for static parameters is independent. We call this algorithm MTiPMMH (multiple-try independent PMMH) for the reasons specified below. In the MTiPMMH algorithm, a prespecified number of parameter particles are sampled from the independent proposal. For each of them, a particle filter is run with a prespecified number of state particles. Then one of the parameter particles is resampled according to their marginal likelihood estimates. The state of the chain is updated with a probability based on the ratio between the averages of marginal likelihood estimates.

We show that the MTiPMMH algorithm has the following three asymptotic properties: (i) the distribution of the Markov chain generated by the algorithm converges to the posterior of interest in total variation; (ii) as the number of parameter particles increases to ∞, the acceptance probability at each iteration converges to 1 with probability 1; and (iii) as the number of parameter particles increases to ∞, the autocorrelation of any order for any parameter with bounded support converges to 0. These results are valid for any number of state particles, indicating that MTiPMMH generates almost i.i.d. samples from the posterior with a sufficiently large number of parameter particles.

To show these results, we start with a general static model and consider the MH algorithm with an independent proposal, which we call iMH (independent MH). We extend iMH by introducing parameter particles. Martino et al. (2016) argue that this extended algorithm (which they call I-MTM2) is a variant of the MTM method with an independent proposal. Based on their argument, we regard the extended algorithm as a multiple-try (MT) extension of iMH; thus we call it MTiMH (multiple-try independent MH). While Martino et al. (2016) only use MTiMH to compare MTM and PMH schemes without establishing its theoretical properties, we show that MTiMH has the three asymptotic properties listed above.

With these results for MTiMH in hand, we turn to a state space model and consider the PMMH algorithm with an independent proposal, which we call iPMMH (independent PMMH). Following Andrieu et al. (2010) in spirit, we reformulate iPMMH as an iMH algorithm. This allows us to reformulate MTiPMMH as an MTiMH algorithm and to show that the three asymptotic results established for MTiMH hold also for MTiPMMH.

There are at least three advantages to using an independent proposal rather than a more general Markovian proposal. First, a combination of an independent proposal and high acceptance probabilities gurantees good mixing. Note that high acceptance probabilities alone do not imply good mixing. A case in point is Gibbs sampling, which always has an acceptance probability of 1, but often suffers from poor mixing. Second, an independent proposal is easy to construct. For example, one can approximate the empirical distribution of samples from a pilot run by a normal distribution, and use it as an independent proposal. We use this approach in some of our numerical experiments. Third, an independent proposal makes it easy to utilize parallel computation. We present a parallelized version of MTiPMMH for which computational cost does not significantly increase as the number of parameter particles increases, provided that a computer with at least the same number of threads is used.

The rest of the paper is organized as follows. In Section 2 we introduce basic notation for state space models and related algorithms, and briefly discuss the PMMH algorithm. In Section 3 we turn to static models and present the MTiMH algorithm. We establish our core results in this static framework. In Section 4 we present the iPMMH and MTiPMMH algorithms. We reformulate iPMMH as an iMH algorithm, and apply the results shown for MTiMH to MTiPMMH. In Section 5 we conduct numerical experiments by utilizing the simple stochastic volatility model employed by Fearnhead and Meligkotsidou (2015). In Section 6 we conclude the paper with discussion of our results.

2 Preliminaries

Given any (possibly multidimensional) random variables a and b, we let $p(a,b)$ denote the joint density of (a,b), $p(a)$ the marginal density of a, and $p(a|b)$ the conditional density of a given b. These densities can be prior or posterior densities. A proposal density is denoted by q instead of p. Throughout this paper all functions are assumed to Borel-measurable.

2.1 State Space Models

Let $n_\theta, n_x, n_y \in \mathbb{N}$. Let Θ and $\mathcal{X}$ be Borel subsets of $\mathbb{R}^{n_\theta}$ and $\mathbb{R}^{n_x}$, respectively. Consider a state space model consisting of a state (or hidden) process $\{x_t\}_{t=1}^\infty \subset \mathcal{X}$, an observation process $\{y_t\}_{t=1}^\infty \subset \mathbb{R}^{n_y}$, and a vector $\theta \in \Theta$ of unknown static parameters, which are taken as random variables. The conditional distribution of x_{t+1} given $x_{1:t}, y_{1:t}$, and θ is given by $p(x_{t+1}|x_t,\theta)$, where $x_{1:t} = (x_1, x_2, \dots, x_t)$, etc. The conditional distribution of y_t given $x_{1:t}, y_{1:t-1}$, and θ is given by $p(y_t|x_t,\theta)$. The parameter vector θ is distributed according to prior $p(\theta)$. The initial value x_1 is distributed as $p(x_1|\theta)$, which can be part of the model or a prior.

Given observed data $y_{1:T}$ with $T \in \mathbb{N}$, Bayesian inference is typically performed based on the posterior density $p(\theta, x_{1:T}|y_{1:T})$, which satisfies

$$\begin{aligned} p(\theta, x_{1:T}|y_{1:T}) &= p(\theta, x_{1:T}, y_{1:T})/p(y_{1:T}) \propto p(\theta, x_{1:T}, y_{1:T}) && (1)\\ &= p(\theta)p(x_1|\theta)\left[\prod_{t=1}^{T-1} p(x_{t+1}|x_t,\theta)\right]\prod_{t=1}^{T} p(y_t|x_t,\theta). && (2)\end{aligned}$$

2.2 The PMMH Algorithm

We start with some notation to present various algorithms in a concise manner. For $m, n \in \mathbb{N}$ with $m \le n$, we let

$$\begin{aligned} [m:n] &= \{m, \dots, n\}, && (3)\\ \text{“}m:n\text{”} &= \text{“}m, m+1, \dots, n,\text{”} && (4)\\ \text{“}n:m\text{”} &= \text{“}n, n-1, \dots, m.\text{”} && (5)\end{aligned}$$

For $a, b \in \mathbb{R}_+$, we let

$$a \wedge b = \min\{a, b\}. \qquad (6)$$

In what follows, the division operator / precedes the minimum operator $\wedge$ (e.g., see line 18 of Algorithm 1). Given $w^1, \ldots, w^N \geq 0$ with $w_1 + \cdots + w_N > 0$, let $D(w^{1:N})$ denote the discrete distribution on $[1:N]$ with the probability of each $n \in [1:N]$ given by

$$\frac{w^n}{\sum_{m=1}^{N} w^m}. \tag{7}$$

Algorithm 1 shows pseudocode for the PMMH (particle marginal MH) algorithm. In line 2, θ^* is initially set arbitrarily, and is subsequently drawn from conditional proposal $q(\theta|\theta_{j-1})$. In line 4, x_1^k is drawn from $q(x_1|y_1, \theta^*)$; in line 9, x_{t+1}^k is drawn from $q(x_{t+1}|y_{t+1}, x_t^{a_t^k}, \theta^*)$. In line 11, k_T^* is drawn from $D(w_T^{1:K})$. In line 19, "w.p." means "with probability."

Andrieu et al. (2010, Theorem 4) show that Algorithm 1 generates a sequence $\{(\theta_j, \xi_j)\}_{j=1}^{\infty}$ such that the distribution of (θ_j, ξ_j) converges to the posterior $p(\theta, x_{1:T}|y_{1:T})$ in total variation as $j \uparrow \infty$. This result relies on the fact that given θ, the product in line 14 is an unbiased estimator of $p(y_{1:T}|\theta)$:

$$\int u(\theta, x_{1:T}^{1:K}, y_{1:T}) p(x_{1:T}^{1:K}|\theta, y_{1:T}) dx_{1:T}^{1:K} = p(y_{1:T}|\theta), \tag{8}$$

where

$$u(\theta^*, x_{1:T}^{1:K}, y_{1:T}) = \prod_{t=1}^{T} \left(K^{-1} \sum_{k=1}^{K} w_t^k \right), \tag{9}$$

$$x_{1:T}^{1:K} = \{x_t^k\}_{t\in[1:T]}^{k\in[1:K]}. \tag{10}$$

See Pit et al. (2012, Theorem 1) and Del Moral (2004, Proposition 7.4.1) for proofs of (8).

3 A Multiple-Try Extension of iMH

In this section we consider general static models and study an extension of the MH algorithm with an independent proposal. The results shown here are applied to state space models in Section 4.

The basis of our analysis is the MH algorithm with an independent proposal $q(\theta)$, which we call iMH. Algorithm 2 shows this algorithm as a function of (q, g), where q and g are functions satisfying the following assumption, which is maintained throughout this section.

Algorithm 1: PMMH (Particle Marginal Metropolis-Hastings)

for $j = 1 : J$

  $\theta^* \begin{cases} \in \Theta \quad \text{arbitrary} & \text{if } j = 1 \\ \sim q(\theta|\theta_{j-1}) & \text{if } j \geq 2 \end{cases}$

  for $k = 1 : K$

    $x_1^k \sim q(x_1|y_1, \theta^*)$

    $w_1^k = p(y_1|x_1^k, \theta^*)p(x_1^k|\theta^*)/q(x_1^k|y_1, \theta^*)$

  for $t = 1 : T - 1$

    for $k = 1 : K$

      $a_t^k \sim D(w_t^{1:K})$

      $x_{t+1}^k \sim q(x_{t+1}|y_{t+1}, x_t^{a_t^k}, \theta^*)$

      $w_{t+1}^k = p(y_{t+1}|x_{t+1}^k, \theta^*)p(x_{t+1}|x_t^{a_t^k}, \theta^*)/q(x_{t+1}|y_{t+1}, x_t^{a_t^k}, \theta^*)$

  $k_T^* \sim D(w_T^{1:K}), x_T^* = x_T^{k_T^*}$

  for $t = (T - 1) : 1$

    $k_t^* = a_t^{k_{t+1}^*}, x_t^* = x_t^{k_t^*}$

  $h^* = p(\theta^*) \prod_{t=1}^{T} \left(K^{-1} \sum_{k=1}^{K} w_t^k \right)$

  if $j = 1$ **then**

    $(\theta_1, \xi_1, h_1) = (\theta^*, x_{1:T}^*, h^*)$

  else

    $r = 1 \wedge [h^*/q(\theta^*|\theta_{j-1})]/[h_{j-1}/q(\theta_{j-1}|\theta^*)]$

    $(\theta_j, \xi_j, h_i) = \begin{cases} (\theta^*, x_{1:T}^*, h^*) & \text{w.p. } r \\ (\theta_{j-1}, \xi_{j-1}, h_{j-1}) & \text{w.p. } 1 - r \end{cases}$

output: $\theta_{1:J}, \xi_{1:J}$

Algorithm 2: iMH(q, g) (Independent Metropolis-Hastings)

for $j = 1 : J$

  $\theta^* \sim q(\theta), w^* = g(\theta^*)/q(\theta^*)$

  if $j = 1$ **then**

    $(\theta_1, w_1) = (\theta^*, w^*)$

  else

    $r = 1 \wedge w^*/w_{j-1}$

    $(\theta_j, w_j) = \begin{cases} (\theta^*, w^*) & \text{w.p. } r \\ (\theta_{j-1}, w_{j-1}) & \text{w.p. } 1 - r \end{cases}$

output: $\theta_{1:J}$

Assumption 3.1. The Lebesgue measure of $\Theta \subset \mathbb{R}^{n_\theta}$ is strictly positive, $q : \Theta \to \mathbb{R}_{++}$ is a probability density function on Θ, and $g : \Theta \to \mathbb{R}_{++}$ is an integrable Borel function.

In what follows, when we refer to an algorithm like Algorithm 2, we explicitly mention arguments like q and g only if necessary. The following result is well known, but we state it for reference purposes.

Proposition 3.1. *The Markov chain $\{\theta_j\}_{j=1}^{\infty}$ generated by Algorithm 2 (with $J = \infty$) has a unique stationary distribution, which has a density representation $\pi(\theta)$ satisfying*

$$\pi(\theta) \propto g(\theta). \tag{11}$$

Furthermore, the distribution of θ_j converges to $\pi(\theta)$ in total variation as $j \uparrow \infty$.

