J. Appl. Prob. **51,** 346–358 (2014)
Printed in England

EXACT SIMULATION FOR DIFFUSION BRIDGES: AN ADAPTIVE APPROACH

HONGSHENG DAI,* *University of Brighton*

Abstract

Exact simulation approaches for a class of diffusion bridges have recently been proposed based on rejection sampling techniques. The existing rejection sampling methods may not be practical owing to small acceptance probabilities. In this paper we propose an adaptive approach that improves the existing methods significantly under certain scenarios. The idea of the new method is based on a layered process, which can be simulated from a layered Brownian motion with reweighted layer probabilities. We will show that the new exact simulation method is more efficient than existing methods theoretically and via simulation.

Keywords: Adaptive rejection sampling; conditioned Brownian motion; diffusion bridge; exact Monte Carlo simulation

2010 Mathematics Subject Classification: Primary 65C05; 65C50
Secondary 60J60

1. Introduction

In this paper we consider the diffusion bridge $X = \{X_t,\ t \in [0, T]\}$, given by

$$\mathrm{d}X_t = \alpha(X_t)\,\mathrm{d}t + \mathrm{d}B_t, \qquad X_0 = x, \qquad X_T = y, \tag{1}$$

where B_t, the coordinate mapping $B_t(\omega) = \omega_t$, is a Brownian motion under the Wiener measure $\mathbb{W}$. Here ω is a typical element of $\mathbb{C} = C([0, T], \mathbb{R})$, the set of continuous mappings from $[0, T]$ to $\mathbb{R}$. Let $\mathbb{Q}_{0,T}^{x,y}$ denote the probability measure induced by the diffusion bridge X, and let $\mathbb{W}_{0,T}^{x,y}$ be the corresponding probability measure for $B = \{B_t,\ t \in [0, T]\}$, $B_0 = x$, $B_T = y$.

In [1], [2], [3], [4], and [5] the authors proposed several novel methods for the exact simulation of diffusion bridges driven by a class of stochastic differential equations (SDEs), which satisfies the following conditions.

Condition 1. (a) $\alpha(\cdot)$ *is continuously differentiable.*

(b) $(\alpha^2 + \alpha')(\cdot)/2$ *is bounded below by a constant l that does not depend on ω.*

(c) $\exp\{\int_0^T \alpha(\omega_s)\,\mathrm{d}\omega_s - \int_0^T \alpha^2(\omega_s)\,\mathrm{d}s/2\}$ *is a martingale with respect to* $\mathbb{W}$.

Their methods involve rejection sampling where the proposals are Brownian bridges, which can be easily simulated. The proposed sample path is accepted according to an appropriate probability density ratio (see [4]), which is derived using Girsanov's transformation formula (see [7] and [8]) if applicable.

Received 23 February 2012; revision received 3 February 2013.

[*] Current address: Department of Mathematical Sciences, University of Essex, Wivenhoe Park, Colchester CO4 3SQ, UK. Email address: hdaia@essex.ac.uk

https://doi.org/10.1239/jap/1402578629 Published online by Cambridge University Press

In practice, their methods may not be efficient owing to a low acceptance probability if (i) the time gap T is large; or (ii) the values of $(\alpha^2+\alpha')(\omega_s)/2-l$ are very large throughout the interval $[0, T]$ (see Example 1 in Section 2). In this paper we focus on dealing with the challenge under case (ii) by using an adaptive approach, where the lower bound of $(\alpha^2+\alpha')(\omega_s)$ is chosen according to the layers of the proposed paths. By doing this, we can always find larger lower bound values for $(\alpha^2+\alpha')(\omega_s)$, thus increasing the acceptance probability significantly. We leave tackling the challenge of case (i) to future research work.

The idea of the new method is based on a layered process, which can be simulated via two steps. First we simulate a layer based on reweighted layer probabilities of a Brownian bridge. Then we simulate a Brownian bridge conditional on the simulated layer. By doing this, the proposal process is not a Brownian bridge any more. We will show that the new method is an exact simulation method. We will also demonstrate that the new method is more efficient than existing methods under certain scenarios.

We begin by stating an example in Section 2 which shows when and why existing methods are not efficient. In Section 3 we will present the probability measure for the proposal process and show how to simulate the proposal process via an adaptive rejection sampling approach. In Section 4 we discuss how to perform rejection sampling for the diffusion bridges based on the proposal process. We will present simulation studies to demonstrate the performance of the new method in Section 5 and conclude with a discussion in Section 6.

2. Rejection sampling and its challenges

Consider the diffusion model (1). Following [4] and using Girsanov's transformation formula we have

$$\frac{\mathrm{d}\mathbb{Q}_{0,T}^{x,y}}{\mathrm{d}\mathbb{W}_{0,T}^{x,y}}(\omega) \propto \exp\biggl\{-\int_0^T \frac{1}{2}(\alpha^2+\alpha')(\omega_s)\,\mathrm{d}s\biggr\}.$$

From Condition 1(b) we know that there exists a constant l such that $l \le \inf_{u\in\mathbb{R}}\{(\alpha^2+\alpha')(u)/2\}$. Then the above formula can be written as

$$\frac{\mathrm{d}\mathbb{Q}_{0,T}^{x,y}}{\mathrm{d}\mathbb{W}_{0,T}^{x,y}}(\omega) \propto \exp\biggl\{-\int_0^T \biggl[\frac{1}{2}(\alpha^2+\alpha')(\omega_s)-l\biggr]\,\mathrm{d}s\biggr\},$$

which is no more than 1. So we can use the rejection sampling methods in [4] and [3] to simulate X, with the proposal process sampled from the measure $\mathbb{W}_{0,T}^{x,y}$.

The acceptance probability, given by

$$\mathbb{E}_{\mathbb{W}}\biggl[\exp\biggl\{-\int_0^T \biggl[\frac{1}{2}(\alpha^2+\alpha')(\omega_s)-l\biggr]\,\mathrm{d}s\biggr\}\biggr], \tag{2}$$

is usually very small if the values of $(\alpha^2+\alpha')(\omega_s)/2-l$ are large throughout the interval $[0, T]$. Even if we choose l as the maximum lower bound, $l = \inf_{u\in\mathbb{R}}\{(\alpha^2+\alpha')(u)/2\}$, the above acceptance probability may still be very small, which we illustrate in the following example.

