Applied Probability Trust (3 February 2013)

EXACT SIMULATION FOR DIFFUSION BRIDGES – AN ADAPTIVE APPROACH

HONGSHENG DAI,* *Department of Mathematics, School of CEM, University of Brighton, UK*

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 due to small acceptance probabilities. This paper proposes an adaptive approach which 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 re-weighted 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 bridges; Exact Monte Carlo simulation

2010 Mathematics Subject Classification: Primary 65C05;65C50

Secondary 60J60

1. Introduction

This paper considers the diffusion bridge $\boldsymbol{X} = \{X_t, t \in [0,T]\}$, given by

$$dX_t = \alpha(X_t)dt + dB_t, X_0 = x, X_T = y, \tag{1}$$

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

* Postal address: Department of Mathematical Sciences, University of Essex Wivenhoe Park, Colchester CO4 3SQ, UK; hdaia@essex.ac.uk

[1], [2], [3], [4] and [5] have proposed several novel methods for 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* $\boldsymbol{l}$ *which does not depend on* $\boldsymbol{\omega}$.
(c): $\exp\left\{\int_0^T \alpha(\omega_s)d\omega_s - \int_0^T \alpha^2(\omega_s)ds/2\right\}$ *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 appropriate probability density ratio ([3]), which is derived using Girsanov's transformation formula ([7, 8]) if applicable.

In practice, their methods may not be efficient due to low acceptance probability, if (1) the time gap T is large; or (2) the values of $(\alpha^2+\alpha')(\omega_s)/2 - \boldsymbol{l}$ are very large throughout the interval $[0,T]$ (see Example 1 in Section 2). This paper focuses on dealing with the challenge under case (2) 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 $(\alpha^2+\alpha')(\omega_s)$ and it increases the acceptance probability significantly. We leave tackling the challenge of case (1) in 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 re-weighted layer probabilities of a Brownian bridge. Then we simulate a Brownian bridge conditional on the simulated layer. By doing so, 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 the new method is more efficient than existing methods under certain scenarios.

We begin from Section 2 by stating an example 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 do rejection sampling for the diffusion bridges based on the proposal process. We will provide simulations studies to demonstrate the performance of the new method in Section 5 and provide a discussion in Section 6.

2. Rejection sampling and its challenges

Consider the diffusion model (1). Following [3] and using Girsanov's transformation formula we have

$$\frac{d\mathbb{Q}^{x,y}_{0,T}}{d\mathbb{W}^{x,y}_{0,T}}(\boldsymbol{\omega}) \propto \exp\left\{-\int_0^T \frac{1}{2}(\alpha^2+\alpha')(\omega_s)ds\right\}. \tag{2}$$

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

$$\frac{d\mathbb{Q}^{x,y}_{0,T}}{d\mathbb{W}^{x,y}_{0,T}}(\boldsymbol{\omega}) \propto \exp\left\{-\int_0^T \left[\frac{1}{2}(\alpha^2+\alpha')(\omega_s)-\boldsymbol{l}\right]ds\right\}, \tag{3}$$

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

The acceptance probability, given by

$$E_{\mathbb{W}}\left[\exp\left\{-\int_0^T \left[\frac{1}{2}(\alpha^2+\alpha')(\omega_s)-\boldsymbol{l}\right]ds\right\}\right], \tag{4}$$

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

Example 1.

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

$$dV_s = RV_s(1-V_s/\Lambda)ds + \sigma V_s dB_s.$$

More details about the logistic growth diffusions and their applications can be found in [12]. To simulate $\{V_s\}$, we only need to simulate the transformed diffusion $X_s = -\log(V_s)/\sigma$, which, following [3], solves

$$\begin{aligned} dX_s &= \alpha(X_s)ds + dB_s, \\ \alpha(u) &= \frac{\sigma}{2} - \frac{R}{\sigma} + \frac{R}{\sigma\Lambda}\exp(-\sigma u). \end{aligned}$$

It can be shown that

$$\frac{\alpha^2(u)+\alpha'(u)}{2} = \frac{\sigma^2}{8} - \frac{R}{2} + \frac{R^2}{2\sigma^2}\left[1-\frac{\exp(-\sigma u)}{\Lambda}\right]^2. \tag{5}$$