Proof. This result follows from Theorem 1 and Corollary 2 in Tierney (1994) under Assumption 3.1. □

Let us now consider an extension of iMH (Alg. 2). Algorithm 3 (Martino et al., 2016) extends iMH with I "parameter particles," and one of the parameter particles is resampled according to the weights w^i in line 3. The average of these weights is used to compute the acceptance probability.

Martino et al. (2016) argue that Algorithm 3 is a variant of the MTM method with an independent proposal. Based on their argument, we regard

Algorithm 3: MTiMH(q, g) (Multiple-Try iMH)

1 **for** $j = 1 : J$
2   **for** $i = 1 : I$
3         $\theta^i \sim q(\theta), w^i = g(\theta^i)/q(\theta^i)$
4     $i^* \sim D(w^{1:I}), \theta^* = \theta^{i^*}$
5     $w^* = I^{-1} \sum_{i=1}^{I} w^i$
6   run lines 3–7 of iMH(q, g) (Alg. 2)

output: $\theta_{1:J}$

Algorithm 3 as a multiple-try extension of iMH (Alg. 2), and call it MTiMH (multiple-try iMH). It turns out that MTiMH (Alg. 3) is a special case of iMH (Alg. 2) with an extended parameter space. This observation is useful in showing theoretical properties of MTiMH; a similar approach is used by Andrieu et al. (2010) to establish various properties of PMMH (Alg. 1) by expressing it as an MH algorithm with an extended space.

To reformulate MTiMH (Alg 3) as an iMH (Alg. 2) algorithm, let $w^0 = 0$. Consider $w^{1:I}$ given in line 3; for $i \in [1 : I]$, define

$$U_i = \left(\frac{\sum_{s=0}^{i-1} w^s}{\sum_{s=0}^{I} w^s}, \frac{\sum_{s=0}^{i} w^s}{\sum_{s=0}^{I} w^s} \right]. \tag{12}$$

In line 4 of MTiMH (Alg. 3), replace "$i^* \sim D(w^{1:I})$" by

$$u \sim U(0,1), \quad i^* = \sum_{i=1}^{I} 1\{u \in U_i\} i, \tag{13}$$

where $U(0, 1)$ denotes the uniform distribution on $(0, 1)$, and $1\{\cdot\}$ is the indicator function. Note that (13) implies that $i^* \sim D(w^{1:I})$; thus the distributional properties of MTiMH (Alg. 3) remain unchanged. We also define

$$\tilde{\theta} = (\theta^{1:I}, u), \tag{14}$$

$$\tilde{\Theta} = \Theta^I \times (0, 1), \tag{15}$$

$$\tilde{q}(\tilde{\theta}) = \prod_{i=1}^{I} q(\theta^i), \tag{16}$$

$$\tilde{g}(\tilde{\theta}) = \tilde{q}(\tilde{\theta}) w^*, \tag{17}$$

where w^* is given in line 5 of MTiMH (Alg. 3). Consider iMH($\tilde{q}, \tilde{g}$), which is Algorithm 2 with $\tilde{q}$ and $\tilde{g}$ replacing q and g. Noticing that $\tilde{g}(\tilde{\theta})/\tilde{q}(\tilde{\theta}) = w^*$, it is easy to see that MTiMH(q, g) is equivalent to iMH($\tilde{q}, \tilde{g}$) with i^* determined by (13).

It follows from Proposition 3.1 that the stationary density $\tilde{\pi}(\tilde{\theta})$ of the Markov chain $\{\tilde{\theta}_j\}_{j=1}^{\infty}$ generated by iMH($\tilde{q}, \tilde{g}$) satisfies

$$\tilde{\pi}(\tilde{\theta}) \propto \tilde{g}(\tilde{\theta}), \tag{18}$$

and that the distribution of $\tilde{\theta}_j$ converges to $\tilde{\pi}(\tilde{\theta})$ in total variation. The following result is shown by verifying that the marginal distribution of θ^* (in line 4 of Alg. 3) with respect to $\tilde{\pi}(\tilde{\theta})$ is the distribution $\pi(\theta)$ in (11).

Theorem 3.1. *Algorithm 3 with $J = \infty$ generates a sequence $\{\theta_j\}_{j=1}^{\infty}$ such that the distribution of θ_j converges to $\pi(\theta)$ in total variation as $j \uparrow \infty$.*

Proof. See Appendix A. □

Our next result shows that as the number I of parameter particles increases to ∞, the acceptance probability for each $j \in [2 : J]$ converges to 1 *a.s.* (almost surely). The result uses Algorithm 4, which is a distributionally equivalent version of Algorithm 3 in which the random seed is reset in a specific way and the dependence of all variables on I and j is made explicit.

Theorem 3.2. *Consider Algorithm 4. For each $j \in [2 : J]$, we have $r_{I,j} \to 1$ a.s. as $I \uparrow \infty$, where $r_{I,j}$ is computed in line 13 of Algorithm 4.*

Proof. See Appendix B. □

Recall from Subsection 2.1 that $\theta \in \mathbb{R}^{n_\theta}$. Let $\theta(n)$ denote the nth component of θ for $n \in [1 : n_\theta]$; i.e., $\theta = (\theta(1), \dots, \theta(n_\theta))$. Let $Cor(\theta_{I,j-s}(n), \theta_{I,j}(n'))$ denote the correlation between $\theta_{I,j-s}(n)$ and $\theta_{I,j}(n')$ for $I \in \mathbb{N}, j \in [1 : J], s \in [2 : j-1]$, and $n, n' \in [1 : n_\theta]$. The following result shows that this correlation converges to 0 as $I \uparrow \infty$ provided that both $\theta_{I,j-s}(n)$ and $\theta_{I,j}(n')$ have bounded support.

Theorem 3.3. *Consider Algorithm 4. Suppose either that Θ is a bounded set (in which case we set $n_1 = n_\theta$), or that there exist sets $\Theta_1 \subset \mathbb{R}^{n_1}$ and $\Theta_2 \subset \mathbb{R}^{n_2}$ with $n_1, n_2 \in \mathbb{N}$ and $\Theta = \Theta_1 \times \Theta_2$ such that Θ_1 is a bounded set. Then for any $j \in [2 : J], s \in [1 : j-1]$, and $n, n' \in [1 : n_1]$, we have*

$$\lim_{I \uparrow \infty} Cor(\theta_{I,j-s}(n), \theta_{I,j}(n')) = 0. \tag{19}$$

Algorithm 4: MTiMH(I) (MTiMH for Varying I)

reset the random seed to any constant independent of I
for $j = 1 : J$
    $u_j \sim U(0,1)$
for $j = 1 : J$
    reset the random seed to u_j
    for $i = 1 : I$
        $\theta^i_{I,j} \sim q(\theta), w^i_{I,j} = g(\theta^i_{I,j})/q(\theta^i_{I,j})$
    $i^* \sim D(w^{1:I}_{I,j}), \theta^*_{I,j} = \theta^{i^*}_{I,j}$
    $w^*_{I,j} = I^{-1} \sum_{i=1}^{I} w^i_{I,j}$
    if $j = 1$ **then**
        $(\theta_{I,1}, w_{I,1}) = (\theta^*_{I,1}, w^*_{I,1})$
    else
        $r_{I,j} = 1 \wedge w^*_{I,j}/w_{I,j-1}$
        $(\theta_{I,j}, w_{I,j}) = \begin{cases} (\theta^*_{I,j}, w^*_{I,j}) & \text{w.p. } r_{I,j} \\ (\theta_{I,j-1}, w_{I,j-1}) & \text{w.p. } 1 - r_{I,j} \end{cases}$
output: $\theta_{I,1:J}$

Algorithm 5: iPMMH(q) (Independent PMMH)

---

for $j = 1 : J$
- $\theta^* \sim q(\theta)$
- run lines 3–14 of PMMH (Alg. 1)
- $w^* = h^*/q(\theta^*)$
- **if** $j = 1$ **then**
  - $(\theta_1, \xi_1, w_1) = (\theta^*, x^*_{1:T}, w^*)$
- **else**
  - $r = 1 \wedge w^*/w_{j-1}$
  - $(\theta_j, \xi_j, w_j) = \begin{cases} (\theta^*, x^*_{1:T}, w^*) & \text{w.p. } r \\ (\theta_{j-1}, \xi_{j-1}, w_{j-1}) & \text{w.p. } 1-r \end{cases}$

output: $\theta_{1:J}, \xi_{1:J}$

Proof. See Appendix C. □

This result shows that as far as bounded parameter components are concerned, autocorrelation of any finite order converges to 0 as $I \uparrow \infty$. As for unbounded components (in Θ_2), one can, for example, apply a monotone transformation to make them bounded and apply Theorem 3.3 to the transformed components.

4 A Multiple-Try Extension of iPMMH

The results shown in Section 3 apply to an MT extension of any scheme that can be reformulated as an iMH algorithm (Alg. 2). In this section we return to the framework of state space models discussed in Subsection 2.1 and extend a special case of the PMMH algorithm (Alg. 1). In particular we consider the PMMH algorithm with an independent proposal, which is given in Algorithm 5 as a function of the proposal q. The idea here is to express this algorithm as an iMH algorithm, and to apply Theorems 3.1–3.3 to its MT extension.

Throughout this section we assume the following in addition to the setup specified in Subsection 2.1.

Assumption 4.1. For all $\theta \in \Theta$, we have $p(\theta), q(\theta) \in (0, \infty)$. For all $\theta \in \Theta$ and $x_1 \in \mathcal{X}$, we have $p(x_1|\theta), q(x_1|\theta) \in (0, \infty)$. For all $\theta \in \Theta, t \in [1 : T-1]$,

Algorithm 6: MTiPMMH(q) (Multiple-Try iPMMH)

for $j = 1 : J$
  for $i = 1 : I$
    $\theta^i \sim q(\theta)$
    run lines 3–14 of PMMH (Alg. 1) with $\theta^* = \theta^i$
    $w^i = h^*/q(\theta^i), x^i_{1:T} = x^*_{1:T}$
  $i^* \sim D(w^{1:I}), \theta^* = \theta^{i^*}, x^*_{1:T} = x^{i^*}_{1:T}$
  $w^* = I^{-1} \sum_{i=1}^{I} w^i$
  run lines 5–9 of iPMMH (Alg. 5)
output: $\theta_{1:J}, \xi_{1:J}$

and $x_t, x_{t+1} \in \mathcal{X}$, we have $p(x_{t+1}|x_t, \theta) \in (0, \infty)$. For all $\theta \in \Theta, t \in [1 : T]$, and $x_t \in \mathcal{X}$, we have $p(y_t|x_t, \theta) \in (0, \infty)$. Finally $p(y_{1:T}) \in (0, \infty)$.