Example 1. Consider the logistic growth diffusion $V = \{V_s,\ s \in [0, T]\}$ with parameters R, Λ, and σ:

$$\mathrm{d}V_s = RV_s\biggl(1-\frac{V_s}{\Lambda}\biggr)\,\mathrm{d}s + \sigma V_s\,\mathrm{d}B_s.$$

https://doi.org/10.1239/jap/1402578629 Published online by Cambridge University Press

More details about the logistic growth diffusions and their applications can be found in [12]. To simulate $\{V_s\}$, we need to simulate only the transformed diffusion $X_s = -\log(V_s)/\sigma$, which, following [4], solves

$$\mathrm{d}X_s = \alpha(X_s)\,\mathrm{d}s + \mathrm{d}B_s, \qquad \alpha(u) = \frac{\sigma}{2} - \frac{R}{\sigma} + \frac{R}{\sigma\Lambda}\exp\{-\sigma u\}.$$

It can be shown that

$$\frac{\alpha^2(u)+\alpha'(u)}{2} = \frac{\sigma^2}{8} - \frac{R}{2} + \frac{R^2}{2\sigma^2}\bigg[1 - \frac{\exp\{-\sigma u\}}{\Lambda}\bigg]^2. \tag{3}$$

Beskos *et al.* [4] used the maximum lower bound $l = \sigma^2/8 - R/2$ for (3). However, $[\alpha^2(\omega_s) + \alpha'(\omega_s)]/2$ may be much larger than l if the term $[R^2/2\sigma^2][1 - \exp\{-\sigma\omega_s\}/\Lambda]^2$, from (3), is much larger than 0. For example, if we choose $R = 0.5, \sigma = 0.1, \Lambda = 1500$, $\omega_0 = -\log(700)/\sigma$, $\omega_T = -\log(700)/\sigma$, and $T = 2$, the value of (3) will be much larger than l. This will lead to a very small acceptance probability.

We can demonstrate this using simple calculations. Since we simulate a Brownian bridge $\{B_t;\, B_0 = B_2 = -\log(700)/\sigma = -65.51\}$ as the proposal, using the results in [9] or the simplified layered Brownian bridge results in [3], we can easily find that the Brownian bridge $\{B_t,\ t \in [0, 2]\}$ has only a small probability (less than 0.02) of hitting the boundaries -65.51 ± 2. This result implies the acceptance probability

$$\begin{aligned}
&\mathbb{E}_{\mathbb{W}}\bigg[\exp\bigg\{-\int_0^T\bigg[\frac{1}{2}(\alpha^2+\alpha')(\omega_s) - l\bigg]\,\mathrm{d}s\bigg\}\bigg]\\
&\quad\le \mathbb{E}_{\mathbb{W}}\bigg[\exp\bigg\{-\int_0^T \frac{R^2}{2\sigma^2}\bigg[1 - \frac{\exp\{-\sigma\omega_s\}}{\Lambda}\bigg]^2\,\mathrm{d}s\bigg\} I\{\omega_s \in [-67.51, -63.51]\}\bigg] + 0.02\\
&\quad\le \exp\bigg\{-25\bigg[1 - \frac{\exp\{6.751\}}{\Lambda}\bigg]^2\bigg\} + 0.02\\
&\quad\approx 0.03,
\end{aligned}$$

where $I\{\cdot\}$ denotes the indicator function. In fact, according to our simulation study in Section 4, the acceptance probability is just about 0.0007.

An important issue implied by the above example is that, although $\frac{1}{2}(\alpha^2 + \alpha')(\omega_s)$ can theoretically reach l, the probability for that can be very small and there is a very large probability that $\frac{1}{2}(\alpha^2 + \alpha')(\omega_s)$ is much larger than l. Therefore, many simulated proposal processes ω will be rejected.

The above example demonstrates that it is important to improve the current exact simulation methods to achieve higher acceptance probabilities.

3. Simulation of the proposal process

In this section we introduce the measure for the proposal process and show how to simulate the proposal process. Based on the proposal process, a rejection sampling method will then be introduced in Section 4.

The proposal process is simulated via two steps. First we simulate a layer based on reweighted layer probabilities of a Brownian bridge. Then we simulate a Brownian bridge conditional on the simulated layer. So we first introduce the *layers* using the notation given in [3].

https://doi.org/10.1239/jap/1402578629 Published online by Cambridge University Press

Let $\{a_i\}_{i\geq 1}$ be an increasing sequence of positive numbers, and let $a_0 = 0$. Let $\bar{x} = x \wedge y$ and $\bar{y} = x \vee y$. Define the events $\mathcal{D}_i(\bar{x}, \bar{y}; 0, T) = \mathcal{U}_i(\bar{x}, \bar{y}; 0, T) \cup \mathcal{L}_i(\bar{x}, \bar{y}; 0, T)$, where

$$\mathcal{U}_i(x, y; 0, T) = \Big\{\omega\colon \sup_{0\leq s\leq T} \omega_s \in [\bar{y} + a_{i-1}, \bar{y} + a_i]\Big\} \cap \Big\{\omega\colon \inf_{0\leq s\leq T} \omega_s > \bar{x} - a_i\Big\},$$
$$\mathcal{L}_i(x, y; 0, T) = \Big\{\omega\colon \inf_{0\leq s\leq T} \omega_s \in [\bar{x} - a_i, \bar{x} - a_{i-1}]\Big\} \cap \Big\{\omega\colon \sup_{0\leq s\leq T} \omega_s < \bar{y} + a_i\Big\},$$

and $\omega_0 = x$, $\omega_T = y$.

We say that the Brownian bridge is in layer i if $\omega \in \mathcal{D}_i$.

With the above definition, we consider different lower bounds of $(\alpha^2+\alpha')(\omega_s)/2$ for different layers. Condition 1(b) implies that we can find l_i such that

$$l_i \leq \inf_{s\in[0,T],\,\omega\in\mathcal{D}_i} \big\{\tfrac{1}{2}(\alpha^2 + \alpha')(\omega_s)\big\}.$$

Obviously, such $l_i \geq l$ for all i.