[3] used the maximum lower bound $\boldsymbol{l} = \sigma^2/8 - R/2$ for (5). However, $[\alpha^2(\omega_s) + \alpha'(\omega_s)]/2$ may be much larger than $\boldsymbol{l}$, if the third term of (5), $[R^2/2\sigma^2]\cdot[1-\exp(-\sigma\omega_s)/\Lambda]^2$, 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 (5) will be much larger than $\boldsymbol{l}$. 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 [4] we can easily find that the Brownian bridge $\{B_t, t \in [0,2]\}$ only has a small probability (less than 0.02) to hit the boundaries -65.51 ± 2. This result implies the acceptance probability

$$\begin{aligned} & E_{\mathbb{W}}\left[\exp\left\{-\int_0^T \left[\frac{1}{2}(\alpha^2+\alpha')(\omega_s) - \boldsymbol{l}\right] ds\right\}\right] \\ \leq\ & E_{\mathbb{W}}\left\{\exp\left(-\int_0^T [R^2/2\sigma^2]\cdot[1-\exp(-\sigma\omega_s)/\Lambda]^2\, ds\right) I[\omega_s \in [-67.51, -63.51]]\right\} + 0.02 \\ \leq\ & \exp(-25\,[1-\exp(6.751)/\Lambda]^2) + 0.02 \approx 0.03. \end{aligned}$$

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 $\boldsymbol{l}$, the probability for that can be very tiny and there is a very large probability that $\frac{1}{2}(\alpha^2+\alpha')(\omega_s)$ is much bigger than $\boldsymbol{l}$. Therefore many simulated proposal process $\boldsymbol{\omega}$ 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 re-weighted layer probabilities of a Brownian bridge. Then we simulate a Brownian bridge conditional on the simulated layer. So we first introduce the *layers* by following

the notations in [4].

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

$$\begin{aligned}
\mathcal{U}_i(x, y; 0, T) &= \left\{\boldsymbol{\omega} : \sup_{0\leq s\leq T} \omega_s \in [\bar{y} + a_{i-1}, \bar{y} + a_i]\right\} \cap \left\{\boldsymbol{\omega} : \inf_{0\leq s\leq T} \omega_s > \bar{x} - a_i\right\}, \\
\mathcal{L}_i(x, y; 0, T) &= \left\{\boldsymbol{\omega} : \inf_{0\leq s\leq T} \omega_s \in [\bar{x} - a_i, \bar{x} - a_{i-1}]\right\} \cap \left\{\boldsymbol{\omega} : \sup_{0\leq s\leq T} \omega_s < \bar{y} + a_i\right\},
\end{aligned}$$

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

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

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

$$l_i \leq \inf_{s\in[0,T], \boldsymbol{\omega}\in\mathcal{D}_i} \{(\alpha^2 + \alpha')(\omega_s)/2\}. \tag{6}$$

Obviously such $l_i \geq \boldsymbol{l}$ for all i.

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

$$\widetilde{\mathbb{W}}^{x,y}_{0,T}(\boldsymbol{\omega}) \propto \mathbb{W}^{x,y}_{0,T}(\boldsymbol{\omega}) \sum_{i=1}^{\infty} \exp\{-Tl_i\} I\{\boldsymbol{\omega} \in \mathcal{D}_i\}. \tag{7}$$

We then have

$$\begin{aligned}
\frac{d\mathbb{Q}^{x,y}_{0,T}}{d\widetilde{\mathbb{W}}^{x,y}_{0,T}}(\boldsymbol{\omega}) &\propto \frac{\exp\left\{-\int_0^T \frac{1}{2}(\alpha^2 + \alpha')(\omega_s) ds\right\}}{\sum_{i=1}^{\infty} \exp\{-Tl_i\} I\{\boldsymbol{\omega} \in \mathcal{D}_i\}} \\
&= \sum_{i=1}^{\infty} \exp\left\{-\int_0^T \left[\frac{1}{2}(\alpha^2 + \alpha')(\omega_s) - l_i\right] ds\right\} I\{\boldsymbol{\omega} \in \mathcal{D}_i\},
\end{aligned} \tag{8}$$

which is also a value no more than 1. Therefore if we can simulate from $\widetilde{\mathbb{W}}^{x,y}_{0,T}(\boldsymbol{\omega})$, which will be discussed later, then based on (8) we can also use rejection sampling. The acceptance probability is now given by

$$\sum_{i=1}^{\infty} E_{\mathbb{W}}\left[\exp\left\{-\int_0^T \left[\frac{1}{2}(\alpha^2 + \alpha')(\omega_s) - l_i\right] ds\right\} I\{\boldsymbol{\omega} \in \mathcal{D}_i\}\right], \tag{9}$$

which will be larger than the acceptance probability in (4), since $l_i \geq \boldsymbol{l}$ for all i.