Let us now reformulate iPMMH(q) as an iMH (Alg. 2) algorithm. First we rearrange state particles x^k_t and relabeling them as z^k_t as in Figure 1. To be more precise, for $k, k^* \in [1 : K]$, define

$$\mu(k, k^*) = \begin{cases} k^* & \text{if } k = 1, \\ k - 1 & \text{if } 2 \leq k \leq k^*, \\ k & \text{if } k > k^*. \end{cases} \tag{20}$$

For $t \in [1 : T]$ and $k \in [1 : K]$, define

$$z^k_t = x_t^{\mu(k, k^*_t)}. \tag{21}$$

Note that $z^1_t = x^{k^*_t}_t$, that $z^k_t = x^{k-1}_t$ if $k \in [2 : k^*_t]$, and that $z^k_t = x^k_t$ if $k > k^*_t$; see Figure 1 again. We define

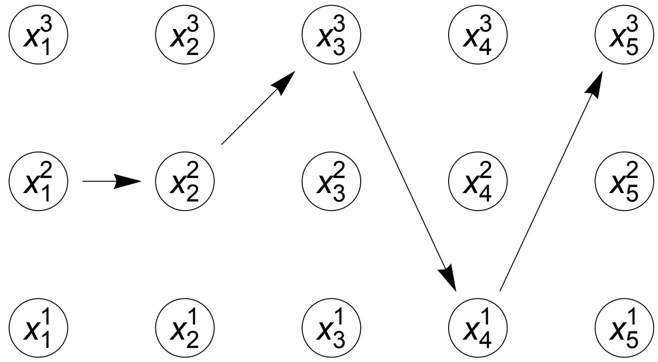
$$\tilde{\theta} = (\theta, z^{1:K}_{1:T}), \tag{22}$$

$$\tilde{\Theta} = \Theta \times (\mathcal{X}^K)^T, \tag{23}$$

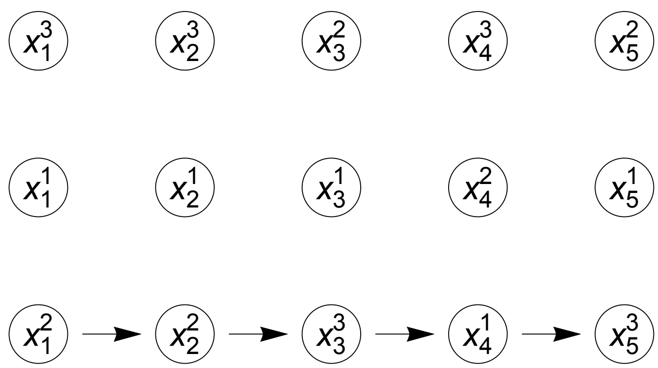
$$\tilde{q}(\tilde{\theta}) = q(\theta)\tilde{p}(z^{1:K}_{1:T}|\theta, y_{1:T}), \tag{24}$$

$$\tilde{g}(\tilde{\theta}) = p(\theta)u(\theta, z^{1:K}_{1:T}, y_{1:T})\tilde{p}(z^{1:K}_{1:T}|\theta, y_{1:T}), \tag{25}$$

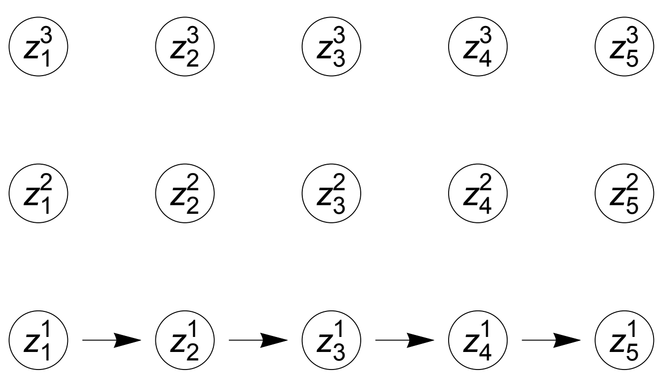
where $\tilde{p}(z^{1:K}_{1:T}|\theta, y_{1:T})$ is the conditional density of $z^{1:K}_{1:T}$ given θ and $y_{1:T}$, and $u(\cdot, \cdot, \cdot)$ is defined by (9).



(a) State particles x_t^k with $k_{1:5}^* = \{2, 2, 3, 1, 3\}$



(b) State particles x_t^k rearranged with $x_{1:t}^*$ at the bottom



(c) State particles relabeled as $z_t^k = x_t^{\mu(k,k_t^*)}$

Figure 1: Transformation from $x_{1:T}^{1:K}$ to $z_{1:T}^{1:K}$ through (21)

Consider iMH($\tilde{q}, \tilde{g}$), which is Algorithm 2 with $q = \tilde{q}$ and $g = \tilde{g}$. It is easy to see that $\tilde{q}(\tilde{\theta}) \in (0, \infty)$ for all $\tilde{\theta} \in \tilde{\Theta}$ by Assumption 4.1. Note that given $\theta \in \Theta$ we have

$$\int u(\theta, z_{1:T}^{1:K}, y_{1:T})\tilde{p}(z_{1:T}^{1:K}|\theta, y_{1:T})dz_{1:T}^{1:K} \tag{26}$$
$$= E[u(\theta, z_{1:T}^{1:K}, y_{1:T})|\theta, y_{1:T}] \tag{27}$$
$$= E[u(\theta, x_{1:T}^{1:K}, y_{1:T})|\theta, y_{1:T}] \tag{28}$$
$$= \int u(\theta, x_{1:T}^{1:K}, y_{1:T})p(x_{1:T}^{1:K}|\theta, y_{1:T})dx_{1:T}^{1:K} \tag{29}$$
$$= p(y_{1:T}|\theta), \tag{30}$$

where $E[a|b]$ is the expectation of a given b, (28) holds since $u(\theta, z_{1:T}^{1:K}, y_{1:T}) = u(\theta, x_{1:T}^{1:K}, y_{1:T})$; and (30) uses (8). Note from (25)–(30) that

$$\int \tilde{g}(\tilde{\theta})d\tilde{\theta} = \int p(\theta)p(y_{1:T}|\theta)d\theta = p(y_{1:T}) < \infty, \tag{31}$$

where the last inequality holds by Assumption 4.1. It follows that iMH($\tilde{q}, \tilde{g}$) satisfies Assumption 3.1.

Note from (24) and (25) that

$$\frac{\tilde{g}(\theta^*, z_{1:T}^{1:K})}{\tilde{q}(\theta^*, z_{1:T}^{1:K})} = \frac{p(\theta^*)u(\theta^*, z_{1:T}^{1:K}, y_{1:T})}{q(\theta^*)} = \frac{h^*}{q(\theta^*)} = w^*, \tag{32}$$

where h^* is given by line 14 of PMMH (Alg. 1), and w^* is given by line 4 of iPMMH(q) (Alg. 5). It follows that the sequences of θ and $z_{1:T}^1$ (as components of $\tilde{\theta}$) generated by iMH($\tilde{q}, \tilde{g}$) are identical to those of θ and $x_{1:T}^*$ generated by iPMMH(q) (Alg. 5).

Note from Proposition 3.1 that iMH($\tilde{q}, \tilde{g}$) generates a sequence $\{\tilde{\theta}_j\}_{j=1}^{\infty}$ such that the distribution of $\tilde{\theta}_j$ converges in total variation to a unique stationary distribution $\tilde{\pi}(\tilde{\theta})$, which satisfies $\tilde{\pi}(\tilde{\theta}) \propto \tilde{g}(\tilde{\theta})$. Theorem 4 in Andrieu et al. (2010) implies that the marginal distribution of $(\theta, z_{1:T}^1) = (\theta, x_{1:T}^*)$ with respect to $\tilde{\pi}(\tilde{\theta})$ is the desired posterior $p(\theta, x_{1:T}|y_{1:T})$. By Theorem 3.1, the stationary distribution and convergence properties of iMH($\tilde{q}, \tilde{g}$) are inherited by MTiMH($\tilde{q}, \tilde{g}$) (recall Alg. 3).

Now consider Algorithm 6, which is a multiple-try extension of iPMMH(q). We call it MTiPMMH (multiple-try PMMH). In this algorithm, I parameter particles are sampled, and for each of them, K state particles are sampled for

Algorithm 7: MTiPMMH(I) (MTiPMMH for Varying I)

1 reset the random seed to any constant independent of I
2 **for** $j = 1 : J$
3     $u_j \sim U(0,1)$
4 **for** $j = 1 : J$
5   reset the random seed to u_j
6   **for** $i = 1 : I$
7         $\theta^i \sim q(\theta)$
8   run lines 3–14 of PMMH (Alg. 1) with $\theta^* = \theta^i$
9         $w^i_{I,j} = h^*/q(\theta^i), x^i_{1:T} = x^*_{1:T}$
10     $i^* \sim D(w^{1:I}_{I,j}), \theta^*_{I,j} = \theta^{i^*}, \xi^*_{I,j} = x^{i^*}_{1:T}$
11     $w^*_{I,j} = I^{-1} \sum_{i=1}^{I} w^i_{I,j}$
12   **if** $j = 1$ **then**
13         $(\theta_{I,1}, \xi_{I,1}, w_{I,1}) = (\theta^*, \xi^*_{I,j}, w^*_{I,j})$
14   **else**
15         $r_{I,j} = 1 \wedge w^*_{I,j}/w_{I,j-1}$
16         $(\theta_{I,j}, \xi_{I,j}, w_{I,j}) = \begin{cases} (\theta^*_{I,j}, \xi^*_{I,j}, w^*_{I,j}) & \text{w.p. } r_{I,j} \\ (\theta_{I,j-1}, \xi_{I,j-1}, w_{I,j-1}) & \text{w.p. } 1 - r_{I,j} \end{cases}$

output: $\theta_{I,1:J}, \xi_{I,1:J}$

each $t \in [1 : T]$. In line 6, one parameter particle is resampled according to the weights w^i given in line 5. Note that the structure of MTiPMMH is essentially the same as that of MTiMH (Alg. 3). Indeed, since iPMMH(q) can be expressed as iMH($\tilde{q}, \tilde{g}$) as discussed above, it follows that MTiPMMH(q) can be expressed as MTiMH($\tilde{q}, \tilde{g}$). Thus we obtain the following results by applying Theorems 3.1–3.3 to MTiMH($\tilde{q}, \tilde{g}$).

Theorem 4.1. *Algorithm 6 (with $J = \infty$) generates a sequence $\{(\theta_j, \xi_j)\}_{j=1}^{\infty}$ such that the distribution of (θ_j, ξ_j) converges to $p(\theta, x_{1,T}|y_{1:T})$ in total variation.*

Theorem 4.2. *Consider Algorithm 7. For each $j \in [2 : J]$, we have $r_{I,j} \to 1$ a.s. as $I \uparrow \infty$, where $r_{I,j}$ is computed in line 15 of Algorithm 7.*

Theorem 4.3. *Consider Algorithm 7. Suppose either that* Θ *is a bounded set (in which case we set* $n_1 = n_\theta$*), or that there exist sets* $\Theta_1 \subset \mathbb{R}^{n_1}$ *and* $\Theta_2 \subset \mathbb{R}^{n_2}$ *with* $n_1, n_2 \in \mathbb{N}$ *and* $\Theta = \Theta_1 \times \Theta_2$ *such that* Θ_1 *is a bounded set. Then for any* $j \in [2 : J], s \in [1 : j-1]$*, and* $n, n' \in [1 : n_1]$*, we have*

$$\lim_{I \uparrow \infty} Cor(\theta_{I,j-s}(n), \theta_{I,j}(n')) = 0. \tag{33}$$

Theorem 4.1 can be shown more directly by following Andrieu et al.'s (2010, Theorem 4) proof or Wilkinson's (2012, pp. 305–307) argument. It is easy to see that Algorithm 7 is distributionally equivalent to Algorithm 6; recall Algorithm 4. With additional notation, Theorem 4.3 can easily be extended to $\Theta \times \mathcal{X}^T$ by replacing Θ with $\Theta \times \mathcal{X}^T$. Overall Theorems 4.1–4.3 indicate that as the number I of parameter particles increases to ∞, one can obtain almost i.i.d. samples from the posterior $p(\theta, x_{1:T}|y_{1:T})$. This is true regardless of the proposal and the number K of state particles.

Before we conclude this section, let us briefly discuss how MTiPMMH (Alg. 6) can be parallelized. Note that lines 3–5 can easily be processed in parallel for all $i \in [1 : I]$, but such a parallel scheme is rather inefficient since for each $j \in [1 : J]$, before line 6 can be executed, lines 3–5 must be completed for all $i \in [1 : I]$. A more efficient but memory-intensive parallel scheme is offered in Algorithm 8. In this algorithm, lines 2–5 can be processed in parallel for all $i \in [1 : I]$, and the resampling and updating steps in lines 7–9 are executed using the data generated in lines 1–5. Depending on the computing environment, one can parallelize lines 1–5 even more efficiently since in effect it suffices to draw $I \times J$ i.i.d. samples from $q(\theta)$ and run lines 4–5 for each sample.

5 Numerical Experiments

5.1 A Simple Stochastic Volatility Model

To illustrate our theoretical results, we consider the simple stochastic volatility model used by Fearnhead and Meligkotsidou (2015):

$$x_t = \gamma x_{t-1} + \sigma_x \eta_t, \quad \eta_t \sim N(0, 1), \tag{34}$$

$$y_t = \sigma_y \exp(x_t) \epsilon_t, \quad \epsilon_t \sim N(0, 1), \tag{35}$$

where $N(\mu, \sigma^2)$ denotes the normal distribution with mean μ and variance σ^2. The initial value x_1 is drawn from $N(0, 1)$.