Based on the layers and the lower bounds l_i, we consider the following measure for the proposal process:

$$\widetilde{\mathbb{W}}^{x,y}_{0,T}(\omega) \propto \mathbb{W}^{x,y}_{0,T}(\omega) \sum_{i=1}^{\infty} \exp\{-Tl_i\} I\{\omega \in \mathcal{D}_i\}. \tag{4}$$

We then have

$$\begin{aligned}\frac{\mathrm{d}\mathbb{Q}^{x,y}_{0,T}}{\mathrm{d}\widetilde{\mathbb{W}}^{x,y}_{0,T}}(\omega) &\propto \frac{\exp\{-\int_0^T (\alpha^2 + \alpha')(\omega_s)\,\mathrm{d}s/2\}}{\sum_{i=1}^{\infty} \exp\{-Tl_i\} I\{\omega \in \mathcal{D}_i\}}\\ &= \sum_{i=1}^{\infty} \exp\bigg\{-\int_0^T \bigg[\frac{1}{2}(\alpha^2 + \alpha')(\omega_s) - l_i\bigg]\,\mathrm{d}s\bigg\} I\{\omega \in \mathcal{D}_i\},\end{aligned} \tag{5}$$

which is also a value no more than 1. Therefore, if we can simulate from $\widetilde{\mathbb{W}}^{x,y}_{0,T}(\omega)$, which will be discussed later, then, based on (5), we can also use rejection sampling. The acceptance probability is now given by

$$\sum_{i=1}^{\infty} \mathbb{E}_{\mathbb{W}}\bigg[\exp\bigg\{-\int_0^T \bigg[\frac{1}{2}(\alpha^2 + \alpha')(\omega_s) - l_i\bigg]\,\mathrm{d}s\bigg\} I\{\omega \in \mathcal{D}_i\}\bigg], \tag{6}$$

which will be larger than the acceptance probability in (2), since $l_i \geq l$ for all i.

To simulate from $\widetilde{\mathbb{W}}^{x,y}_{0,T}(\omega)$, we write (4) as

$$\begin{aligned}\widetilde{\mathbb{W}}^{x,y}_{0,T}(\omega) &= \frac{\mathbb{W}^{x,y}_{0,T}(\omega) \sum_{i=1}^{\infty} \exp\{-Tl_i\} I\{\omega \in \mathcal{D}_i\}}{\sum_{i=1}^{\infty} \exp\{-Tl_i\} \mathbb{W}^{x,y}_{0,T}\{\omega \in \mathcal{D}_i\}}\\ &= \sum_{i=1}^{\infty} \bigg\{\frac{\exp\{-Tl_i\} \mathbb{W}^{x,y}_{0,T}\{\omega \in \mathcal{D}_i\}}{\sum_{i=1}^{\infty} \exp\{-Tl_i\} \mathbb{W}^{x,y}_{0,T}\{\omega \in \mathcal{D}_i\}} \frac{\mathbb{W}^{x,y}_{0,T}(\omega) I\{\omega \in \mathcal{D}_i\}}{\mathbb{W}^{x,y}_{0,T}\{\omega \in \mathcal{D}_i\}}\bigg\},\end{aligned} \tag{7}$$

where $\mathbb{W}^{x,y}_{0,T}\{\omega \in \mathcal{D}_i\}$ is the probability that the Brownian bridge belongs to the event $\mathcal{D}_i$ (in layer i). To simulate from (7), we can first simulate the layer, say I, according to the probability

$$\widetilde{\mathbb{P}}(I = i) = \frac{\exp\{-Tl_i\} \mathbb{W}^{x,y}_{0,T}\{\omega \in \mathcal{D}_i\}}{\sum_{i=1}^{\infty} \exp\{-Tl_i\} \mathbb{W}^{x,y}_{0,T}\{\omega \in \mathcal{D}_i\}} \tag{8}$$

https://doi.org/10.1239/jap/1402578629 Published online by Cambridge University Press

(this will be discussed in Section 3.1) and then, conditional on the layer I, we simulate ω from $\mathbb{W}_{0,T}^{x,y}(\omega)I\{\omega \in \mathcal{D}_i\}$ (this will be discussed in Section 3.2).

3.1. Simulation for the layer I from (8)

3.1.1. *Preliminaries.* We can write (8) as

$$\widetilde{\mathbb{P}}(I = i) \propto b_i \gamma_i, \tag{9}$$

where

$$b_i = \exp\{-T(l_i - l)\} \tag{10}$$

and $\gamma_i = \mathbb{W}_{0,T}^{x,y}\{\omega \in \mathcal{D}_i\}$. Let $F_i = \sum_{j=1}^{i} \gamma_j = \mathbb{W}_{0,T}^{x,y}\{\omega \in \bigcup_{j=1}^{i} \mathcal{D}_j\}$. For each F_i, there exists an alternating sequence $\{S_{i,j}\}_{j=1}^{\infty}$ such that

$$0 < S_{i,2} < S_{i,4} < S_{i,6} < \cdots < F_i < \cdots < S_{i,5} < S_{i,3} < S_{i,1}, \tag{11}$$

where $S_{i,j}$ depends on T, x, and y, and it can be easily calculated. The formula of $S_{i,j}$ can be found in [3]. Note that $S_{i,j} \to F_i$ as $j \to \infty$. Therefore, an event with probability γ_i or F_i can be simulated as follows, using the above alternating sequence. We simulate a standard uniform random variable U first. If $U \le S_{i,j}$ for an even number of j then $U < F_i$; if $U \ge S_{i,j}$ for an odd number of j then $U > F_i$. Since $S_{i,j} \to F_i$, we can always find either $U \le S_{i,j}$ for an even number of j or $U \ge S_{i,j}$ for an odd number of j by searching from $j = 1$ to ∞. This simulation is given by the subroutine SA($\{S_{i,j}\}, U$) presented in Algorithm 1 below, with the alternating sequence and a standard uniform variable U as inputs. It outputs an indicator $\mathcal{I}_i$ with $\mathbb{P}(\mathcal{I}_i = 1) = F_i$ and $\mathbb{P}(\mathcal{I}_i = 0) = 1 - F_i$, and a value τ_i such that $U \le \tau_i < F_i$ if $\mathcal{I}_i = 1$ and $U \ge \tau_i > F_i$ if $\mathcal{I}_i = 0$.

Algorithm 1. (SA*($\{S_{i,j}\}, U$)*.) Simulate indicator $\mathcal{I}_i$ with $\mathbb{P}(\mathcal{I}_i = 1) = F_i$ and $\mathbb{P}(\mathcal{I}_i = 0) = 1 - F_i$. If $\mathcal{I}_i = 1$, $U \le \tau_i < F_i$; if $\mathcal{I}_i = 0$, $U > \tau_i \ge F_i$.
Inputs. Alternating sequence $\{S_{i,j}\}$, which satisfies (11), and a standard uniform variable U.
Outputs. Indicator $\mathcal{I}_i$ and a value τ_i.

```
1 Set $j = 1$;
2 if $U \le S_{i,j}$ for an even number of $j$ then
3     output $\mathcal{I}_i = 1$ and $\tau_i = S_{i,j}$;
4 else
5     if $U > S_{i,j}$ for an odd number of $j$ then output $\mathcal{I}_i = 0$ and $\tau_i = S_{i,j}$;
6     else $j = j + 1$ and go to step 2;
7 end.
```

See [3] for more details of alternating sequences and how to simulate from $\mathbb{P}(I = i) \propto \gamma_i$.