To simulate from $\widetilde{\mathbb{W}}_{0,T}^{x,y}(\boldsymbol{\omega})$ we write (7) as

$$\begin{aligned}\widetilde{\mathbb{W}}_{0,T}^{x,y}(\boldsymbol{\omega}) &= \frac{\mathbb{W}_{0,T}^{x,y}(\boldsymbol{\omega})\sum_{i=1}^{\infty}\exp\{-Tl_i\}I\{\boldsymbol{\omega}\in\mathcal{D}_i\}}{\sum_{i=1}^{\infty}\exp\{-Tl_i\}\mathbb{W}_{0,T}^{x,y}\{\boldsymbol{\omega}\in\mathcal{D}_i\}} \\ &= \sum_{i=1}^{\infty}\left\{\frac{\exp\{-Tl_i\}\mathbb{W}_{0,T}^{x,y}\{\boldsymbol{\omega}\in\mathcal{D}_i\}}{\sum_{i=1}^{\infty}\exp\{-Tl_i\}\mathbb{W}_{0,T}^{x,y}\{\boldsymbol{\omega}\in\mathcal{D}_i\}}\frac{\mathbb{W}_{0,T}^{x,y}(\boldsymbol{\omega})I\{\boldsymbol{\omega}\in\mathcal{D}_i\}}{\mathbb{W}_{0,T}^{x,y}\{\boldsymbol{\omega}\in\mathcal{D}_i\}}\right\}\end{aligned} \tag{10}$$

where $\mathbb{W}_{0,T}^{x,y}\{\boldsymbol{\omega}\in\mathcal{D}_i\}$ is the probability that the Brownian bridge belongs to the event $\mathcal{D}_i$ (in layer i). To simulate from (10), we can first simulate the layer, say I, according to the probability

$$\widetilde{\mathbf{P}}(I=i)=\frac{\exp\{-Tl_i\}\mathbb{W}_{0,T}^{x,y}\{\boldsymbol{\omega}\in\mathcal{D}_i\}}{\sum_{i=1}^{\infty}\exp\{-Tl_i\}\mathbb{W}_{0,T}^{x,y}\{\boldsymbol{\omega}\in\mathcal{D}_i\}} \tag{11}$$

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

3.1. Simulation for the layer I from (11)

3.1.1. **Preliminaries.** We can write (11) as

$$\widetilde{\mathbf{P}}(I=i)\propto b_i\gamma_i, \tag{12}$$

where

$$b_i=\exp(-T(l_i-\boldsymbol{l})) \tag{13}$$

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

$$0<S_{i,2}<S_{i,4}<S_{i,6}<\cdots<F_i<\cdots<S_{i,5}<S_{i,3}<S_{i,1}, \tag{14}$$

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 [4]. 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 for $U\ge S_{i,j}$ for an odd number of j, by searching from $j=1$ to ∞.

This is given by the following Algorithm 1, the subroutine $\mathbf{SA}(\{S_{i,j}\}, U)$, where the inputs are the alternating sequence and a standard uniform variable U. It will output an indicator $\mathcal{I}_i$ with $P(\mathcal{I}_i = 1) = F_i$ and $P(\mathcal{I}_i = 0) = 1 - F_i$ and output a value τ_i such that $U \leq \tau_i < F_i$ if $\mathcal{I}_i = 1$ and $U \geq \tau_i > F_i$ if $\mathcal{I}_i = 0$.

Input: Alternating sequence $\{S_{i,j}\}$, which satisfies (14) and a standard uniform variable U

Output: Indicator $\mathcal{I}_i$ and a value τ_i

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

Algorithm 1: $\mathbf{SA}(\{S_{i,j}\}, U)$; Simulate indicator $\mathcal{I}_i$ with $P(\mathcal{I}_i = 1) = F_i$ and $P(\mathcal{I}_i = 0) = 1 - F_i$. If $\mathcal{I}_i = 1$, $U \leq \tau_i < F_i$; if $\mathcal{I}_i = 0$, $U > \tau_i \geq F_i$

See [4] for more details of alternating sequences and how to simulate from $\mathbf{P}(I = i) \propto \gamma_i$.

Simulation of the layer I from (12), however, is not straightforward due to the factor b_i. Note that, if $\boldsymbol{l} = \inf_{s\in[0,T]}\{(\alpha^2+\alpha')(\omega_s)/2\}$ and $l_i = \inf_{s\in[0,T],\boldsymbol{\omega}\in\mathcal{D}_i}\{(\alpha^2+\alpha')(\omega_s)/2\}$ then $\{l_i\}$ is a decreasing sequence and $l_i \to \boldsymbol{l}$. Then from (13) we further have $\{b_i\}$ to be an increasing sequence and $b_i \to 1$. In practice, it is often challenge to find the maximum lower bound $\inf_{s\in[0,T],\boldsymbol{\omega}\in\mathcal{D}_i}\{(\alpha^2+\alpha')(\omega_s)/2\}$ explicitly. We usually consider a sequence $\{l_i\}$ such that $l_i < \inf_{s\in[0,T],\boldsymbol{\omega}\in\mathcal{D}_i}\{(\alpha^2+\alpha')(\omega_s)/2\}$ and its limiting value $\boldsymbol{l} < \inf_{s\in[0,T]}\{(\alpha^2+\alpha')(\omega_s)/2\}$. Although l_i is not the maximum lower bound for $(\alpha^2+\alpha')(\omega_s)/2$, we can always choose $\{l_i\}$ to be decreasing and converges to $\boldsymbol{l}$. Then $\{b_i\}$ is increasing and converges to 1. Therefore, in this section we will show how to simulate the layer I from (12) with $\{b_i\}$ as an increasing sequence.