Algorithm 8: MTiPMMH for Parallelism

for $i = 1 : I$
  for $j = 1 : J$
    $\theta_j^i \sim q(\theta)$
    run lines 3–14 of PMMH (Algorithm 1) with $\theta^* = \theta_j^i$
    $w_j^i = h^*/q(\theta_j^i), x_{1:T}^{i,j} = x_{1:T}^*$
for $j = 1 : J$
  $i^* \sim D(w_j^{1:I}), \theta^* = \theta_j^{i^*}, x_{1:T}^* = x_{1:T}^{i^*,j}$
  $w^* = I^{-1} \sum_{i=1}^{I} w_j^i$
  run lines 5–9 of iPMMH (Alg. 5)
output: $\theta_{1:J}, \xi_{1:J}$

Though it is more common in the literature to divide x_t by 2 in (35) (e.g., Shephard and Pitt, 2007), we simply use Fearnhead and Meligkotsidou's (2015) setting, where all parameter values including priors are fully specified (thus we do not need to choose them arbitrarily). Following Fearnhead and Meligkotsidou (2015, Section 4.1.2), we fix $T = 1000$, and assume that the priors for γ, σ_x^2, and σ_y^2 are independent and given by

$$\gamma \sim N(0.9, 0.1)|_{(-1,1)}, \quad 1/\sigma_x^2 \sim \Gamma(1, 1/100), \quad 1/\sigma_y^2 \sim \Gamma(1, 1), \tag{36}$$

where the distribution for γ is truncated to $(-1, 1)$, and $\Gamma(a, b)$ denotes the gamma distribution with shape parameter a and scale parameter b.

Given simulated data $y_{1:T}$ (to be specified below), we consider two different proposals on $\theta = (\gamma, \sigma_x^2, \sigma_y^2)$. The first proposal is simply the prior $p(\theta)$ given by (36). We construct the second proposal $\hat{q}(\theta)$ using samples from a pilot run of a standard MCMC algorithm; see Appendix D for details.

For each of the two proposals, we consider two values, 4400 and 110000, for J, where J is the number of iterations. If $J = 4400$, then we discard the first 400 iterations as burn-in. If $J = 110000$, we discard the first 10000 iterations as burn-in. Note however that our theoretical results require no burn-in. In total we consider the four pairs of $q(\theta)$ and J given by

$$q(\theta) \in \{p(\theta), \hat{q}(\theta)\}, \quad J \in \{4400, 110000\}. \tag{37}$$

In each case, we increase I from 1 to $\overline{I}$ one by one, where $\overline{I} = 500$ if $J = 4400$

and $\overline{I} = 10$ if $J = 110000$. We fix $K = 500$ throughout, where K is the number of state particles; recall PMMH (Alg. 1).

Since it is costly to literally implement Algorithm 8 for many different values of I, we modify it to reuse as much data as possible. The modified version is shown in Algorithm 9. In lines 1–2 of this algorithm, basic data $\theta_{1:J}^{1:\overline{I}}, w_{1:J}^{1:\overline{I}}$, and $x_{1:T}^{1:\overline{I},1:J}$ for the largest possible value of I are generated and stored. Using the stored data, the steps for resampling i^* and updating θ_j summarized in line 4 are executed for each $I = 1, \dots, \overline{I}$. This way we generate basic data only once, saving a significant amount of time. In addition, this algorithm is consistent with Theorems 4.2 and 4.3; in fact these results directly apply to Algorithm 9 with $\overline{I} = \infty$.

We define

$$\mu = \ln \sigma_y^2, \quad \beta_x = 1/\sigma_x^2. \tag{38}$$

Following Fearnhead and Meligkotsidou (2015), we present our simulation results in terms of μ, β_x, and γ.

In implementing Algorithm 9 (or any of our MTiPMMH algorithms), it is useful to introduce an additional parameter, say M, to line 14 of PMMH (Alg. 1) as follows:

$$h^* = p(\theta^*) \prod_{t=1}^{T} \left(M K^{-1} \sum_{k=1}^{K} w_t^k \right). \tag{39}$$

Theoretically the new parameter M cancels out in line 8 of iPMMH(q) (Alg. 5) as long as $M > 0$. In practice, however, it often enables us to avoid "0/0" by choosing an appropriate value for M. Note that the right-hand side of (39) involves the product of T expressions. Since $T = 1000$ here, if $M = 1$, the product can easily evaluate to 0 numerically even if it is strictly positive theoretically. This problem can mostly be avoided if $I = 1$, in which case most computations can be done in terms of logarithms. This is not the case here since the sum of I terms in line 8 of Algorithm 8 must be evaluated. Accordingly, we replace line 14 of PMMH (Alg. 1) with (39). We set $M = 5$, which seems to work well in the current setting (including (40) below).

5.2 Simulation Results

We simulated data $x_{1:T}$ and $y_{1:T}$ with the following parameter values:

$$\gamma = 0.99, \quad \sigma_x^2 = (1 - \gamma^2), \quad \sigma_y^2 = 1. \tag{40}$$

The above values were borrowed from Fearnhead and Meligkotsidou (2015, Section 4.1.2). The simulated data are common to all the results presented below.

Figures 2–5 illustrate how the performance of Algorithm 8 depends on I by implementing Algorithm 9 for the four different cases given by (37). Since Algorithm 9 heavily uses lines from other algorithms, we use Algorithm 7 instead to explain these figures. In each of Figures 2–5, the top panel plots the average of acceptance probabilities $r_{I,j}$ in line 15 of Algorithm 7 over $j \in [1 : J]$. The second panel plots the realized overall acceptance rate in each run, i.e., the number of acceptance (or updating) events in line 16 of Algorithm 7 divided by J. The remaining three panels plot the autocorrelation times for μ, β_x, and γ, respectively.

The data for Figure 2 were generated by setting $q(\theta) = p(\theta)$ and $J = 4400$ and by varying I from 1 to 500 one by one. The top panel shows that the average acceptance probability tends to increase as I increases. This observation is consistent with Theorem 4.2, which implies that the average acceptance probability converges *a.s.* to 1 as I increases to ∞. The second panel shows that the realized overall acceptance rate in each run is essentially identical to the corresponding average acceptance probability. The remaining three panels show that the autocorrelation times for μ, β_x, and γ quickly decline as I increases from 1, and have a tendency to decrease as I further increases. This observation is consistent with Theorem 4.3, which shows that the autocorrelation of any order for each parameter with bounded support converges to 0 as I increases to ∞. Although μ and β_x do not have bounded support, their autocorrelation times also tend to decrease as I increases.

The data for Figure 3 were generated by setting $q(\theta) = \hat{q}(\theta)$ and $J = 4400$ and by varying I from 1 to 500. Since, as mentioned above, $\hat{q}(\theta)$ was constructed using samples from a pilot run of a standard MCMC sampler, $\hat{q}(\theta)$ is expected to be substantially closer to the posterior $p(\theta|y_{1:T})$ than the prior $p(\theta)$ is. One can see this effect in the top panel in Figure 3, which shows that the average acceptance probability quickly rises as I increases from 1, and is considerably higher for any I than the corresponding value in Figure 2. As in Figure 2, the bottom three panels show that the autocorrelation times for μ, β_x, and γ quickly decline as I increases from 1.

In Figures 2 and 3, due to a memory constraint, we had to choose a rather small value for J in order to allow I to increase up to 500. In Figures 4 and 5, we chose a larger value for J by restricting the largest value for I to 10. These figures exhibit patterns similar to those observed in Figures 2 and 3