Simulation of the layer I from (9), however, is not straightforward owing to the factor b_i. Note that, if $l = \inf_{s\in[0,T]}\{(\alpha^2 + \alpha')(\omega_s)/2\}$ and $l_i = \inf_{s\in[0,T],\, \omega\in\mathcal{D}_i}\{(\alpha^2 + \alpha')(\omega_s)/2\}$, then $\{l_i\}$ is a decreasing sequence and $l_i \to l$. Then from (10) we further have $\{b_i\}$ to be an increasing sequence and $b_i \to 1$. In practice, it is often a challenge to find the maximum lower bound $\inf_{s\in[0,T],\, \omega\in\mathcal{D}_i}\{(\alpha^2 + \alpha')(\omega_s)/2\}$ explicitly. We usually consider a sequence $\{l_i\}$ such that $l_i < \inf_{s\in[0,T],\, \omega\in\mathcal{D}_i}\{(\alpha^2 + \alpha')(\omega_s)/2\}$ and its limiting value $l < \inf_{s\in[0,T]}\{(\alpha^2 + \alpha')(\omega_s)/2\}$. Although l_i is not the maximum lower bound for $(\alpha^2 + \alpha')(\omega_s)/2$, we can always choose $\{l_i\}$ such that it is decreasing and converges to l. Then $\{b_i\}$ is increasing and converges to 1. Therefore, in this section we will show how to simulate the layer I from (9) with $\{b_i\}$ an increasing sequence.

https://doi.org/10.1239/jap/1402578629 Published online by Cambridge University Press

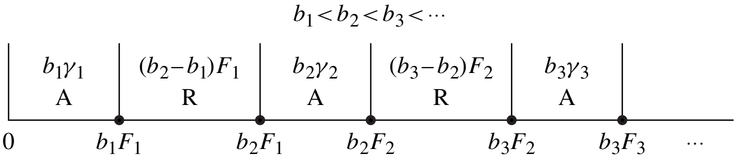

FIGURE 1: Naive rejection sampling when b_i is increasing, where A intervals denote acceptance regions for U and R intervals denote rejection regions for U.

3.1.2. *Simulation from (9).* First we can partition the interval [0, 1] by the ascending sequence

$$\{0, b_1F_1, b_2F_1, b_2F_2, b_3F_2, \ldots, b_jF_{j-1}, b_jF_j, \ldots\},$$

as shown in Figure 1.

To simplify the notation, we define $\delta_1 = 0$, $\{\delta_i = b_iF_{i-1}\}_{i\ge 2}$, $\{\Delta_i = b_iF_i\}_{i\ge 1}$, and $p_i = \Delta_i - \delta_i = b_i\gamma_i$. Then simulating a layer I from $\tilde{\mathbb{P}}$ in (9) is equivalent to simulating a random value U^* from the mixture of uniform distributions

$$f(u) = \sum_{i=1}^{\infty} p_i \frac{\mathrm{U}\{[\delta_i, \Delta_i]\}}{\sum_{i=1}^{\infty} p_i} \propto \sum_{i=1}^{\infty} I\{u \in [\delta_i, \Delta_i]\}, \tag{12}$$

where $\mathrm{U}\{[\delta_i, \Delta_i]\}$ is the uniform distribution on the interval $[\delta_i, \Delta_i]$. This is because $\mathbb{P}(U^* \in [\delta_i, \Delta_i]) = \tilde{\mathbb{P}}(I = i)$.

Note that $\sum_{i=1}^{\infty} p_i$ usually cannot be calculated explicitly in practice. So direct sampling from (12) is not feasible. To solve this problem, we first consider a naive rejection sampling. First simulate a standard uniform variable U. If U is in the acceptance interval $[\delta_i = b_iF_{i-1}, \Delta_i = b_iF_i]$, which has length $b_i\gamma_i$, then accept U as U^* and output $I = i$. On the other hand, if U is in the rejection interval $(\Delta_{i-1} = b_{i-1}F_{i-1}, \delta_i = b_iF_{i-1})$, we resample U. This is a rejection sampling algorithm with acceptance probability $\sum_{i=1}^{\infty} b_i\gamma_i$.

Note that the acceptance probability of the above naive rejection sampling algorithm is $\sum_{i=1}^{\infty} b_i\gamma_i$, which may be very small. However, we can improve the above naive algorithm by considering an adaptive sampling approach.

In the first step we sample a standard uniform random variable U from the interval [0, 1]. If the proposal U is in an acceptance interval then we accept it; if it is in a rejection interval, say in $[\Delta_i, \delta_{i+1}]$, then we reject it and change the proposal density by removing the rejection interval $[\Delta_i, \delta_{i+1}]$ from [0, 1]. Then in the second step we sample U from the new proposal density $\mathrm{U}\{[0,1] \setminus [\Delta_i, \delta_{i+1}]\}$. Repeat the second step and remove the rejection interval whenever the proposal U lies in it, until a simulated U lies in some acceptance interval. Denote the union of all rejection intervals removed from [0, 1] as $\mathcal{R}$. The following algorithm illustrates this procedure.

Algorithm 2. (*A naive adaptive algorithm.*)

```
Set $\mathcal{R}$ as an empty set;
Simulate $U$ uniformly from the set $[0,1] \setminus \mathcal{R}$;
Find the value $i$, such that $\delta_i \le U \le \Delta_i$ or $\Delta_i \le U \le \delta_{i+1}$;
  /* This is to find the subinterval (see the partition in Figure 1) to which $U$ belongs. */
if $\delta_i \le U \le \Delta_i$ then
    accept $U^* = U$, and output $I = i$ as the layer from (9);
else
```

https://doi.org/10.1239/jap/1402578629 Published online by Cambridge University Press

```
7      set $\mathcal{R} = \mathcal{R} \cup [\Delta_i, \delta_{i+1}]$, and go to step 2;
8  end.
```

Lemma 1. *Algorithm 2 simulates U^* from the mixture of uniform distributions (12) and the layer I from (9).*

Proof. The proof follows easily since Algorithm 2 is a rejection sampling with the proposal density uniformly distributed on the set $[0, 1] \setminus \mathcal{R}$, and the target density uniformly distributed on the set $[0, 1] \setminus \bigcup_i [\Delta_i, \delta_{i-1}]$.