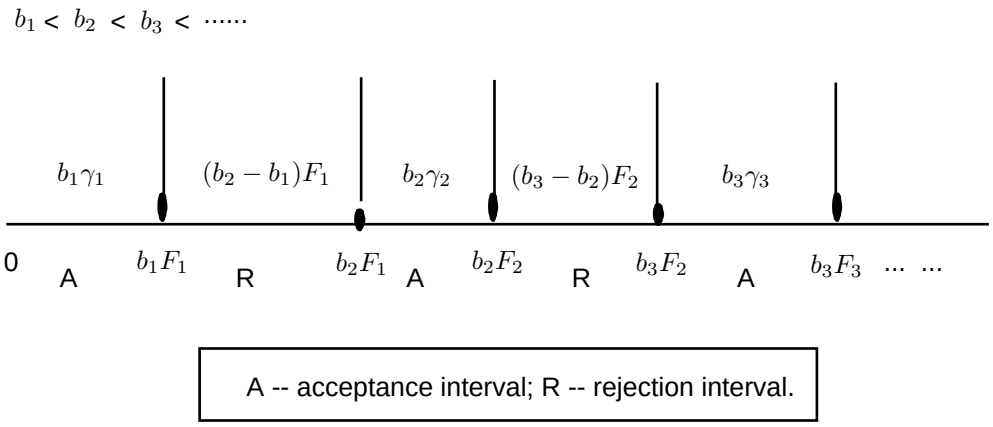

FIGURE 1: Naive rejection sampling when b_i is increasing, where intervals with letter A means the acceptance region for U and intervals with letter R means the rejection region for U.

3.1.2. **Simulation from (12).** First we can partition the interval $[0,1]$ by the ascending sequence

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

as shown in Figure 1.

To simplify the notations, we define $\delta_1 = 0, \{\delta_i = b_iF_{i-1}\}_{i\geq 2}$, $\{\Delta_i = b_iF_i\}_{i\geq 1}$ and $p_i = \Delta_i - \delta_i = b_i\gamma_i$. Then simulating a layer I from $\widetilde{\mathbf{P}}$ in (12) is equivalent to simulating a random value U^* from the mixture of uniform distributions

$$\begin{aligned} f(u) &= \sum_{i=1}^{\infty} p_i \cdot \mathbf{U}\{[\delta_i, \Delta_i]\} / \sum_{i=1}^{\infty} p_i \\ &\propto \sum_{i=1}^{\infty} I\{u \in [\delta_i, \Delta_i]\}, \end{aligned} \tag{15}$$

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

Note that $\sum_{i=1}^{\infty} p_i$ usually cannot be calculated explicitly in practice. So direct sampling from (15) 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 $[\delta_i = b_i F_{i-1}, \Delta_i = b_i F_i]$, which has length $b_i\gamma_i$, then accept U as U^* and output $I = i$. On the contrary, if U is in the rejection interval $(\Delta_{i-1} = b_{i-1}F_{i-1}, \delta_i = b_i F_{i-1})$, 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. We, however, 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 $\mathbf{U}\{[0, 1]\backslash[\Delta_i, \delta_{i+1}]\}$. 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 explains this procedure.

1 Set $\mathcal{R}$ as an empty set ;
2 Simulate U uniformly from the set $[0, 1]\backslash\mathcal{R}$;
3 Find the value i, such that $\delta_i \le U \le \Delta_i$ or $\Delta_i \le U \le \delta_{i+1}$;
`/* This is to find which subinterval (see the partition in Figure 1) that` U `belongs to. */`
4 **if** $\delta_i \le U \le \Delta_i$ **then**
5 | accept $U^* = U$, output $I = i$ as the layer from (12) and stop
6 **else**
7 | set $\mathcal{R} = \mathcal{R} \cup [\Delta_i, \delta_{i+1}]$;
8 **end**
9 Go to step 2;

Algorithm 2: A naive adaptive algorithm.

