HYBRIDGE

Distributed Control and Stochastic Analysis of Hybrid Systems
Supporting Safety Critical Real-Time Systems Design

WP8: Accident risk decomposition

Monte Carlo simulation of rare events in hybrid systems

Jaroslav Krystul[1] **and Henk Blom**[2]

1st July 2004

Version: *0.7*
Task number: *8.3*
Deliverable number: *D8.3*
Contract: *IST-2001-32460 of European Commission*

[1] Twente University, The Netherlands

[2] National Aerospace Laboratory NLR, The Netherlands

DOCUMENT CONTROL SHEET

Title of document: *Accident Risk Assessment Methods*

Authors of document: *J.Krystul, H.Blom*

Deliverable number: *D8.3*

Contract: *IST-2001-32460 of European Commission*

Project: *Distributed Control and Stochastic Analysis of Hybrid Systems Supporting Safety Critical Real-Time Systems Design (HYBRIDGE)*

DOCUMENT CHANGE LOG

Version #	Issue Date	Sections affected	Relevant information
0.1	15 Dec. 2003	All	1st draft
0.2	13 Jan. 2004	All	2nd draft
0.3	08 Feb. 2004	1, 4, 6.	3rd draft
0.4	17 Mar. 2004	3, 4, 5, 6	4th draft
0.5	6 May 2004	All	5th draft
0.6	15 Jun. 2004	2, 3, 4, 5	Final version
0.7	1 Jul. 2004	Reviewers	Corrected

Version 0.7		Organisation	Signature/Date
Authors	Jaroslav Krystul	TWEN	
	Henk Blom	NLR	
Internal reviewers	Pascal Lezaud	CENA	
	Oliver Watkins	UCAM	
	Costanzo Manes	AQUI	
	François LeGland	INRIA	

Contents

1 Introduction

1.1 The objectives of Work Package 8

Within the HYBRIDGE project, the objective of WP8 is to develop novel methods for the decomposition of accident risk such that extreme low risk values can be assessed through a hierarchy of conditional Monte Carlo simulations which are scalable relative to the increasing complexity of the application considered. The research within WP8 is organized in four tasks:

8.1 - Review existing accident risk decomposition and assessment methods, both analytical ones, Monte Carlo simulation approaches and combinations of these two. This review should distinguish between theory based methods and heuristic methods.

8.2 - Develop new accident risk decomposition and assessment methods. One of the key directions to be explored is the development of risk decomposition methods that make use of the fact that for strong Markov processes the Markov property holds true for stopping times.

8.3 - Development of Conditional Monte Carlo simulation techniques for accident risk assessment that make use of the risk decomposition developed in Task 8.2, and comparison of the new approach with the existing ones identified in Task 8.1.

8.4. - Extend the accident risk decomposition approach with a recursive Bayesian estimation approach which enables the updating of the accident risk assessment while more and new information is becoming available.

The results of task 8.1 and 8.2 have been reported in [Krystul et al 03a] and the intermediate report [Lezaud et al 03] respectively. The current report addresses the consolidation of the results obtained during the performance of Task 8.2 and Task 8.3.

1.2 Accident risk assessment

Accident risk assessment has been widely studied for various safety-critical operations, such as the nuclear and chemical industries, advanced air traffic management (ATM) and many others. All accident risk assessment approaches can be subdivided into two groups: approaches based on statistical analysis of collected data and those that are based on the modelling of the processes leading to the accident. The statistical analysis of extreme values needs a long observation time due to very low probability of the events considered. Statistical approaches are based on the standard extreme value distributions as the Gumbel, Fréchet and Weibull laws [Reiss & Thomas 97]. The modelling approaches firstly consist in formalizing the system considered and secondly by using mathematical or simulation tools obtaining some estimate. The aim of work package WP8 is to study

the second group of accident risk assessment methods and to develop novel methods for the obtaining of accurate estimates of rare event probabilities.