Note that the above algorithm is adaptive, since the proposal density of U, $\mathrm{U}\{[0, 1] \setminus \mathcal{R}\}$, will change and become closer to the target density, if the proposal is rejected.

Step 3 of Algorithm 2, checking $\delta_i \le U \le \Delta_i$, can be carried out easily. Although we cannot calculate the explicit value for δ_i and Δ_i, we can have an alternating sequence which converges to δ_i and Δ_i, as

$$\begin{gathered} 0 < b_i S_{i,2} < b_i S_{i,4} < \cdots < \Delta_i = b_i F_i < \cdots < b_i S_{i,3} < b_i S_{i,1}, \\ 0 < b_i S_{i-1,2} < b_i S_{i-1,4} < \cdots < \delta_i = b_i F_{i-1} < \cdots < b_i S_{i-1,3} < b_i S_{i-1,1}. \end{gathered} \tag{13}$$

Clearly, Algorithm 1 can be used to check that $\delta_i \le U \le \Delta_i$. We can run Algorithm 1 for both alternating sequences in (13). $\mathrm{SA}(\{b_i S_{i,j}\}, U)$ and $\mathrm{SA}(\{b_i S_{i-1,j}\}, U)$ will output $(\mathcal{I}_{i,1}, \tau_{i,1})$ and $(\mathcal{I}_{i,2}, \tau_{i,2})$. If $\mathcal{I}_{i,1} = 1$ and $\mathcal{I}_{i,2} = 0$ then $\delta_i < \tau_{i,2} \le U \le \tau_{i,1} < \Delta_i$; if $\mathcal{I}_{i,1} = 0$ and $\mathcal{I}_{i+1,2} = 1$ then $\Delta_i < \tau_{i,1} \le U \le \tau_{i+1,2} < \delta_{i+1}$.

Step 2 of Algorithm 2, however, only works in theory. This is because the proposal density $\mathrm{U}\{[0, 1] \setminus \mathcal{R}\}$ is not available, since the evaluation $\mathcal{R} = \mathcal{R} \cup [\Delta_i, \delta_{i+1}]$ in step 7 cannot be performed because explicit values for Δ_i and δ_{i+1} are not available. However, using (13), we can always work out a subinterval of $[\Delta_i, \delta_{i+1}]$ and we can remove this subinterval from the proposal density instead of removing $[\Delta_i, \delta_{i+1}]$. Note that the subinterval of $[\Delta_i, \delta_{i+1}]$ can be determined via a dynamic approach. This is because it can be obtained when checking the event $\delta_i \le U \le \Delta_i$ using Algorithm 1. When U is in the rejection interval $U \in [\Delta_i, \delta_{i+1}]$, we must have $\Delta_i < \tau_{i,1} \le U \le \tau_{i+1,2} < \delta_{i+1}$ for some $\tau_{i,1}$ and $\tau_{i+1,2}$ outputted by Algorithm 1. Therefore, we have the following practical algorithm, which is equivalent to simulating a proposal U from the hat function

$$g(u) \propto \mathrm{U}\{[0, 1] \setminus \mathcal{R}\},$$

where $\mathcal{R} \subset \bigcup_i [\tau_{i,1}, \tau_{i+1,2}]$, and then accepting U as a sample from $f(u)$ if U belongs to the acceptance interval.

Algorithm 3. (*The practical algorithm.*)

```
1  Set $\mathcal{R}$ as an empty set;
2  Simulate $U$ uniformly from the set $[0, 1] \setminus \mathcal{R}$;
3  Find the value $i$, such that $\delta_i \le U \le \Delta_i$ or $\Delta_i \le U \le \delta_{i+1}$;
     /* $(\mathcal{I}_{i,1}, \tau_{i,1}) = \mathrm{SA}(\{b_i S_{i,j}\}, U)$ and $(\mathcal{I}_{i+1,2}, \tau_{i,2}) = \mathrm{SA}(\{b_i S_{i-1,j}\}, U)$; if $\mathcal{I}_{i,1} = 1$ and
        $\mathcal{I}_{i,2} = 0$ then $\delta_i \le U \le \Delta_i$; if $\mathcal{I}_{i,1} = 0$ and $\mathcal{I}_{i+1,2} = 1$ then $\Delta_i \le U \le \delta_{i+1}$; */
4  if $\mathcal{I}_{i,1} = 1$ and $\mathcal{I}_{i,2} = 0$ (i.e. $\delta_i \le U \le \Delta_i$)
5  then
6      accept $U^* = U$, and output $i$ as the layer;
7  else
8      set $\mathcal{R} = \mathcal{R} \cup [\tau_{i,1}, \tau_{i+1,2}]$, and go to step 2;
9  end.
```

3.2. Simulation of ω conditional on I

Assuming that the Brownian bridge is in layer I, we can construct the layered Brownian bridge with the method in [3] as follows.

Algorithm 4. (*Sampling a Brownian bridge conditional on* $\omega \in \mathcal{D}_I$ *using the method in [3].*)

```
1  Simulate the skeletons of $\omega$ together with either its minimum (with probability 0.5) or its
   maximum (with probability 0.5);
2  if $\omega \notin \mathcal{D}_I$ then
3      reject $\omega$ and go to step 1;
4  else
5      if $\omega \notin \mathcal{U}_I \cap \mathcal{L}_I$ then
6          accept $\omega$;
7      else
8          Simulate $U$ from U[0, 1];
9          if $U < 0.5$ then accept $\omega$;
10         else reject $\omega$ and go to step 1;
11     end
12 end.
```

Using the results in Section 3.1 and Section 3.2, we can actually simulate the proposal processes from (4). Based on the simulated proposal processes, we introduce the rejection sampling in the following section.

4. Rejection sampling for X

The rejection sampling for X is similar to that in [3] and [4], except that we should calculate the lower bound l_I for $(\alpha^2 + \alpha')(\omega_s)$ for all $\omega \in \mathcal{D}_I$. We now provide the new exact simulation algorithm.