Lemma 1. *Algorithm 2 simulates U^* from the mixture of uniform distribution (15) and the layer I from (12).*

Proof. The theorem follows easily as it is a rejection sampling with the proposal den-

sity, uniform distribution on set $[0,1]\backslash\mathcal{R}$, and the target density, uniform distribution on set $[0,1]\backslash \cup_i [\Delta_i, \delta_{i-1}]$. □

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

Step 3 of Algorithm 2, checking $\delta_i \leq U \leq \Delta_i$, can be carried out easily. Although we can not 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

$$\begin{aligned} &0 < b_i S_{i,2} < b_i S_{i,4} < \cdots < \Delta_i = b_i F_i < \cdots < b_i S_{i,3} < b_i S_{i,1} \\ &0 < b_i S_{i-1,2} < b_i S_{i-1,4} < \cdots < \delta_i = b_i F_{i-1} < \cdots < b_i S_{i-1,3} < b_i S_{i-1,1}. \end{aligned} \quad (16)$$

Clearly, Algorithm 1 can be used to check $\delta_i \leq U \leq \Delta_i$. We can run Algorithm 1 for both alternating sequences in (16). $\mathbf{SA}(\{b_i S_{i,j}\}, U)$ and $\mathbf{SA}(\{b_i S_{i-1,j}\}, U)$ 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 $\delta_i < \tau_{i,2} \leq U \leq \tau_{i,1} < \Delta_i$; if $\mathcal{I}_{i,1} = 0$ and $\mathcal{I}_{i+1,2} = 1$ then $\Delta_i < \tau_{i,1} \leq U \leq \tau_{i+1,2} < \delta_{i+1}$.

Step 2 of Algorithm 2, however, only works in theory. This is because the proposal density $\mathbf{U}\{[0,1]\backslash\mathcal{R}\}$ is not available, since the evaluation $\mathcal{R} = \mathcal{R} \cup [\Delta_i, \delta_{i+1}]$ in step 7 can not be performed due to not having the explicit values for Δ_i and δ_{i+1}. However, using (16) 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 $\delta_i \leq U \leq \Delta_i$ using Algorithm 1. When U is in the rejection interval $U \in [\Delta_i, \delta_{i+1}]$, we must have $\Delta_i < \tau_{i,1} \leq U \leq \tau_{i+1,2} < \delta_{i+1}$ for some $\tau_{i,1}$ and $\tau_{i+1,2}$ outputed by Algorithm 1.

Therefore, we can have the following practical Algorithm 3, which is equivalent to simulating a proposal U from the hat function

$$g(u) \quad \propto \quad \mathbf{U}\{[0,1]\backslash\mathcal{R}\}, \quad (17)$$

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

1 Set $\mathcal{R}$ as an empty set ;
2 Simulate U uniformly from the set $[0,1]\backslash\mathcal{R}$;
3 Find the value i, such that $\delta_i \le U \le \Delta_i$ or $\Delta_i \le U \le \delta_{i+1}$;
/* $(\mathcal{I}_{i,1}, \tau_{i,1})$ = **SA**$(\{b_i S_{i,j}\}, U)$ and $(\mathcal{I}_{i+1,2}, \tau_{i,2})$ = **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 $U^* = U$, output i as the layer and stop
7 **else**
8 | set $\mathcal{R} = \mathcal{R} \cup [\tau_{i,1}, \tau_{i+1,2}]$;
9 **end**
10 Go to step 2;

Algorithm 3: The practical algorithm.

3.2. Simulation of $\boldsymbol{\omega}$ conditional on $\boldsymbol{I}$

Given the condition that the Brownian bridge is in layer I, we can construct the layered Brownian bridge with the method in [4], which is given by the following Algorithm 4.

Using the results in Section 3.1 and Section 3.2, we can actually simulate the proposal processes from (7). Based on the simulated proposal processes, we introduce the rejection sampling in the following section.

4. Rejection sampling for $\boldsymbol{X}$

The rejection sampling for $\boldsymbol{X}$ 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 $\boldsymbol{\omega} \in \mathcal{D}_I$. We now provide the new exact simulation Algorithm 5.

In the algorithm, 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 thinning algorithm ([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

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

Algorithm 4: Sampling a Brownian bridge conditional on $\boldsymbol{\omega} \in \mathcal{D}_I$ using the method in [4].

$[0,T]$, is equal to the acceptance ratio (8). 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 [3] and [4].

5. Simulation studies

In the previous section, we have shown that Algorithm 5 draws a perfect realization of $\{X_t, t \in [0,T]\}$ driven by (1). We can also use simulation to show that Algorithm 5 provide 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 [4]. 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 [4].