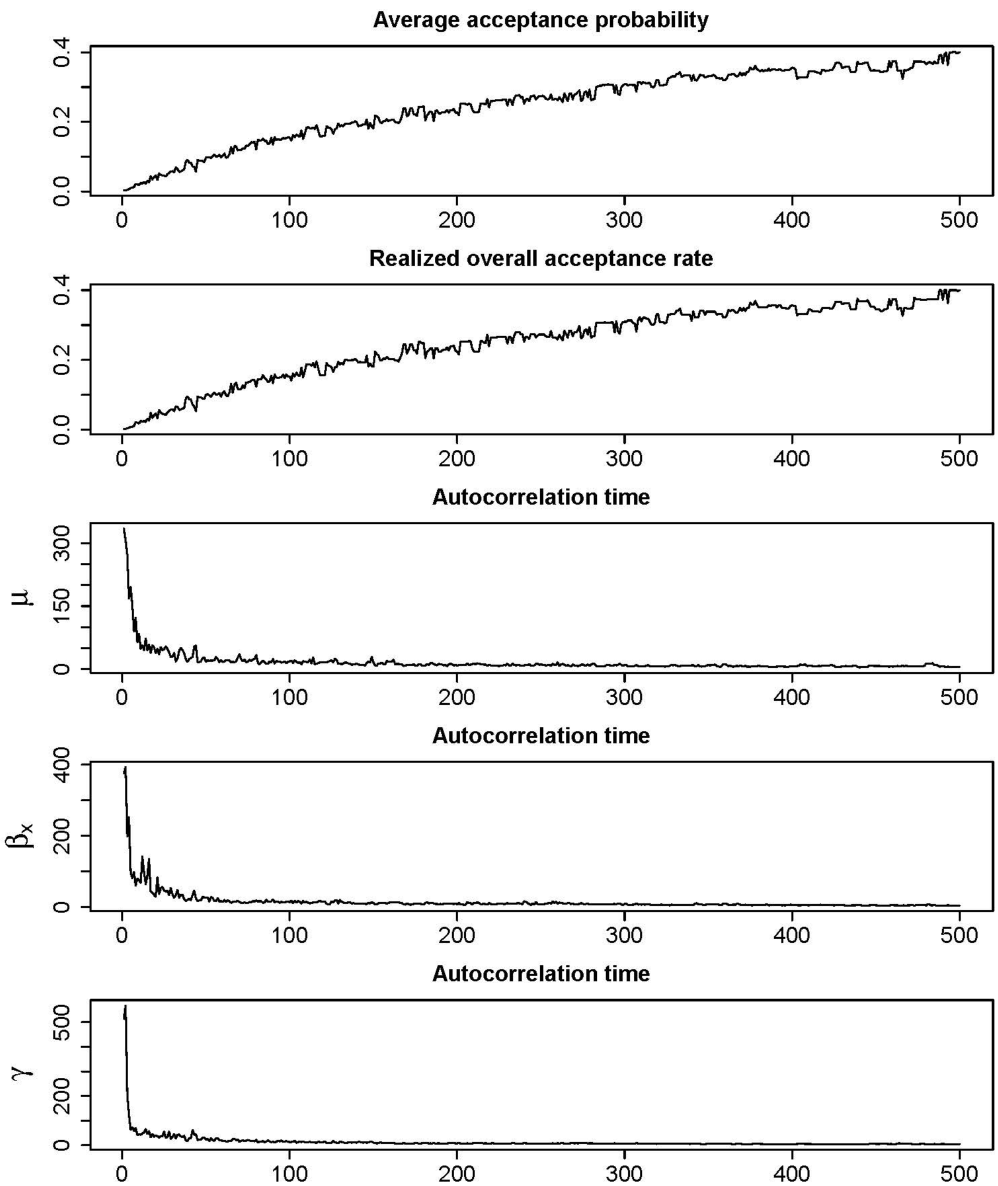


Figure 2: $q(\theta) = p(\theta), J = 4400$

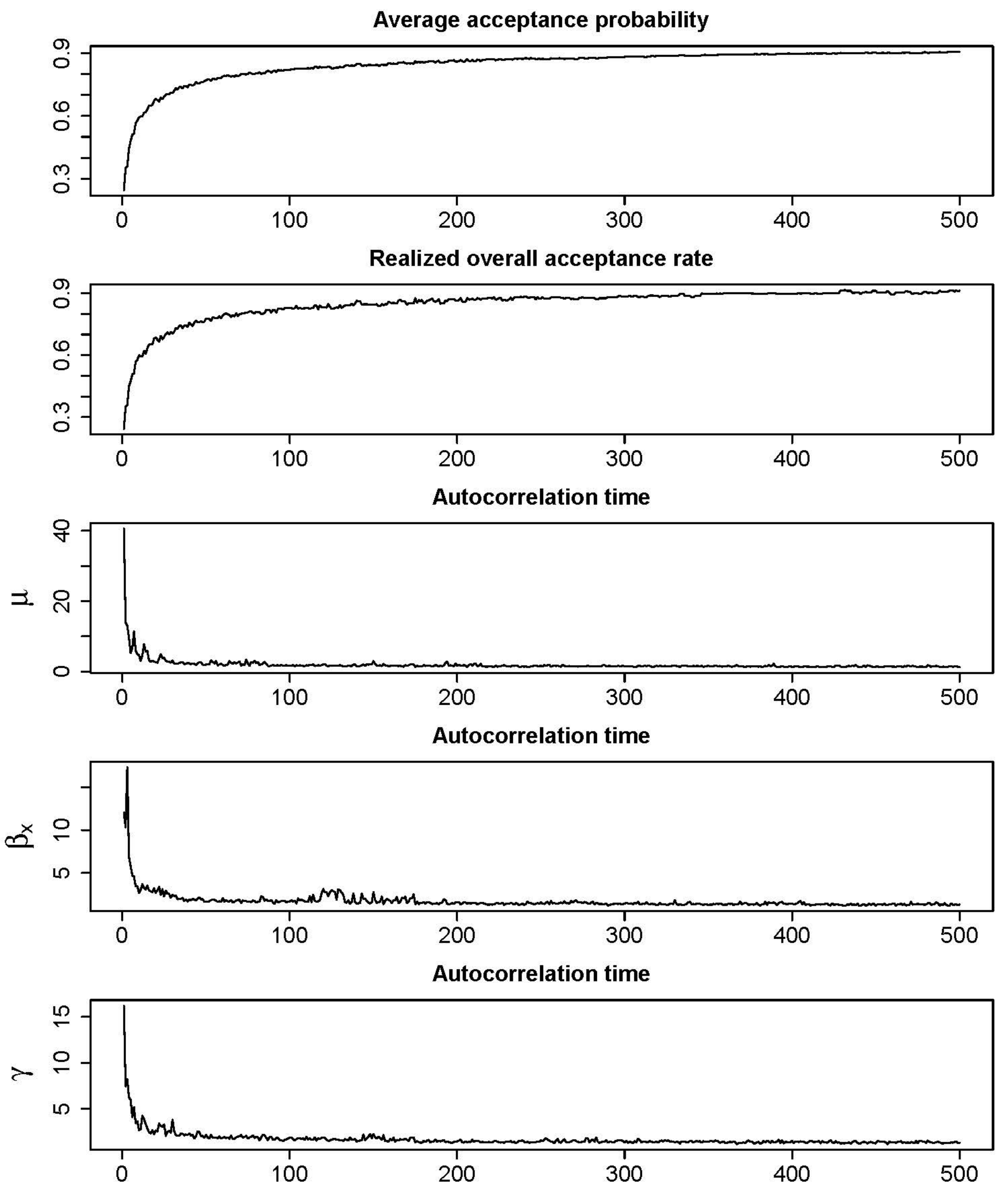


Figure 3: $q(\theta) = \hat{q}(\theta), J = 4400$

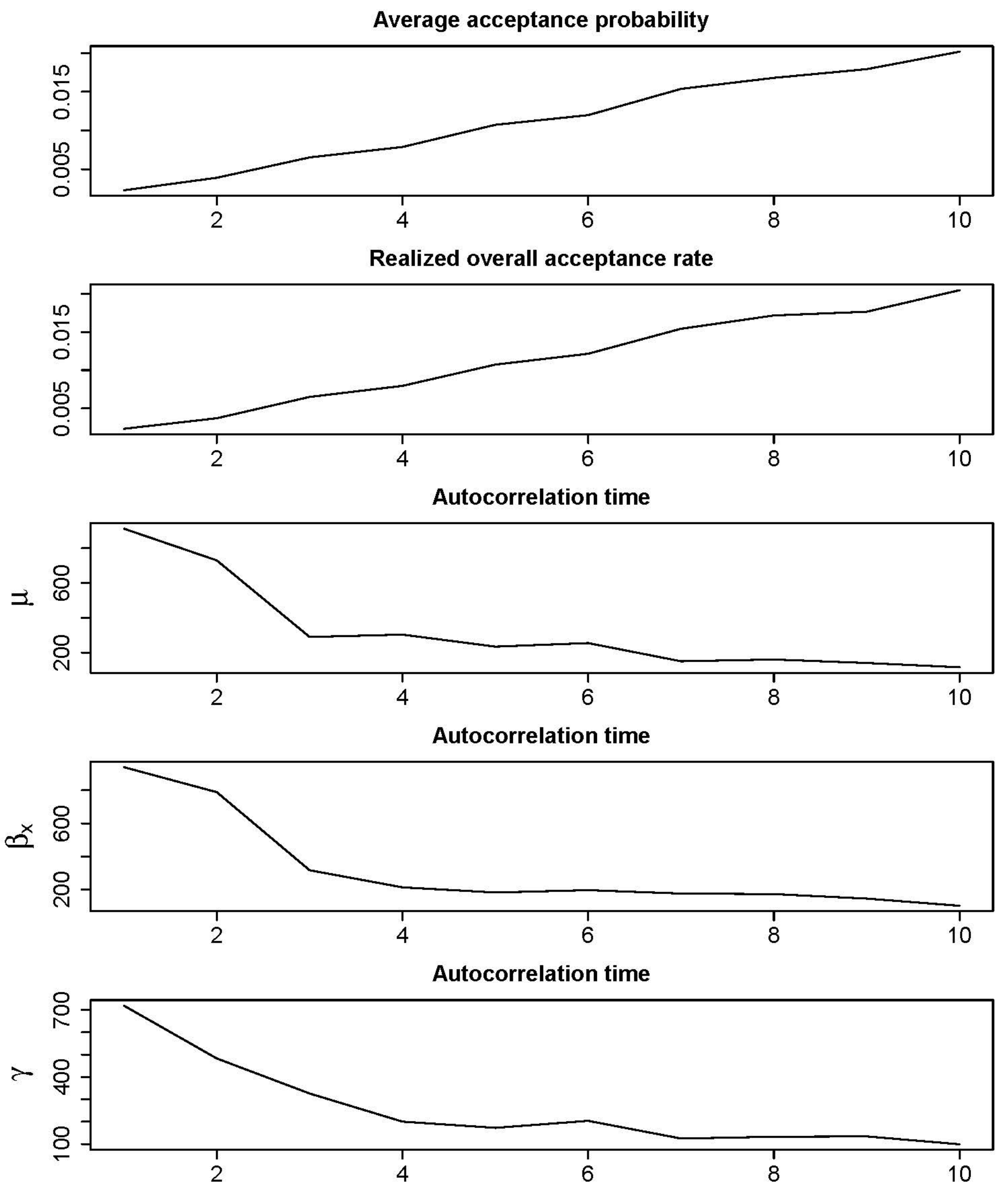


Figure 4: $q(\theta) = p(\theta), J = 110000$

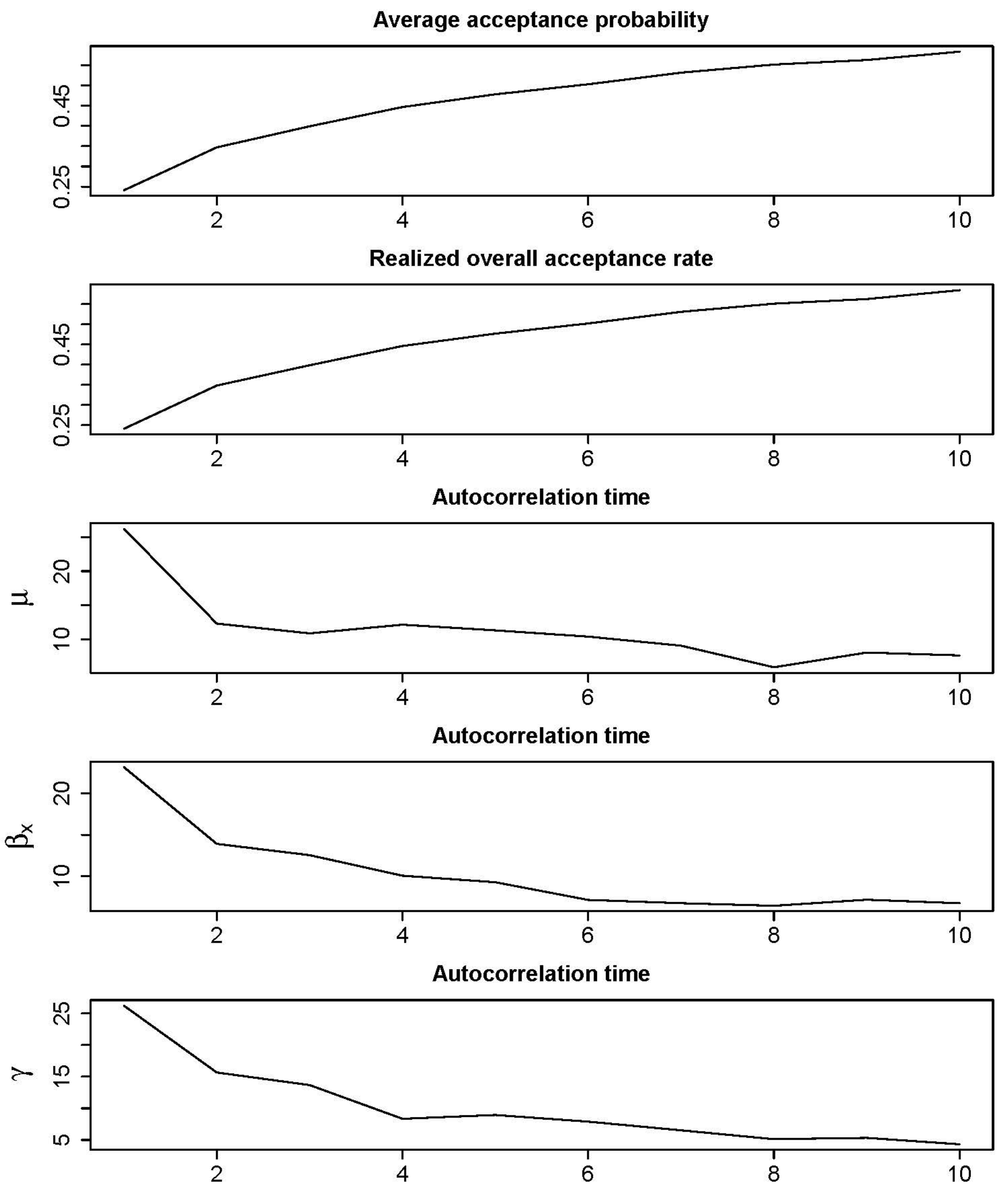


Figure 5: $q(\theta) = \hat{q}(\theta), J = 110000$

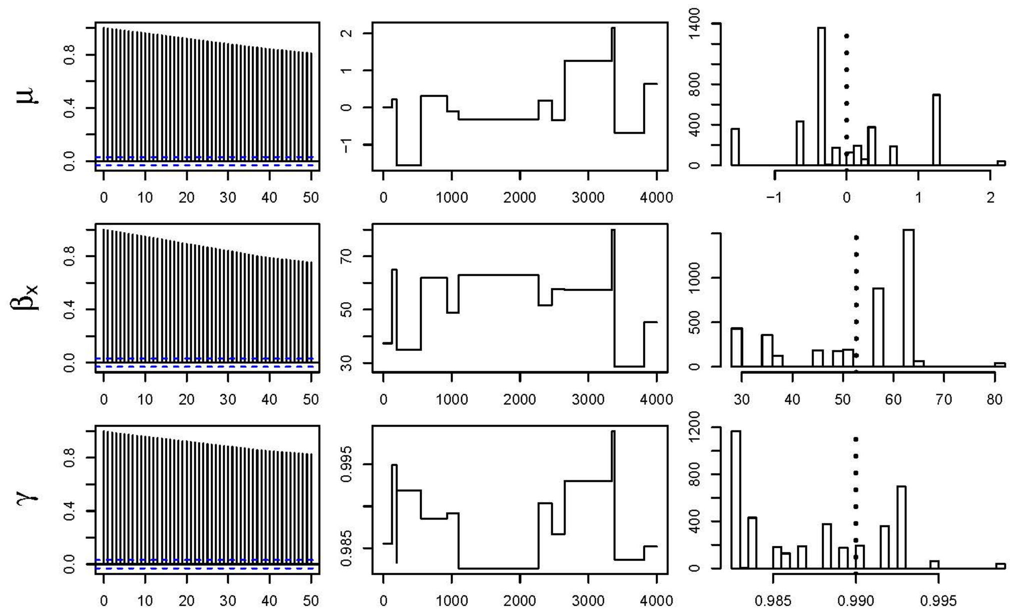
Algorithm 9: MTiPMMH for Different Values of I

for $i = 1 : \overline{I}$
    run lines 2–5 of Algorithm 8
for $I = 1 : \overline{I}$
    run lines 6–9 of Algorithm 8
    $(\theta_{I,1:J}, \xi_{I,1:J}) = (\theta_{1:J}, \xi_{1:J})$
output: $\theta_{1:\overline{I},1:J}, \xi_{1:\overline{I},1:J}$