We distinguish three different types of modelling approaches: analytical, numerical and simulation techniques. Analytical, numerical and Monte Carlo simulation speed-up techniques have been studied in Tasks 2.2, 8.1 and 8.2 of the work packages WP2 and WP8 ([Blom et al 03a], [Blom et al 03b], [Krystul et al 03a] and [Lezaud et al 03]). Analytical and numerical approaches appeared to be useful, but can require many simplifying assumptions. This causes Monte Carlo simulation to be a practical alternative when the analysis calls for fewer simplifying assumptions. However, obtaining accurate estimates of rare event probabilities, say about 10^{-9} to 10^{-12}, is not realistic using straightforward Monte Carlo simulation. An overview of complementary and novel techniques for speeding up Monte Carlo simulation has been given in [Lezaud et al 03]. Of these, the most promising are particle systems, importance sampling (IS) and importance splitting. Particle system based Monte Carlo has the advantage over traditional Monte Carlo simulation that the shape of the probability density is available at every moment of the simulation. Importance sampling is based on the notion of modifying underlying probability distribution in such a way that the rare events occur much more frequently. But to use importance sampling, we need to have a good prior knowledge of the applicable underlying probability distribution. And, even in such a case, importance sampling may not provide any speed-up. An alternative way to increase the relative number of visits to the rare event is to use trajectory splitting, based on the idea that there exist some well identifiable intermediate system states that are visited much more often than the target states themselves and behave as gateway states to reach the target states [Townsend et al 98]. For strong Markov diffusion the importance splitting idea has been combined with interacting particle systems approximations and Feynman-Kac distributions [Cerou et al 02]. These particular techniques will now be further developed for switching diffusion that appear in accident risk assessment.

1.3 Problem description

The objective of this report is to develop efficient rare event Monte Carlo algorithm for the case when the evolution dynamics of a particular safety critical operation is described by a coupled and interacting discrete event models and stochastic differential equations. The time evolution of such models is often represented by hybrid state Markov processes. For example, for accident risk modeling in ATM we describe the aircraft evolution by a hybrid state Markov process composed of two components: a continuous component and a discrete component. The continuous component is required to model uncertain motion of aircraft due to, for example, variations in wind, while the discrete component is required

to model switches in aircraft operational modes, for example, switches from a nominal behavior to a non-nominal due to hardware failures. In particular, we will consider the switching diffusion model where switching rates of the discrete component may depend on continuous component [Ghosh93]. We define the rare event as the entrance of continuous component into some target set.

The interacting particles system (IPS) approximation approach of [Cerou et al 02] is very general, and theoretically it can be applied to any strong Markov process. However, the IPS approximation approach will not suffice in obtaining useful estimates when applied to switching diffusion process with rare switchings. It may be unlikely that there is even one switch during a straightforward simulation. In such case, the interactions between system modes are not taken into account and this badly affects estimator performance. To improve this situation we will combine the IPS approximation approach with an IS method. The role of the IPS is to increase the relative number of visits of the continuous component to the rare set. The role of the IS is to force switchings of discrete component. This would induce a bias, but introduction of appropriate weights can ensure that the expected value of the resulting estimator is unchanged.

Stochastic processes are commonly simulated through discrete-time approximations. Numerical schemes for diffusions are well explained in the literature [Kloeden & Platen 92], but not for general switching diffusion processes. To run simulations we will use an Euler-type discrete-time approximation for the switching diffusion model. The question of convergence of the discretized problem to the original continuous-time problem was studied in [Krystul & Bagchi 03]. There it is shown that the distribution of first exit time of discretized model (diffusion and switching diffusion) weakly converges to that of continuous time model as the discretization step tends to zero.

The report is organized as follows. Section 2 gives a brief overview and classification of techniques used in Monte Carlo simulations. Section 3 presents an interacting particle system algorithms for estimating a class of rare events in case of a diffusion process. A numerical example is given. Section 4 deals with extensions to switching diffusion process. Several versions of interacting particle system algorithms for switching diffusion processes are proposed. Numerical evaluation and comparison of these algorithms is described in section 5. Section 6 draws conclusions.

2 Introduction to Monte Carlo simulation techniques

In WP8.1 and WP8.2 several existing Monte Carlo simulation methods for rare event estimation have been studied. Of these, the most suitable methods for rare event estimation are: importance sampling, sequential importance sampling, importance splitting, restart method and interacting particle system approach. Below, we present a brief introduction to the basic techniques used in Monte Carlo simulation approaches. In Table 1, it is shown which of these basic techniques are used by the different methods studied within WP8.