Simulate I from $\widetilde{\mathbf{P}}(I = i)$ using Algorithm 3;
Calculate $l_I = \inf[\alpha^2(u) + \alpha'(u)]/2$, given that $u \in [\bar{x} - a_I, \bar{y} + a_I]$;
Calculate r_I such that $r_I \geq \sup_{t\in[0,T],\boldsymbol{\omega}\in\mathcal{D}_I}\{[\alpha^2(\omega_t) + \alpha'(\omega_t)]/2 - l_I\}$;
Simulate $\boldsymbol{\Psi} = \{\psi_1, \cdots, \psi_\rho\}$ uniformly distributed on $\mathbf{U}[0, T]$ and marks $\Upsilon = \{\nu_1, \cdots, \nu_\rho\}$ uniformly distributed on $\mathbf{U}[0, 1]$, where ρ is from $\text{Poi}(r_I T)$;
Simulate a sample path $\boldsymbol{\omega}$, from $\mathbb{W}_{0,T}^{x,y}$ conditional on $\boldsymbol{\omega} \in D_I$, using Algorithm 4;
Compute the acceptance indicator $\mathcal{I} := \prod_{j=1}^{\rho} I[\phi(\omega_{\psi_j}) < \nu_j]$, with $\phi(\cdot) = r_I^{-1}[(\alpha^2 + \alpha')(\cdot)/2 - l_I]$;
if $\mathcal{I} = 1$ **then**
    accept the $\boldsymbol{\omega}$
else
    return to step 1
end

Algorithm 5: Exact sampling for $\boldsymbol{X}$.

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, (1) $a_i = 0.1i$; (2) $a_i = 0.3i$; (3) $a_i = 0.6i$; (4) $a_i = 1.0i$. The running time and acceptance probabilities are given in Table 1.

New method	$a_i = 0.1i$	$a_i = 0.3i$	$a_i = 0.6i$	$a_i = 1.0i$
running time	99s	17s	7.2s	7.4s
acceptance prob	0.1676	0.1542	0.1309	0.0957
Existing method	$a_i = 0.1i$	$a_i = 0.3i$	$a_i = 0.6i$	$a_i = 1.0i$
running time	1338s	1026s	843s	891s
acceptance prob	0.0008	0.0007	0.0007	0.0008

TABLE 1: Running time comparisons.

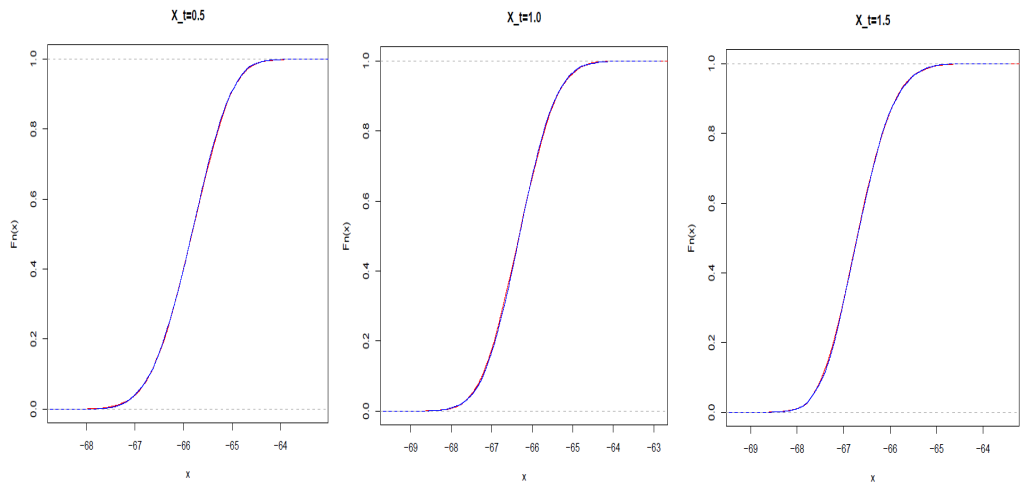

FIGURE 2: The empirical distribution estimates are the same based on the new method and the method in [4].

From the results we can see that the acceptance probability (9) for the new method is overall much larger than the acceptance probability (4) for the existing method. The acceptance probability (4) for existing methods should remain constant for different choice of layer sequence $\{a_i\}$, as the proposal measure $\mathbb{W}_{0,T}^{x,y}$ does not change. On the contrary, the acceptance probability (9) for the new method increases when we choose thinner layers (having smaller values for $a_i - a_{i-1}$). This is consistent to what we expect.