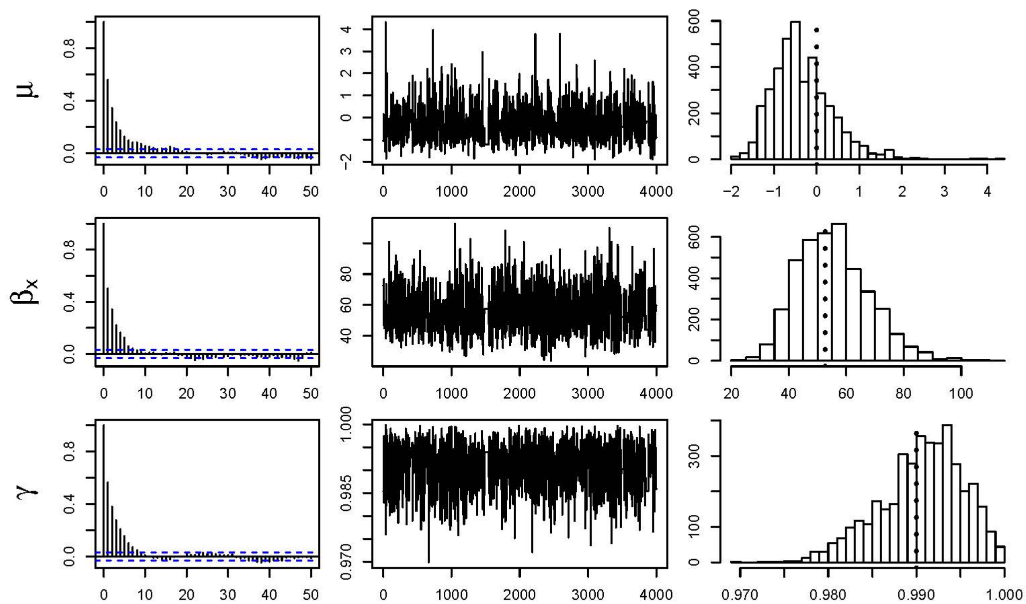
even if $J = 110000$ and I is varied from 1 to only 10. The figures suggest that one can significantly improve mixing if one increases I from 1 to 10, regardless the proposal used. This is useful since the time required to run Algorithm 8 on a multithreaded computer does not increase significantly as I is increased from 1 up to the number of threads available. Thus the time required to run Algorithm 8 with $I = 10$ is not substantially longer than the time required to run the algorithm with $I = 1$, which is simply iPMMH.

Figure 6 shows the ACF (autocorrelation function) plots, trace plots, and histograms for $I = 1$ and $I = 500$ using the same data as used for Figure 2. Figure 6(a) indicates that iPMMH suffers from high autocorrelation and poor mixing. This is mostly because the prior $p(\theta)$ here is not an efficient proposal. Even with this poor proposal, if I is chosen to be sufficiently large, MTiPMMH exhibits low autocorrelation and good mixing, as shown in Figure 6(b). The histograms are rather rough here since J is only 4400.

On a standard PC, it may be easier to use a large value for J and a small value for I. It may also be possible to significantly improve mixing by constructing a proposal using samples from a pilot run. This is indeed the case here, as shown in Figure 7(a), which uses the same data as used for Figure 5. With $q(\theta) = \hat{q}(\theta)$ and $J = 110000$, iPMMH exhibits fairly low autocorrelation and good mixing. Figure 7(b) shows that one can visibly improve the performance by using MTiPMMH with $I = 10$. As discussed above, this does not significantly increase computation time if Algorithm 8 is implemented on a PC with at least 10 threads.

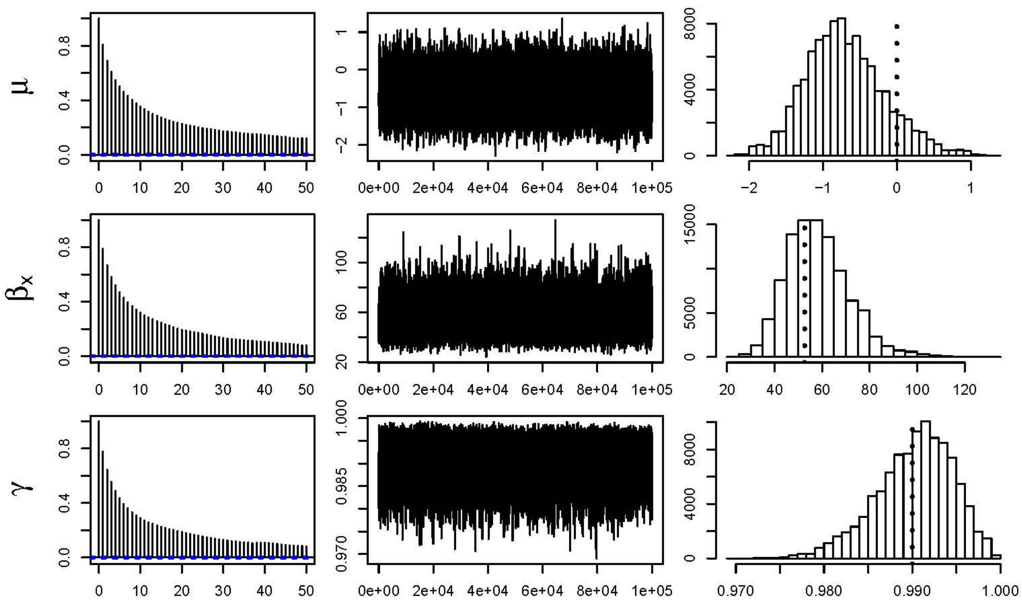


(a) $I = 1$

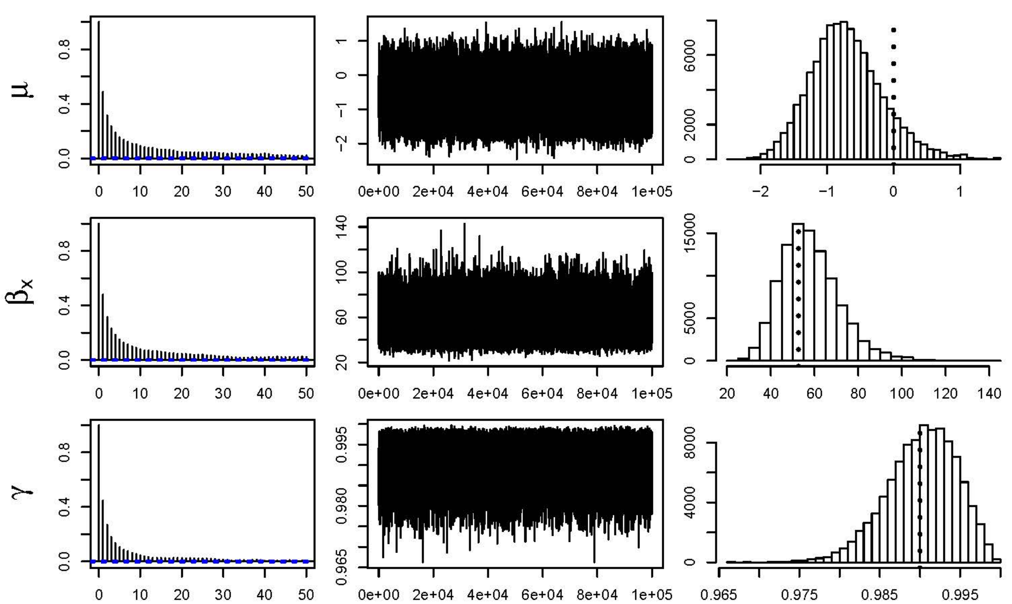


(b) $I = 500$

Figure 6: $q(\theta) = p(\theta), J = 4400$: ACF plots, trace plots, and histograms (dotted lines indicate true values)



(a) $I = 1$



(b) $I = 10$

Figure 7: $q(\theta) = \hat{q}(\theta), J = 110000$: ACF plots, trace plots, and histograms (dotted lines indicate true values)

6 Discussion

In this paper we have proposed a multiple-try extension of the PMMH algorithm with an independent proposal, which we call MTiPMMH. In this algorithm, $I \in \mathbb{N}$ parameter particles are sampled from the independent proposal. For each of them, a particle filter with $K \in \mathbb{N}$ state particles is run. Then one of the parameter particles is resampled according to their marginal likelihood estimates. The state of the chain is updated with a probability based on the ratio between the averages of marginal likelihood estimates. We have shown that MTiPMMH has the following three asymptotic properties: (i) the distribution of the Markov chain generated by the algorithm converges to the posterior of interest in total variation; (ii) as I increases to ∞, the acceptance probability at each iteration converges to 1 *a.s.*; and (iii) as I increases to ∞, the autocorrelation of any order for any parameter with bounded support converges to 0. These results indicate that MTiPMMH generates almost i.i.d. samples from the posterior for sufficiently large I.

It is well known that as the number of iterations increases to ∞, the Markov chain generated by a standard MCMC algorithm converges in distribution to the posterior of interest. Our results offer another dimension in which one can approach the correct posterior. As they are valid for any proposal (satisfying regularity conditions) and any number K of state particles, we believe that our asymptotic results become increasingly relevant as parallel computing with supercomputers, GPUs, and clouds becomes increasingly accessible.

To show our theoretical results, we reformulated iPMMH as an iMH algorithm. The same procedure can in fact be used to reformulate PMMH as an MH algorithm. Thus one can apply the standard MTM method to this MH algorithm to obtain a multiple-try extension of PMMH without assuming an independent proposal. However, the practical value of this extension is not necessarily clear if one wishes to obtain almost i.i.d. samples asymptotically. Note that if one is to obtain i.i.d. samples, then the acceptance probability must always be equal to 1. This implies that the proposal must be effectively independent; for otherwise it would inevitably cause autocorrelation.

Although our theoretical results require I to be sufficiently large, our numerical experiments suggest that one can visibly improve mixing by increasing I from 1 to a small number like 10. This does not significantly increase computation time if a computer with at least the same number of threads is used. One can further improve the overall performance of the algorithm by constructing a proposal using samples from a pilot run of a standard MCMC algorithm. Our algorithm, which assumes an independent proposal, makes it easy to utilize such a proposal.