Algorithm 5. (*Exact sampling for* X.)

```
1  Simulate $I$ from $\widetilde{\mathbb{P}}(I = i)$ using Algorithm 3;
2  Calculate $l_I = \inf[\alpha^2(u) + \alpha'(u)]/2$, given that $u \in [\bar{x} - a_I, \bar{y} + a_I]$;
3  Calculate $r_I$ such that
```

$$r_I \geq \sup_{t \in [0,T],\, \omega \in \mathcal{D}_I} \left\{ \frac{\alpha^2(\omega_t) + \alpha'(\omega_t)}{2} - l_I \right\};$$

```
4  Simulate $\boldsymbol{\Psi} = \{\psi_1, \dots, \psi_\rho\}$ uniformly distributed on U[0, $T$] and marks $\Upsilon = \{\nu_1, \dots, \nu_\rho\}$
   uniformly distributed on U[0, 1], where $\rho$ is from Poi($r_I T$);
5  Simulate a sample path $\omega$, from $\mathbb{W}_{0,T}^{x,y}$ conditional on $\omega \in D_I$, using Algorithm 4;
6  Compute the acceptance indicator $\mathcal{I} := \prod_{j=1}^{\rho} I\{\phi(\omega_{\psi_j}) < \nu_j\}$, with
```

$$\phi(\cdot) = r_I^{-1}[(\alpha^2 + \alpha')(\cdot)/2 - l_I];$$

```
7  if $\mathcal{I} = 1$ then
8      accept the $\omega$;
9  else
10     return to step 1;
11 end.
```

https://doi.org/10.1239/jap/1402578629 Published online by Cambridge University Press

In Algorithm 5, steps 1, 2, and 5 simulate the proposal process. Steps 3 and 4 simulate a marked Poisson process with constant rate r_I and use the thinning algorithm (see [11]) to simulate a Poisson process with rate $[\alpha^2(\omega_t) + \alpha'(\omega_t)]/2 - l_I$. The probability that the Poisson process with rate $[\alpha^2(\omega_t) + \alpha'(\omega_t)]/2 - l_I$ has no event occurring in $[0, T]$ is equal to the acceptance ratio (5). No event occurring in $[0, T]$ is given by $\mathcal{L} = 1$ in step 7 of the algorithm, which means that the proposal is accepted. More details about the Wiener–Poisson decomposition of $\mathbb{Q}$ can be found in [4] and [3].

5. Simulation studies

In the previous section we showed that Algorithm 5 draws a perfect realization of $\{X_t,\ t \in [0, T]\}$ driven by (1). We can also use a simulation to show that Algorithm 5 provides perfect simulations.

5.1. Simulation example 1

Consider the model in Example 1 with $R = 0.2$, $\sigma = 0.1$, $\Lambda = 1000$, $X_0 = -\log(700)/\sigma$, $X_T = -\log(800)/\sigma$, and $T = 2$. We simulate 10 000 realizations using Algorithm 5 and the exact simulation algorithm in [3]. The two algorithms provide almost exactly the same distribution estimates for X_t at any $t \in [0, T]$. For example, the empirical distribution estimates for $X_{0.5}$, $X_{1.0}$, and $X_{1.5}$ are the same based on simulated realizations using Algorithm 5 and the exact simulation algorithm in [3]. This is shown in Figure 2, from which we can see that the empirical distribution function estimates are the same (almost completely overlap) based on the two different methods.

Now we show that the proposed algorithm is much more efficient than existing algorithms in certain cases. Consider the model in Example 1 with different parameter values: $R = 0.5$, $\sigma = 0.1$, $\Lambda = 1500$, $X_0 = -\log(700)/\sigma$, $X_T = -\log(700)/\sigma$, and $T = 2$. We consider different layer values: $a_i = 0.1i$, $a_i = 0.3i$, $a_i = 0.6i$, and $a_i = 1.0i$. The running time and acceptance probabilities are given in Table 1.

From the results given in Table 1 we can see that the acceptance probability (6) for the new method is overall much larger than the acceptance probability (2) for the existing method. The acceptance probability (2) for existing methods should remain constant for different choices of the layer sequence $\{a_i\}$, as the proposal measure $\mathbb{W}_{0,T}^{x,y}$ does not change. However, the acceptance probability (6) for the new method increases when we choose thinner layers (having smaller values for $a_i - a_{i-1}$). This is consistent with what we expect.

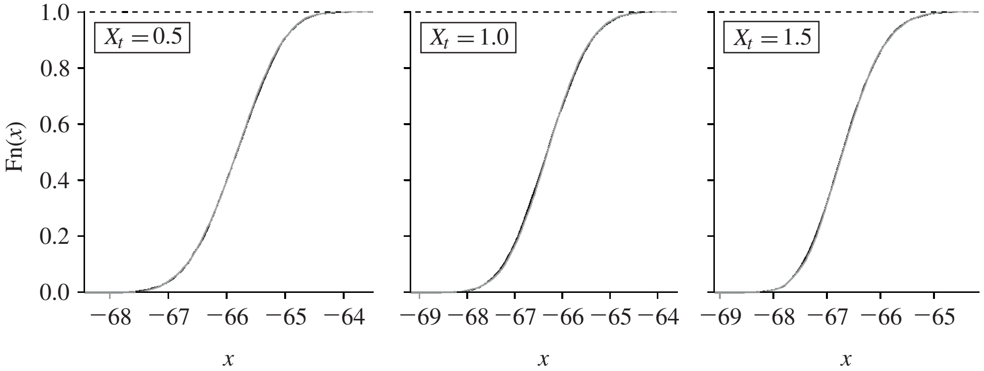

FIGURE 2: The empirical distribution estimates are the same based on the new method and the method in [3].

https://doi.org/10.1239/jap/1402578629 Published online by Cambridge University Press

TABLE 1: Running time comparisons.

a_i	New method		Existing method	
	Running time (s)	Acceptance probability	Running time (s)	Acceptance probability
$0.1i$	99	0.1676	1338	0.0008
$0.3i$	17	0.1542	1026	0.0007
$0.6i$	7.2	0.1309	843	0.0007
$1.0i$	7.4	0.0957	891	0.0008

We also note that the running times of the algorithms increase when we choose thinner layers. For the method in [3] the running time just increases slightly (about $1338/891 \approx 1.5$ times) when the layer decreases from 1.0 to 0.1. This is mainly owing to searching the layers with probability $\mathbb{P}(I = i) = \gamma_i$. For the new method, however, the running time increases more (about $99/7.4 \approx 13$ times) when the layer decreases from 1.0 to 0.1. This is mainly owing to step 1 in Algorithm 5, which uses Algorithm 3 to simulate the layer I with probability $\widetilde{\mathbb{P}}(I = i) \propto b_i \gamma_i$. The probability $\widetilde{\mathbb{P}}$ reweights $\mathbb{P}$ by multiplying the factor b_i. This reweighting will mean that a layer with a larger I value is simulated. Step 3 of Algorithm 3 always starts searching from $i = 1$ to ∞ and it needs more time if we use thinner layers. Therefore we do not suggest using very thin layers for the new methods. For the example that we presented, choosing a layer sequence $a_i = 0.6i$ will provide a very efficient algorithm. In fact, this was also why we used only the adaptive approach for sampling the layers of the proposal process (Algorithm 3) with fixed $\{a_i\}$, but not to select from different $\{a_i\}$ layer sequences.