We also notice that the running times of the algorithms increase, when we choose thinner layers. For the method in [4] 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 due to searching the layers with probability $\mathbf{P}(I = i) = \gamma_i$. 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 due to step 1 in Algorithm 5, which uses Algorithm 3 to simulate the layer I with probability $\widetilde{\mathbf{P}}(I = i) \propto b_i\gamma_i$. The probability $\widetilde{\mathbf{P}}$ re-weights $\mathbf{P}$ by multiplying the factor b_i. This re-weighting will result in that a layer with larger value of I is simulated. Step 3 of Algorithm 3 always start searching from $i = 1$ to ∞ and it needs more time if we use thinner layers. Therefore we do not suggest to 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 is also the reason that we only use adaptive approach for sampling the layers of the proposal process (Algorithm 3) with fixed $\{a_i\}$ but we did not use adaptive approach to select from different layer sequences $\{a_i\}$.

5.2. Simulation example 2

To show that the proposed algorithm is correct, we also consider to 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 think about a Langevin diffusion example. The Langevin diffusions are important in practice, as it is the basis for the construction of a variety of MCMC algorithms ([10]) and it could be used to develop new simulation method 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

$$dX_t = \alpha(X_t)dt + d\omega_t,\ \alpha(X_t) = \frac{1}{2}\left.\frac{d\log(\pi(x))}{dx}\right|_{x=X_t}. \tag{18}$$

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 develop new methodologies for simulation from Bayesian posterior of mixture models, for which existing simulation methods (such as Markov chain Monte Carlo) suffers from slow convergence ([6]). The results here will also show that the proposed method works for complicated scenarios.

Suppose that we have observations $\xi_1, \cdots, \xi_n$ 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. Given a prior $\pi_0(x)$ the posterior distribution is given by

$$\pi(x) \propto \prod_{i=1}^{n} \left[q_1 h_1(\xi_i; x) + q_2 h_2(\xi_i)\right] \pi_0(x). \tag{19}$$

For the α given in (18), 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, however, results in tiny acceptance probabilities and make the existing method not working practically.

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

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

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

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

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 dealt by all modern computers). We simulate 10000 samples directly from $\pi(x)$ in (19) and use this as the reference sample. Then we simulate a diffusion process given by (18), using the adaptive method, with X_0 from $\pi(x)$. Note that the diffusion bridge simulation algorithm proposed here can be easily extend to simulate a diffusion process, if we simulate the end point 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 for 10000 times to obtain 10000 realisations. Samples obtained via the adaptive method have almost the same empirical distribution as the reference sample. See Figure 3.

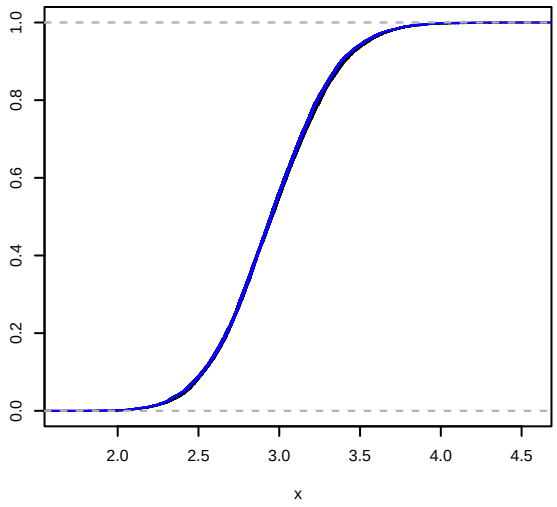

FIGURE 3: The empirical distributions are the same based on the direct sampling and the proposed method

For the above example with $T = 0.1$, it took about 5 hours to simulate 10,000 realisations via the adaptive approach, by choosing layer as $a_i = 0.05i$. However, the existing rejection sampling in [4] does not work practically, not returning a diffusion bridge in the time interval $[0, 0.1]$, within an hour.

6. Discussion

We have provided an adaptive approach to do exact simulation for a class of diffusion bridges. The new methods draw proposals by simulating re-weighted 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 the paper only shows 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 demonstrate that for many complicated problems, we may not obtain the maximum lower bound, but a much smaller lower

bound $\boldsymbol{l} < \inf_s \frac{1}{2}(\alpha^2 + \alpha')(\omega_s)$. For such cases, it is important to use the new method in this paper, otherwise the existing methods may not work practically simply due to $\boldsymbol{l}$ 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 as a future research work.