Importance sampling
Assume that we want to estimate $\mathbb{E}[f(x_t)]$ and suppose the underlying probability density of random variable x_t is $p_{x_t}(x)$. Then main idea of importance sampling method is to use another probability density $q_{x_t}(x)$. When trying to estimate the mean value for an arbitrary function $f(x_t)$, we have

$$\mathbb{E}[f(x_t)] = \int f(x)p_{x_t}(x)dx = \int f(x)\frac{p_{x_t}(x)}{q_{x_t}(x)}q_{x_t}(x)dx.$$

Hence,

$$\mathbb{E}[f(x_t)] = \mathbb{E}[\frac{f(y)p_{x_t}(y)}{q_{x_t}(y)}]$$

where the random variable y has the probability density $q_{x_t}(y)$. We therefore have the method of estimating $\mathbb{E}[f(x_t)]$ by using N trials of y, $y_1, \ldots, y_N$ and by approximating $\mathbb{E}[f(x_t)]$ by

$$\frac{1}{N}\sum_{i=1}^{N}\frac{f(y_i)p_{x_t}(y_i)}{q_{x_t}(y_i)}.$$

Good choice of $q_{x_t}(x)$ can produce an estimate with far lower variance. The main drawback of this method is that it requires a good choice of $q_{x_t}(x)$, and thus a thorough analysis of the problem under consideration.

Control Variables
We write $\mathbb{E}[f(x_t)]$ in form

$$\mathbb{E}[f(x_t)] = \mathbb{E}[f(x_t) - q(x_t)] + \mathbb{E}[q(x_t)],$$

Table 1. Basic techniques used by Monte Carlo simulation methods studied within Hybridge WP2 and WP8

Distinguishing features	Monte Carlo approach							
	A	B	C	D	E.1	E.2	F	G
	D8.2 §3.1	D8.2 §3.2	D8.2 §4.1	D8.2 §4.2	D8.2 §5.2	D8.2 §5.2	D2.2	[Blom& Bloem03]
Applicability to Hybrid processes	–	Yes	–	–	–	–	Yes	Yes
Importance sampling	Yes	Yes	–	–	–	–	–	–
Splitting	–	–	Yes	Yes		Yes	–	–
Multi-Level crossing	–	–	Yes	Yes	Yes	Yes	–	–
Stopping time based decomposition	–	–	–	–	Yes*	Yes*	Yes*	–
N particles	–	Yes	–	–	Yes	Yes	–	Yes
Control variables	–	–	–	–	–	–	–	–
Resampling	–	Yes	–	–	Yes	–	–	Yes
Observation filtering	–	–	–	–	Yes	Yes	–	Yes
Objective	Rare event	Rare event	Rare event	Rare event	Rare event	Rare event	Rare event	Bayesian filtering

*) applicable only for strong Markov processes

D8.2=[Lezaud et al 03]
D2.2=[Blom et al 03a]

where $\mathbb{E}[q(x_t)]$ can be evaluated analytically and $Var[f(x_t) - q(x_t)]$ is appreciably smaller than $Var[f(x_t)]$. We then use a Monte Carlo method to evaluate $\mathbb{E}[f(x_t) - q(x_t)]$.

Conditioning
Suppose that we want to estimate

$$\mathbb{E}[f(x_t, y_t)] = \int f(u, v) p_{x_t, y_t}(u, v) du dv,$$

where $p_{x_t,y_t}(u, v)$ is the density function of the pair (x, y). If we set:

$$q(x) = \frac{1}{m(x)} \int f(x, v) p_{x_t, y_t}(x, v) dv = \mathbb{E}[f(x_t, y_t)|x],$$

with $m(x) = \int p_{x_t,y_t}(x, v) dv$, it is easy to see that $\mathbb{E}[f(x_t, y_t)] = \mathbb{E}[q(x_t)]$. In effect, the distribution of x_t is $m(x)dx$, and therefore,

$$\mathbb{E}[q(x_t)] = \int m(u) q(u) du = \int du \int f(x, y) p_{x_t, y_t}(u, v) dv = \mathbb{E}[f(x_t, y_t)].$$

One can prove that

$$Var(q(x_t)) \leq Var(f(x_t, y_t)).$$

If we can explicitly evaluate the function $q(\cdot)$, it is preferable to use a Monte Carlo simulation for $q(x_t)$.

Multi-Level crossing
Suppose the target rare set D is contained within a sequence of nested subsets of the state space S, i.e.:

$$D = D_m \subset \cdots \subset D_0 = S. \tag{1}$$

We denote by $L_i = \partial D_i$ the boundary of the set D_i. If the initial state of process starts in $D_0 \backslash D_1$, then the nesting (1) implies that the rare set can only be reached through crossing all the boundaries (levels) L_i $(i = 1, \ldots, m)$.