Appendix A Proof of Theorem 3.1

Consider iMH($\tilde{q}, \tilde{g}$) (Alg. 2). We start by verifying Assumption 3.1 for $\tilde{q}$ and $\tilde{g}$. Note from (16) that $\tilde{q} : \tilde{\Theta} \to \mathbb{R}_{++}$ is a probability density. We have

$$\begin{aligned}
\int \tilde{g}(\tilde{\theta}) d\tilde{\theta} &= \int \int \tilde{q}(\theta^{1:I}, u) w^* du \, d\theta^{1:I} &(41)\\
&= \int \int \left[\tilde{q}(\theta^{1:I}, u) w^* \sum_{i=1}^{I} 1\{u \in U_i\} \right] du \, d\theta^{1:I} &(42)\\
&= \sum_{i=1}^{I} \int \int \tilde{q}(\theta^{1:I}, u) w^* 1\{u \in U_i\} du \, d\theta^{1:I} &(43)\\
&= \sum_{i=1}^{I} \int \left[\prod_{s=1}^{I} q(\theta^s) \right] w^* \frac{w^i}{I w^*} d\theta^{1:I} &(44)\\
&= I^{-1} \sum_{i=1}^{I} \int \left[\prod_{s=1}^{I} q(\theta^s) \right] \frac{g(\theta^i)}{q(\theta^i)} d\theta^{1:I} &(45)\\
&= I^{-1} \sum_{i=1}^{I} \int \left[\prod_{s \in [1:I], s \neq i}^{I} q(\theta^s) \right] g(\theta^i) d\theta^{1:I} &(46)\\
&= I^{-1} \sum_{i=1}^{I} \int g(\theta^i) d\theta^i < \infty, &(47)
\end{aligned}$$

where $\sum_{i=1}^{I} 1\{u \in U_i\} = 1$ in (42), (44) uses (16) and (12) and (17), (45) uses line 3 of Algorithm 3, and the last inequality holds by Assumption 3.1. It follows that iMH($\tilde{q}, \tilde{g}$) satisfies Assumption 3.1. Thus by Proposition 3.1, it generates a sequence $\{\tilde{\theta}_j\}_{j=0}^{\infty}$ such that the distribution of $\tilde{\theta}_j$ converges to the stationary density $\tilde{\pi}(\tilde{\theta})$ satisfying (18) as $j \uparrow \infty$.

Let $p(\theta^*)$ be the marginal density of θ^* with respect to $\tilde{\pi}(\tilde{\theta})$. We claim that

$$p(\theta^*) \propto g(\theta^*). \tag{48}$$

To show this, let $B \subset \Theta$ be a Borel set. Recall from line 4 of MTiMH(q, g) (Alg. 3) that $\theta^* = \theta^{i^*}$. Let $P\{\theta^* \in B\}$ be the probability of $\theta^* \in B$ with respect to $\tilde{\pi}(\tilde{\theta})$. Note from (18) that

$$
\begin{aligned}
P\{\theta^* \in B\} &\propto \int\int \tilde{g}(\theta^{1:I}, u) 1\{\theta^{i^*} \in B\} du \, d\theta^{1:I} && (49) \\
&= \int\int \tilde{g}(\theta^{1:I}, u) \sum_{i=1}^{I} [1\{i^* = i\} 1\{\theta^i \in B\}] du \, d\theta^{1:I} && (50) \\
&= \sum_{i=1}^{I} \int\int \tilde{g}(\theta^{1:I}, u) 1\{u \in U_i\} 1\{\theta^i \in B\} du \, d\theta^{1:I} && (51) \\
&= \sum_{i=1}^{I} \int \left[\prod_{s=1}^{I} q(\theta^s)\right] w^* \frac{w^i}{I w^*} 1\{\theta^i \in B\} d\theta^{1:I} && (52) \\
&= I^{-1} \sum_{i=1}^{I} \int \left[\prod_{s=1}^{I} q(\theta^s)\right] \frac{g(\theta^i)}{q(\theta^i)} 1\{\theta^i \in B\} d\theta^{1:I} && (53) \\
&= I^{-1} \sum_{i=1}^{I} \int g(\theta^i) 1\{\theta^i \in B\} d\theta^i && (54) \\
&= \int g(\theta^*) 1\{\theta^* \in B\} d\theta^*, && (55)
\end{aligned}
$$

where (52) uses (12) and (17), and (53) uses line 3 of Algorithm 3. Since (49)–(55) hold for any Borel set $B \subset \Theta$, (48) follows.

To see that the marginal distribution of θ_j converges to $\pi(\theta)$ in total variation, let $\tilde{\pi}_j(\tilde{\theta})$ be the distribution of $\tilde{\theta}_j$ for $j \in \mathbb{N}$. By Proposition 3.1, $\tilde{\pi}_j$ converges to $\tilde{\pi}$ in total variation as $j \uparrow \infty$. Let $j \in \mathbb{N}$, and let $B \subset \Theta$ be a Borel set. Let $P_j\{\theta^* \in B\}$ be the probability of $\theta^* \in B$ with respect to $\tilde{\pi}_j(\tilde{\theta})$. We have

$$
\begin{aligned}
&P_j\{\theta^* \in B\} - P\{\theta^* \in B\} && (56) \\
&= \int \tilde{\pi}_j(\tilde{\theta}) 1\{\theta^* \in B\} d\tilde{\theta} - \int \tilde{\pi}(\tilde{\theta}) 1\{\theta^* \in B\} d\tilde{\theta} && (57) \\
&= \int (\tilde{\pi}_j(\tilde{\theta}) - \tilde{\pi}(\tilde{\theta})) 1\{\theta^* \in B\} d\tilde{\theta} && (58) \\
&\leq \int |\tilde{\pi}_j(\tilde{\theta}) - \tilde{\pi}(\tilde{\theta})| d\tilde{\theta} \to 0 \quad \text{as } j \uparrow \infty, && (59)
\end{aligned}
$$

where the convergence holds by the total variation convergence of $\tilde{\pi}_j$ to $\tilde{\pi}$. Since the right-hand side of the inequality in (59) does not depend on B, we have

$$\lim_{j\uparrow\infty} \sup_B |P_j\{\theta^* \in B\} - P\{\theta^* \in B\}| = 0, \tag{60}$$

where the supremum is taken over all Borel sets $B \subset \tilde{\Theta}$. It follows that the marginal distribution of θ_j converges to $\pi(\theta)$ in total variation.

Appendix B Proof of Theorem 3.2

Note that the random seed is reset in lines 1 and 5 of Algorithm 3 in such a way that for any $I_0, I_1 \in \mathbb{N}$ with $I_0 \leq I_1$,

$$\forall i \in [1 : I_0], \quad w^i_{I_1,j} = w^i_{I_0,j}, \tag{61}$$

where $\{w^i_{I_0,j}\}^{i\in[1:I_0]}_{j\in[1:J]}$ and $\{w^i_{I_1,j}\}^{i\in[1:I_1]}_{j\in[1:J]}$ are given by line 7 of MTiMH(I) (Alg. 4) with $I = I_0$ and $I = I_1$, respectively. In words, by increasing I from I_0 to I_1, the values of $w^i_{I,j}$ chosen for $I = I_0$ remain unchanged. We need this property to apply the strong law of large numbers below.

Fix $j \in [1 : J]$. Note that for any $I \in \mathbb{N}$,

$$\{\theta^i_{I,j}\}^{i\in[1:I]}_{j\in[1:J]} \text{ is i.i.d. across } i \in [1 : I] \text{ and } j \in [1 : J]. \tag{62}$$

Let $I \in \mathbb{N}$. By (62), for any $i \in [1 : I]$ we have

$$Ew^i_{I,j} = \int \frac{g(\theta)}{q(\theta)} q(\theta) d\theta = \int g(\theta) d\theta < \infty, \tag{63}$$

where E is the expectation operator, and the inequality holds by integrability of g. Note from lines 11 and 14 of MTiMH(I) (Alg. 4) that $w_{I,j-1} \leq \max\{w^*_{I,j'}\}^{j-1}_{j'=1}$; thus

$$\frac{w^*_{I,j}}{\max\{w^*_{I,j'}\}^{j}_{j'=1}} \leq r_{I,j} \leq 1. \tag{64}$$

For each $j' \in \mathbb{N}$ with $j' \leq j$, we have $w^*_{I,j'} \to Ew^i_{I,j'}$ *a.s.* as $I \uparrow \infty$ by the strong law of large numbers and (63). Hence letting $I \uparrow \infty$ in (64), we have $r_{I,j} \to 1$ *a.s.*

Appendix C Proof of Theorem 3.3

If Θ is a bounded set, we set $\Theta_1 = \Theta$. Since Θ_1 is a bounded subset of $\mathbb{R}^{n_1}$, there exists $\overline{\theta} \in (0, \infty)$ such that

$$\forall \theta \in \Theta, \quad \max_{n \in [1:n_1]} |\theta(n)| \le \overline{\theta}. \tag{65}$$

For $I, j \in \mathbb{N}$ and $n \in [1 : n_1]$, we define

$$\mu_I(n) = E\theta^*_{I,j}(n), \tag{66}$$
$$\mu_{I,j}(n) = E\theta_{I,j}(n), \tag{67}$$
$$\nu_I(n) = E(\theta^*_{I,j}(n))^2, \tag{68}$$
$$\nu_{I,j}(n) = E(\theta_{I,j}(n))^2, \tag{69}$$
$$\mu(n) = \frac{Ew^i_{I,j}\theta^i_{I,j}(n)}{Ew^i_{I,j}}, \tag{70}$$
$$\nu(n) = \frac{Ew^i_{I,j}(\theta^i_{I,j}(n))^2}{Ew^i_{I,j}}. \tag{71}$$

Note from (62) that μ_I and ν_I do not depend on j, and that μ and ν depend neither i nor j.

C.1 Preliminary Lemmas

In this subsection we show some properties of the functions defined in (66)–(71).

Lemma C.1. *For any $n \in [1, n_1]$ we have*

$$\text{(a)} \lim_{I \uparrow \infty} \mu_I(n) = \mu(n), \qquad \text{(b)} \lim_{I \uparrow \infty} \nu_I(n) = \nu(n). \tag{72}$$

Proof. Fix $n \in [1, n_1]$. Throughout the proof we suppress dependence on n. Line 8 of Algorithm 4 implies that

$$\mu_I = E\left[\frac{\sum_{i=1}^I w^i_{I,j}\theta^i_{I,j}}{\sum_{i=1}^I w^i_{I,j}}\right]. \tag{73}$$

Note from (65) that

$$\left| \frac{\sum_{i=1}^{I} w_{I,j}^i \theta_{I,j}^i}{\sum_{i=1}^{I} w_{I,j}^i} \right| \leq \overline{\theta}. \tag{74}$$

From (73) we have

$$\mu_I = E\left[\frac{I^{-1}\sum_{i=1}^{I} w_{I,j}^i \theta_{I,j}^i}{I^{-1}\sum_{i=1}^{I} w_{I,j}^i} \right] \to \frac{E w_{I,j}^i \theta_{I,j}^i}{E w_{I,j}^i} = \mu \quad \text{as } I \uparrow \infty, \tag{75}$$

where the convergence holds by the strong law of large numbers, (63), the dominated convergence theorem, and (74). We have verified (72)(a). We obtain (72)(b) by the same argument with $(\theta_{I,j}^i)^2$ replacing $\theta_{I,j}^i$. $\square$

Lemma C.2. *For any $j \in [1:J]$ we have*

$$\lim_{I \uparrow \infty} E(1 - r_{I,j}) = 0. \tag{76}$$

Proof. By Theorem 3.2, we have $r_{I,j} \to 1$ *a.s.* as $I \uparrow \infty$. Hence $1 - r_{I,j} \to 0$ *a.s.* as $I \uparrow \infty$. Since $1 - r_{I,j} \leq 1$ for all $I \in \mathbb{N}$, we obtain (76) by the dominated convergence theorem. $\square$

Lemma C.3. *For any $n \in [1, n_1]$ and $j \in [1:J]$ we have*

$$\text{(a)} \ \lim_{I \uparrow \infty} \mu_{I,j}(n) = \mu(n), \qquad \text{(b)} \ \lim_{I \uparrow \infty} \nu_{I,j}(n) = \nu(n). \tag{77}$$

Proof. Fix $n \in [1, n_1]$. Throughout the proof we suppress dependence on n. Let $I \in \mathbb{N}$. Note from line 11 of Algorithm 4 that $\theta_{I,1} = \theta_{I,1}^*$. Hence $\mu_{I,1} = \mu_I$; thus (77)(a) holds for $j = 1$ by Lemma C.1.