5.2. Simulation example 2

To show that the proposed algorithm is correct, we also simulate a diffusion process with a known stationary distribution, which allows us to compare the simulated samples with the target distribution. For this purpose, we consider a Langevin diffusion example. Langevin diffusions are important in practice as they form the basis for the construction of a variety of Markov chain Monte Carlo algorithms (see [10]) and can be used to develop new simulation methods from the target distribution, the equilibrium distribution of the Langevin diffusion.

For a given target distribution $\pi(x)$, the Langevin diffusion X_t is defined as

$$\mathrm{d}X_t = \alpha(X_t)\,\mathrm{d}t + \mathrm{d}\omega_t, \qquad \alpha(X_t) = \frac{1}{2}\frac{\mathrm{d}\log(\pi(x))}{\mathrm{d}x}\bigg|_{x=X_t}. \tag{14}$$

Then X_t will have invariant distribution $\pi(x)$ under certain conditions.

We use the following Bayesian posterior distribution of a mixture model as the target distribution $\pi(x)$. Such a simulation example would motivate us to develop new methodologies for simulation from the Bayesian posterior of mixture models, for which existing simulation methods (such as Markov chain Monte Carlo) suffers from slow convergence (see [6]). The results here will also show that the proposed method works for complicated scenarios.

Suppose that we have observations $\xi_1, \dots, \xi_n$ which follow a mixture distribution of two components, having density function $h(\xi_i) = q_1 h_1(\xi_i; x) + q_2 h_2(\xi_i)$. For simplicity, we here assume that the first component density h_1 has unknown parameter x, and the second component density h_2 is known. We consider a Bayesian approach to estimate the unknown parameter x.

https://doi.org/10.1239/jap/1402578629 Published online by Cambridge University Press

Given a prior $\pi_0(x)$, the posterior distribution is given by

$$\pi(x) \propto \prod_{i=1}^{n} [q_1 h_1(\xi_i; x) + q_2 h_2(\xi_i)] \pi_0(x). \tag{15}$$

For the α given in (14), it is extremely challenging to find $\inf_s(\alpha^2 + \alpha')(\omega_s)/2$ explicitly. In practice, we have to use a much smaller lower bound which results in tiny acceptance probabilities, making the existing method impractical.

A lower bound of $(\alpha^2 + \alpha')(x)/2$ can be found as follows. Since $\alpha^2 \geq 0$, we need to find only the lower bound for

$$\begin{aligned}\alpha'(x) &= \biggl[\sum_i \frac{q_1 h_1'(\xi_i; x)}{q_1 h_1(\xi_i; x) + q_2 h_2(\xi_i)}\biggr]' + \biggl[\frac{\pi_0'(x)}{\pi_0(x)}\biggr]' \\ &= \sum_i \frac{q_1 h_1''(\xi_i; x)}{q_1 h_1(\xi_i; x) + q_2 h_2(\xi_i)} - \sum_i \frac{q_1^2 h_1'(\xi_i; x)^2}{[q_1 h_1(\xi_i; x) + q_2 h_2(\xi_i)]^2} + \frac{\pi_0''(x)}{\pi_0(x)} - \frac{\pi_0'(x)^2}{\pi_0(x)^2}.\end{aligned}$$

The function $\alpha'(x)$ is usually bounded, for example, when h is a mixture of two normal components, with $q_1 = 0.7, q_2 = 0.3$, $h_1 = \mathcal{N}(\xi_i; x, 1), h_2 = \mathcal{N}(\xi_i; 0, 1)$, and the prior $\pi_0(x)$ standard normal. Then it is easy to show that

$$\alpha'(x) \geq \sum_i \frac{-0.7\mathcal{N}(\xi_i; x, 1)}{0.7\mathcal{N}(\xi_i; x, 1) + 0.3\mathcal{N}(\xi_i; 0, 1)} - 1 \geq -n + 1.$$

Therefore, we can let the lower bound for each layer be

$$l_i = \inf_x \biggl[\sum_i \frac{-0.7\mathcal{N}(\xi_i; x, 1)}{0.7\mathcal{N}(\xi_i; x, 1) + 0.3\mathcal{N}(\xi_i; 0, 1)} - 1\biggr] < \inf_x \frac{(\alpha^2 + \alpha')(x)}{2}.$$

To justify the correctness of the adaptive algorithm, we consider the above mixture model example with a small sample size $n = 20$. We choose a small sample size because, for small n, we can sample directly from the posterior (with $n = 20$, expanding the product in the posterior will result in about 1 000 000 terms which can be easily handled by all modern computers). We simulate 10 000 samples directly from $\pi(x)$ in (15) and use this as the reference sample. Then we simulate a diffusion process given by (14), using the adaptive method, with X_0 from $\pi(x)$. Note that the diffusion bridge simulation algorithm proposed here can be easily extended to simulate a diffusion process if we simulate the endpoint X_T according to that in [2] and then simulate the diffusion bridge in $[0, T]$. The simulated diffusion process is actually in equilibrium and with equilibrium distribution $\pi(x)$. We choose $T = 0.1$ and collect $X_{0.1}$ as a sample from equilibrium. We repeat this 10 000 times to obtain 10 000 realizations. Samples obtained via the adaptive method have almost the same empirical distribution as the reference sample. See Figure 3.

For the above example with $T = 0.1$, it took about 5 hours to simulate 10 000 realizations via the adaptive approach, by choosing a layer value of $a_i = 0.05i$. However, the existing rejection sampling in [3] is impractical as it does not return a diffusion bridge in the time interval $[0, 0.1]$ within an hour.

https://doi.org/10.1239/jap/1402578629 Published online by Cambridge University Press

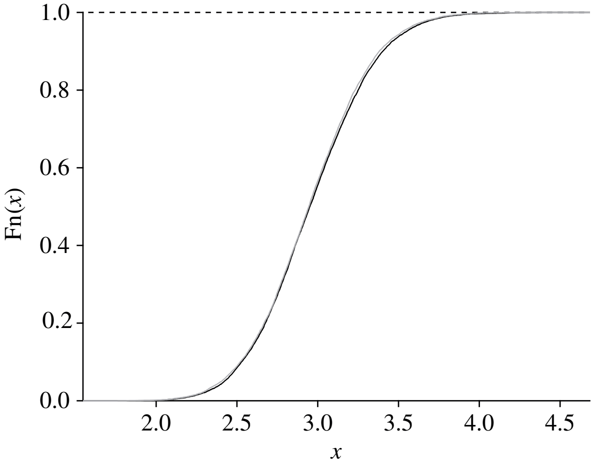

FIGURE 3: The empirical distributions are the same based on the direct sampling and the proposed method.