Stopping time based decomposition

The original problem is decomposed into conditional problem. It is done by introduction of a conditioning on an event and the moment that event happens (the stopping time). For example, suppose that $\{x_t\}$ is a strong time-homogeneous Markov process, then for any stopping time τ the following equation holds:

$$\begin{aligned}
\mathbb{E}[f(x_{\tau+t})] &= \int f(x)\mathbb{P}_{x_{\tau+t}}(x \in dx) = \int f(x)p_{x_{\tau+t}}(x)dx \\
&= \int\int f(x)p_{x_{t+\tau}|x_\tau}(x|y)p_{x_\tau}(y)dxdy \\
&= \int \mathbb{E}[f(x_{\tau+t})|x_\tau = y]p_{x_\tau}(y)dy \\
&= \int \mathbb{E}[f(x_t)|x_0 = y]p_{x_\tau}(y)dy
\end{aligned}$$

here we have assumed that a conditional density $p_{x_t|x_0}(x|y)$ exists and is uniquely characterized for all $t > 0$, x and y.

N-particles
Let x_t denote the state of the system. The N-particles Monte Carlo simulation consists of approximating the density $p_{x_t}(x)$ by a large set of N particles $\{x_t^i, \omega_t^i\}_{i=1}^N$, where each particle has an assigned relative weight, ω_t^i, such that all weights sum to unity. The density $p_{x_t}(x)$ can be approximated by the empirical distribution:

$$p_{x_t}(x)dx \approx \sum_{i=1}^{N} \omega_t^i \delta_{x_t^i}(dx),$$

where δ denotes the Dirac measure, $\delta_x(B) = 1$ if $x \in B$ and 0 otherwise.

Resampling
During resampling N particles are independently drawn from the empirical distribution generated by the original set of particles. Resampling is performed to reduce the number of particles with almost zero weight, in favour of the particles with significant weight.

Splitting
Entering some intermediate state, which is usually characterized by crossing a threshold (level) by a control parameter, triggers the splitting of the trajectory. The current system state is saved and a number of independent subtrajectories are simulated from the state.

The main drawback is that it is difficult to find the optimal splitting parameters (splitting level and number of splits).

Observation filtering
Let $x_t \in \mathbb{R}^n$ denote the state of the observed system and y_t be the observation at time t, then the filtering density $p_{x_t|y_t}(x|y)$ satisfies Bayes theorem:

$$p_{x_t|y_t}(x|y) = \frac{p_{y_t|x_t}(y|x)p_{x_t}(x)}{p_{y_t}(y)}.$$

If the density function $p_{x_t|y_t}(x|y)$ is approximated by the empirical density of set of N particles $\{x_t^i, \omega_t^i\}_{i=1}^N$, where each particle has an assigned relative weight, ω_t^i, such that all weights sum to unity, then Bayes theorem updates the weights as follows:

$$\widehat{\omega}_t^i = \omega_t^i \cdot \frac{p_{y_t|x_t^i}(y|x)}{c_t}$$

where c_t is a normalizing constant.

3 Rare event Monte Carlo simulation for diffusion

3.1 Problem considered

Throughout this and the next sections, all stochastic processes taking values in some metric state space (S, d) are defined on the canonical filtered probability space $(\Omega, \mathcal{F}, \mathbb{F}, \mathbb{P})$ of right continuous and left limited paths $\Omega = \mathbb{D}([0, \infty), S)$ from $[0, \infty)$ into S. The filtration $\mathbb{F}$ is assumed to be right-continuous and complete.

We consider a multidimensional diffusion process which is assumed to start in some Borel set $D_0 \subset \mathbb{R}^n$ with a given initial probability $P_{x_0}(\cdot)$:

$$dx_t = a(x_t)dt + b(x_t)dW_t, \tag{2}$$

where $a(x) : \mathbb{R}^n \to \mathbb{R}^n$, $b(x) : \mathbb{R}^n \to \mathbb{R}^{n \times n}$ and $\{W_t\}$ is a Wiener process in $\mathbb{R}^n$ independent of x_0. For a given target Borel set $D \subset \mathbb{R}^n$ $(D \cap D_0 = \varnothing)$, we define the first time the process $\{x_t\}$ hits D, namely

$$\tau_D = \inf\{t \geq 0 : x_t \in D\},\ \tau_D = \infty \text{ if this set is empty.}$$

We would like to estimate the quantity

$$P_{hit}(0, T) \stackrel{\triangle}{=} \mathbb{P}(\tau_D \leq T) \tag{3}$$

for some $T < \infty$, i.e. the probability that diffusion $\{x_t\}$ will hit the rare event set D before time T.