Let $j \in [2:J]$ and $I \in \mathbb{N}$. From line 14 of Algorithm 4, we have

$$\mu_{I,j} = E[(1 - r_{I,j})\theta_{I,j-1} + r_{I,j}\theta_{I,j}^*]. \tag{78}$$

By the triangle inequality, we have

$$\begin{aligned} |\mu_{I,j} - \mu| &\leq |\mu_{I,j} - \mu_I| + |\mu_I - \mu| && (79)\\ &= |E(1 - r_{I,j})(\theta_{I,j-1} - \theta_{I,j}^*)| + |\mu_I - \mu| && (80)\\ &\leq E(1 - r_{I,j}) 2\overline{\theta} + |\mu_I - \mu|, && (81) \end{aligned}$$

where (80) uses (66). Hence (77)(a) holds by Lemmas C.2 and C.1.

We obtain (77)(b) by the same argument with $(\theta_{I,j})^2$ and $(\theta_{I,j}^*)^2$ replacing $\theta_{I,j}$ and $\theta_{I,j}^*$. $\square$

Lemma C.4. *For any $n \in [1, n_1]$ and $j \in [1 : J]$ we have*

$$\lim_{I\uparrow\infty} Var(\theta_{I,j}(n)) = \nu(n) - \mu(n)^2 > 0, \tag{82}$$

where $Var(\theta_{I,j}(n))$ is the variance of $\theta_{I,j}$.

Proof. Fix $n \in [1, n_1]$; we suppress dependence on n. Let $j \in [1 : J]$ and $I \in \mathbb{N}$. Recalling (69) and (67), we have

$$Var(\theta_{I,j}) = E(\theta_{I,j})^2 - (E\theta_{I,j})^2 = \nu_{I,j} - (\mu_{I,j})^2. \tag{83}$$

Hence the equality in (82) holds by Lemma C.3. To see the inequality, let $\omega^i = w^i_{I,j}/Ew^i_{I,j}$. Then in view of (70) and (71), it is straightforward to show that

$$E\omega^i(\theta^i_{I,j} - E\omega^i\theta^i_{I,j})^2 = \nu - \mu^2. \tag{84}$$

The left-hand side is strictly positive since $\omega^i > 0$ and $\theta^i_{I,j} \neq \mu$ *a.s.* by Assumption 3.1. Thus the desired inequality follows. $\square$

C.2 Completing the Proof of Theorem 3.3

Fix $j \in [2 : J], s \in [1 : j-1]$, and $n, n' \in [1 : n_1]$. For $I \in \mathbb{N}$ define

$$z_{I,j-s} = \theta_{I,j-s}(n) - \mu_{I,j-s}(n), \tag{85}$$
$$z'_{I,j-1} = \theta_{I,j-1}(n') - \mu_{I,j-1}(n'), \tag{86}$$
$$z'_{I,j} = \theta_{I,j}(n') - \mu_{I,j}(n'), \tag{87}$$
$$z^*_{I,j} = \theta^*_{I,j}(n') - \mu_{I,j}(n'), \tag{88}$$
$$\beta_{I,j,s} = Ez_{I,j-s}z'_{I,j}. \tag{89}$$

For any $I \in \mathbb{N}$ we have

$$Cor(\theta_{I,j-s}(n), \theta_{I,j}(n')) = \beta_{I,j,s}/[Var(\theta_{I,j-s}(n))Var(\theta_{I,j}(n'))]^{1/2}. \tag{90}$$

By Lemma C.4, as $I \uparrow \infty$, the denominator converges to

$$\{[\nu(n) - \mu(n)^2][\nu(n') - \mu(n')^2]\}^{1/2} > 0. \tag{91}$$

Thus to conclude (19), it suffices to show that

$$\lim_{I\uparrow\infty} \beta_{I,j,s} = 0. \tag{92}$$

To this end, let $I \in \mathbb{N}$. Note from line 14 of Algorithm 4 that

$$\beta_{I,j,s} = E[(1 - r_{I,j}) z_{I,j-s} z'_{I,j-1} + r_{I,j} z_{I,j-s} z^*_{I,j}]. \tag{93}$$

Since $E z_{I,j-s} z^*_{I,j} = E z_{I,j-s} E z^*_{I,j} = 0$ by independence and (85), we have

$$\begin{aligned} |\beta_{I,j,s}| = |\beta_{I,j,s} - E z_{I,j-s} z^*_{I,j}| &= |E(1 - r_{I,j})(z_{I,j-s} z'_{I,j-1} - z_{I,j-s} z^*_{I,j})| \quad (94)\\ &\le E(1 - r_{I,j}) 2\overline{\theta}^2 \to 0 \quad \text{as } I \uparrow \infty, \quad (95) \end{aligned}$$

where the convergence holds by Lemma C.2. Now (92) follows.

Appendix D Construction of $\hat{q}(\theta)$

In this appendix we describe the pilot run and the associated proposal $\hat{q}(\theta)$ used in Section 5. Let $\phi(a|b, c)$ denote the normal density as a function of $a \in \mathbb{R}$ parameterized by $b \in \mathbb{R}$ and $c > 0$ as follows:

$$\phi(a|b, c) = \frac{1}{\sqrt{2\pi c}} \exp\left[-\frac{(a - b)^2}{2c}\right]. \tag{96}$$

Consider an MH sampler for (34) and (35). Initially we draw γ, σ_x^2, and σ_y^2 from the priors given in (36), x_1 from $N(0, 1)$, and $x_{2:T}$ from (34). In each iteration afterwards, given $x_{1:T}$, we define

$$\nu = \sigma_x^{-2} \sum_{t=2}^{T} x_{t-1}^2, \quad \delta = 10, \quad \hat{\gamma} = \frac{\sum_{t=2}^{T} x_t x_{t-1}}{\sum_{t=2}^{T} x_{t-1}^2}. \tag{97}$$

Then we apply the MH update to $\gamma, \sigma_y^2, \sigma_x^2, x_1, \ldots, x_T$ one by one with the following sampling schemes:

$$\begin{aligned} \gamma &\sim \phi\left(\gamma \left| \frac{\hat{\gamma}\nu + 0.9\delta}{\nu + \delta}, \frac{1}{\nu + \delta}\right)\right|_{(-1,1)}, \quad (98)\\ 1/\sigma_y^2 &\sim \Gamma\left(\frac{T + 2}{2}, \frac{\sum_{t=1}^{T}[y_t^2 \exp(-2\alpha^{-1} x_t)] + 2}{2}\right), \quad (99)\\ 1/\sigma_x^2 &\sim \Gamma\left(\frac{T + 2}{2}, \frac{\sum_{t=2}^{T}(x_t - \gamma x_{t-1})^2 + x_1^2(1 + \gamma^2) + 0.02}{2}\right), \quad (100)\\ x_1 &\sim \phi(x_1 | \gamma x_2, \sigma_x^2), \quad (101) \end{aligned}$$

$$x_t \sim \phi\left(x_t \left| \frac{\gamma(x_{t+1}+x_{t-1})}{(1+\gamma^2)}, \frac{\sigma_x^2}{(1+\gamma^2)}\right.\right), \quad \forall t \in [2:T-1], \tag{102}$$

$$x_T \sim \phi(x_T|\gamma x_{T-1}, \sigma_x^2). \tag{103}$$

These distributions are components of the corresponding full conditionals, so that the acceptance probability for each parameter and state variable takes a rather simple form.

We run this algorithm for 110000 iterations, discard the first 10000 iterations as burn-in, and construct the proposal $\hat{q}(\theta)$ using the remaining samples as follows. First let m_x and v_x be the sample mean and variance of the 100000 samples of $\ln \sigma_x^2$ from this pilot run; define m_y and v_y similarly. We also define m_γ and v_γ as the sample mean and variance of the 100000 samples of γ. Let

$$\alpha = \frac{m_\gamma^2(1-m_\gamma)}{v_\gamma} - m_\gamma, \quad \beta = \frac{\alpha}{m_\gamma} - \alpha. \tag{104}$$

Now we specify the proposal $\hat{q}(\theta)$ by

$$\gamma \sim Beta(\alpha, \beta), \tag{105}$$

$$\ln(\sigma_x^2) \sim N(m_x, v_x), \tag{106}$$

$$\ln(\sigma_y^2) \sim N(m_y, v_y), \tag{107}$$

where $Beta(\alpha, \beta)$ is the beta distribution with shape parameters α and β.

References

Andrieu, C., Doucet, A., Holenstein, R., 2009, Particle Markov chain Monte Carlo for efficient numerical simulation, in Monte Carlo and Quasi-Monte Carlo Methods, L'Ecuyer, P., Owen, A.B., eds., Springer, Berlin, pp. 45–60.

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

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

Beaumont, M., 2003, Estimation of population growth or decline in genetically monitored populations, Genetics 164, 1139–1160.

Bédard, M., Douc, R., Moulines, E., 2012, Scaling analysis of mutiple-try MCMC methods, Stochastic Processes and their Applications 122, 758–786.

Del Moral, P., 2004, Feynman-Kac Formulae: Genealogical and Interacting Particle Systems with Applications, Springer, New York.

Doucet, A., Godsill, S., Andrieu, C., 2000, On sequential Monte Carlo sampling methods for Bayesian filtering, Statistics and Computing 10, 197–208.

Fearnhead, P., 2011, MCMC for state-space models, in Handbook of Markov Chain Monte Carlo, ed. by Brooks, S., Gelman, A., Jones, G.L., Meng, X.-L., pp. 513–529.

Golightly, A., Henderson, D.A., Sherlock, C., 2015, Delayed acceptance particle MCMC for exact inference in stochastic kinetic models, Statistical Computing 25, 1039–1055.

Kokkala, J., Särkkä, S., 2015, Combining particle MCMC with Rao-Blackwellized Monte Carlo data association for parameter estimation in multiple target tracking, Digital Signal Processing 47, 84–95.

Liu, J.S., Liang, F., Wong, W.H., 2000, The multiple-try method and local optimization in Metropolis sampling, Journal of the American Statistical Association 95, 121–134.

Martino, L., Louzada, F., 2016, Issues in the multiple-try Metropolis mixing, Computational Statistics, doi:10.1007/s00180-016-0643-9.

Martino, L., Leisen, F., Corander, J., 2016, On multiple try schemes and the particle Metropolis-Hastings algorithm, memo.

Pandolfi, S., Bartolucci, F., Friel, N., 2010, A generalization of the Multiple-Try Metropolis algorithm for Bayesian estimation and model selection, Journal of Machine Learning Research: W&CP 9, 581–588.

Pitt, M.K., Silva, R.d.S, Giordani, P., Kohn, R., 2012, On some properties of Markov chain Monte Carlo simulation methds based on the particle filter, Journal of Econometrics 171, 134–151.

Roberts, G.O., Rosenthal, J.S., 2004, General state space Markov chains and MCMC algorithms, Probability Surveys 1, 20–71.

Shephard, N., Pitt, M.K., 1997, Likelihood analysis of non-Gaussian measurement time series, Biometrika 84, 653–667.

Storvik, G., 2011, On the flexibility of Metropolis-Hastings acceptance probabilities in auxiliary variable proposal generation, Scandinavian Journal of Statistics 38, 342–358.

Tierney, L., 1994, Markov chains for exploring posterior distributions, Annals of Statistics 22, 1701–1728.

Tran, M.-N., Pitt, M.K., Kohn, R., 2016, Adaptive Metropolis-Hastings sampling using reversible dependent mixture propsals, Statistical Computing 26, 361–381.

Wang, G., 2009, Signal extraction from long-term ecological data using Bayesian and non-Bayesian state-space models, Ecological Informatics 4, 69–75.

Wilkinson, D.J., 2012, Stochastic Modelling for Systems Biology, 2nd Ed., CRC Press, Boca Raton.

Yuan, K., Girolami, M., Niranjan, M., 2012, Markov chain Monte Carlo Methods for state-space models with point process observations, Neural Computation 24, 1462–1486.