6. Discussion

We have provided an adaptive approach to carry out exact simulations for a class of diffusion bridges. The new methods draw proposals by simulating reweighted layers and then simulating Brownian bridges conditional on the layer. With such a proposal process, we can use different lower bounds for proposal processes with different layers (Algorithm 5).

Although we only showed that the new method is much more efficient than existing methods under two special scenarios, the arguments can be generalized to many other cases. Simulation example 2 actually demonstrated that, for many complicated problems, we may not obtain the maximum lower bound, but a much smaller lower bound $l < \inf_s \frac{1}{2}(\alpha^2 + \alpha')(\omega_s)$. For such cases, it is important to use the new method in this paper; otherwise, the existing methods may not work practically simply because l is too small. Although the Bayesian mixture model considered in this paper is far from being a practical model, the idea here of simulating Langevin diffusions with a target density as equilibrium could be used to develop new simulation methods. We leave this to future research work.

Simulation example 1 demonstrated that, even if the maximum lower bound can be achieved, the acceptance probability

$$\mathbb{E}_{\mathbb{W}}\biggl[\exp\biggl\{-\int_0^T \biggl[\frac{1}{2}(\alpha^2 + \alpha')(\omega_s) - l\biggr]\,\mathrm{d}s\biggr\}\biggr]$$

can still be very small. This is usually true when there is just a very small probability that $\frac{1}{2}(\alpha^2 + \alpha')(\omega_s)$ reaches the maximum lower bound l, but there is a very large probability that $\frac{1}{2}(\alpha^2 + \alpha')(\omega_s)$ is much larger than l throughout the interval $[0, T]$.

Algorithm 3 was proposed based on an increasing sequence $\{b_i\}$, since we can always choose $\{b_i\}$ to be an increasing sequence. In practice, one may want to use numerical methods to find the lower bound l_i, if the analytical lower bound available is too small. The lower bound obtained via numerical methods, however, may depend on the initial value and the multimodality of $(\alpha^2 + \alpha')(\cdot)$ (simulation example 2), which makes it impossible to always guarantee that l_i is decreasing. Therefore, if numerical methods are used, $\{b_i\}$ may not be an increasing sequence. Even if $\{b_i\}$ is nonincreasing, we can still use an adjusted version of Algorithm 3. We can

https://doi.org/10.1239/jap/1402578629 Published online by Cambridge University Press

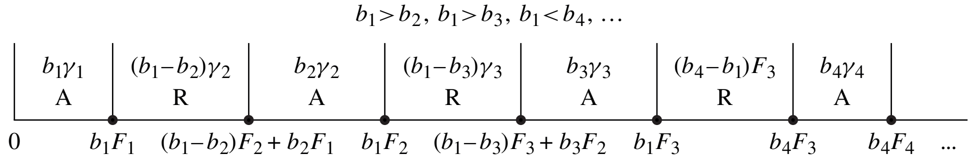

Figure 4: Partition of [0, 1] for a general sequence of b_i, where A intervals denote acceptance regions and R intervals denote rejection regions.

choose a different partition of [0, 1], $\{\delta_1, \Delta_1, \delta_2, \Delta_2, \ldots\}$ as follows. Suppose that b_{j^*} is the maximum value for all $\{b_j,\ j = 1, \ldots, i-1\}$. We let $\delta_1 = 0$ and $\Delta_1 = b_1F_1$. For $i \geq 2$, we let $\delta_i = b_iF_{i-1}$ and $\Delta_i = b_iF_i$ when $b_{j^*} \leq b_i$. We let $\delta_i = (b_{j^*} - b_i)F_i + b_iF_{i-1}$ and $\Delta_i = b_{j^*}F_i$. In Figure 4 we illustrate how to partition [0, 1] when b_i is not an increasing sequence. The acceptance and rejection intervals are given in Figure 4. Note that the partition can be found, step by step from $i = 1$ to ∞, as the algorithm is searching which interval the standard uniform random variable U belongs to. Algorithm 3 can therefore be simply adapted for such a nonincreasing sequence $\{b_i\}$.

The new method, however, will still be inefficient if T is large. We leave this to future research work and a possible way might be looking for a kind of adaptive approach as well.

References

[1] Beskos, A. and Roberts, G. O. (2005). Exact simulation on diffusions. *Ann. Appl. Prob.* **15,** 2422–2444.

[2] Beskos, A., Papaspiliopoulos, O. and Roberts, G. O. (2006). Retrospective exact simulation of diffusion sample paths with applications. *Bernoulli* **12,** 1077–1098.

[3] Beskos, A., Papaspiliopoulos, O. and Roberts, G. O. (2008). A factorisation of diffusion measure and finite sample path constructions. *Methodology Comput. Appl. Prob.* **10,** 85–104.

[4] Beskos, A., Papaspiliopoulos, O., Roberts, G. O. and Fearnhead, P. (2006). Exact and computationally efficient likelihood-based estimation for discretely observed diffusion processess. *J. R. Statist. Soc. B* **68,** 333–382.

[5] Casella, B. and Roberts, G. O. (2008). Exact Monte Carlo simulation of killed diffusions. *Adv. Appl. Prob.* **40,** 273–291.

[6] Celeux, G., Hurn, M. and Roberts, C. P. (2000). Computational and inferential difficulties with mixture posterior distributions. *J. Amer. Statist. Assoc.* **95,** 957–970.

[7] Karatzas, I. and Shreve, S. E. (1991). *Brownian Motion and Stochastic Calculus*, 2nd edn. Springer, New York.

[8] Øksendal, B. (1995). *Stochastic Differential Equations*, 4th edn. Springer, Berlin.

[9] Potzelberger, K. and Wang, L. (2001). Boundary crossing probability for Brownian motion. *J. Appl. Prob.* **38,** 152–164.

[10] Roberts, G. O. and Tweedie, R. L. (1996). Exponential convergence of Langevin distributions and their discrete approximations. *Bernoulli* **2,** 341–363.

[11] Ross, S. M. (2007). *Introduction to Probability Models*. Academic Press.

[12] Schurz, H. (2007). Modeling, analysis and discretization of stochastic logistic equations. *Internat. J. Numer. Anal. Modeling* **4,** 178–197.