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

$$E_{\mathbb{W}}\left[\exp\left\{-\int_0^T \left[\frac{1}{2}(\alpha^2 + \alpha')(\omega_s) - \boldsymbol{l}\right] ds\right\}\right]$$

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

Algorithm 3 is 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 multi-modality of $(\alpha^2 + \alpha')(\cdot)$ (Simulation example 2), which makes it is impossible to always guarantee l_i to be decreasing. Therefore if numerical methods are used, $\{b_i\}$ may not be an increasing sequence. Even if $\{b_i\}$ is non-increasing, we can still use an adjusted version of Algorithm 3. We can choose a different partition of $[0, 1]$, $\{\delta_1, \Delta_1, \delta_2, \Delta_2, \cdots\}$ as follows. Suppose that b_{j^*} is the maximum value for all $\{b_j, j = 1, \cdots, i-1\}$. We let $\delta_1 = 0$, $\Delta_1 = b_1 F_1$. For $i \geq 2$ we let $\delta_i = b_i F_{i-1}$ and $\Delta_i = b_i F_i$ when $b_{j^*} \leq b_i$. We let $\delta_i = (b_{j^*} - b_i)F_i + b_i F_{i-1}$ and $\Delta_i = b_{j^*} F_i$. Figure 4 explains how to partition $[0, 1]$ when b_i is not an increasing sequence. The acceptance and rejection intervals are explained in Figure 4. Note that the partition can be found, step by step from $i = 1$ to ∞, when the algorithm is searching which interval the standard uniform random variable U belongs to. Algorithm 3 therefore can be simply adapted for such non-increasing 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.

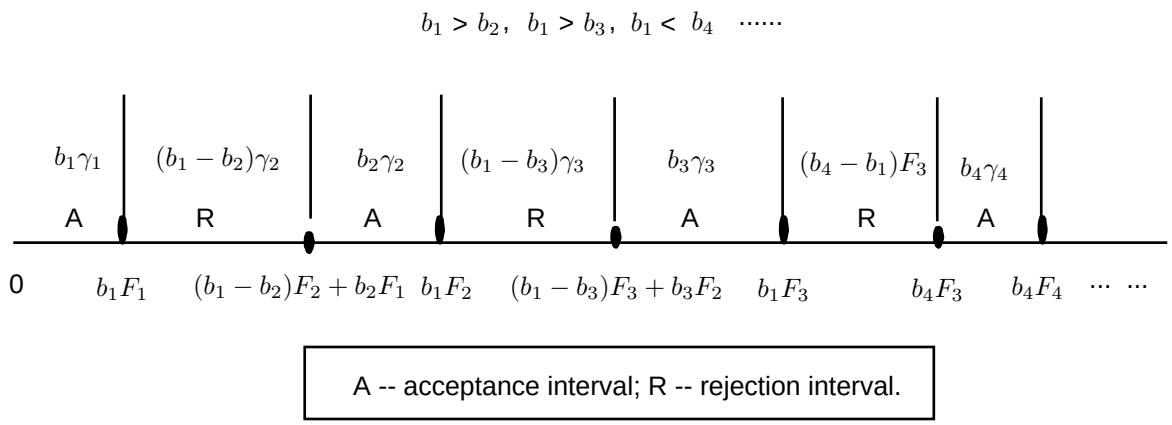

FIGURE 4: partition of $[0,1]$ for a general sequence of b_i.

References

[1] BESKOS A. AND ROBERTS G. O. (2005). Exact simulation on diffusions. *The Annals of Applied Probability* **15**: 2422–2444.

[2] BESKOS A., PAPASPILIOPOULOS O. AND ROBERTS G. O. (2006a). Retrospective exact simulation of diffusion sample paths with applications. *Bernoulli* **12**(6): 1077–1098.

[3] BESKOS A., PAPASPILIOPOULOS O., ROBERTS G. O. AND FEARNHEAD P. (2006b). Exact and computationally efficient likelihood-based estimation for discretely observed diffusion processess. *Journal of Royal Statistical Society, B* **68**: 333–382.

[4] BESKOS A., PAPASPILIOPOULOS O. AND ROBERTS G. O. (2008). A factorisation of diffusion measure and finite sample path constructions. *Methodol Comput Appl Probab* **10**: 85–104.

[5] CASELLA B. AND ROBERTS G. O. (2007). Exact Monte Carlo simulation of killed diffusions. *Advances in Applied Probability* **41**: 273–291.

[6] CELEUX G., HURN M. AND ROBERTS C. P. (2000). Computational and inferential difficulties with mixture posterior distributions. *Journal of the American Statistical Association* **95**: 957–970.

[7] KARATZAS I. AND SHREVE S. E. (1991). *Brownian Motion and Stochastic Calculus*, 2th Edition, Springer-Verlag.

[8] Øksendal B. (1995). *Stochastic Differential equations – An Introduction with Applications*, 4th Edition, Springer-Verlag.

[9] Potzelberger K. and Wang L. (2001). Boundary crossing probability for Brownian motion. *Journal of applied probability* **38**:152-164.

[10] Roberts G. O. and Tweedie R. L. (1996). Exponential convergence of Langevin distributions and their discrete approximations, *Bernoulli* **2**(4):341-363.

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

[12] Schurz H. (2007). Modeling, analysis and discretization of stochastic logistic equations. *International Journal of Numerical Analysis and Modeling*, **4**:178-197.