Remark 1 *Examples of P_{hit} in air traffic are:*

- *Conflict probability [Hu et al 03] in which D forms the subset in the state space where aircraft are closer to each other than some minimum separation criterion (e.g. 5 Nm in horizontal direction).*
- *Collision probability [Blom & Bloem 03b] in which D is the subset in the state space where aircraft are closer to each other than their physical sizes (of order 100m in horizontal).*

In air traffic, collision probability should be orders of magnitude smaller than conflict probability. As such the prime objective of this report is to address the more rare event of collision.

3.2 Product of conditional probabilities

In [Cerou et al 02] a sequence of gateway states has been used to characterize the rare event probability as a product of conditional probabilities by using Feynman-Kac model in path space. Here we explain how this product form can be obtained for a diffusion. We assume that diffusion (2) starts at $t = 0$ in a Borel set $D_0 \subset \mathbb{R}^n$ with a known initial probability distribution $P_{x_0}(\cdot)$. As in [Cerou et al 02] we assume a sequence of nested Borel sets

$$D = D_m \subset \cdots \subset D_1 \tag{4}$$

where D_k is a closed Borel set of $\mathbb{R}^n$, and D_1 such that $D_1 \cap D_0 = \varnothing$. The first moment that $\{x_t\}$ hits a set D_k is defined as the stopping time:

$$\tau_k \stackrel{\triangle}{=} \inf\{t \geq 0 : x_t \in D_k\},$$

$\tau_k = \infty$ if this set is empty. We want to estimate $\mathbb{P}(\tau_m < T)$, for some $T < \infty$, i.e. the probability that diffusion $\{x_t\}$ will hit the rare event set D before time T. The process $\{x_t\}$, before hitting D, passes through a sequence of nested sets (4). To capture the behavior of $\{x_t\}$ between the first hittings, we introduce the stochastic sequence $\{\varkappa_k\}$ representing the paths of $\{x_t\}$ between the successive hittings.

$$\varkappa_k = \{x_t : \tau_{k-1} \wedge T \leq t \leq \tau_k \wedge T\},\ \varkappa_0 = x_0.$$

If $T < \tau_{k-1}$, then $\varkappa_k = (x_T)$ and $x_{\tau_k \wedge T} = x_T \notin D_k$. Similarly, if $\tau_{n-1} \leq T < \tau_k$, then $\varkappa_k = \{(x_t), \tau_{k-1} \leq t \leq T\}$ and $x_{\tau_k \wedge T} = x_T \notin D_k$. Whereas, if $\tau_k \leq T$, then $\varkappa_k = \{(x_t) : \tau_{k-1} \leq t \leq \tau_k\}$ represents the path of $\{x_t\}$ between the successive sets D_{k-1} and D_k, and $x_{\tau_k \wedge T} = x_{\tau_k} \in D_k$.

By the strong Markov property of $\{x_t\}$ the stochastic sequence $\{\varkappa_k\}$ forms a Markov process [Cerou et al 02] with value in the Borel path-space $(E, \mathcal{E})$ of $\{x_t\}$. Following [Cerou et al 02] we also introduce the $\{0, 1\}$-valued variables $\{y_k, k = 1, \ldots, m\}$ defined as follows:

$$y_k(\omega) \stackrel{\triangle}{=} \mathbf{1}_{\{\omega : x_{\tau_k \wedge T}(\omega) \in D_k\}} = \mathbf{1}_{\{\omega : \tau_k(\omega) \leq T\}}, \quad \omega \in \Omega. \tag{5}$$

Hence, for each k we have

$$y_k(\omega) = \mathbf{1}_{\{\omega : \tau_k(\omega) \leq T\}} = \prod_{i=1}^{k} \mathbf{1}_{\{\omega :\ \tau_i(\omega) \leq T\}} = \prod_{i=1}^{k} y_i(\omega). \tag{6}$$

Next we characterize $P_{hit}(0, T)$ in terms of the sequence $\{y_k\}$. By its definition,

$$P_{hit}(0, T) = \mathbb{P}(\tau_m \leq T) = \mathbb{E}[\mathbf{1}_{\{\tau_m \leq T\}}]$$

Subsequent substitution of (5) and (6) yields:

$$P_{hit}(0,T) = \mathbb{E}[y_m] = \mathbb{E}[\prod_{k=1}^{m} y_k]. \tag{7}$$

Since y_k assumes values from $\{0,1\}$,

$$\mathbb{E}[\prod_{k=1}^{m} y_k] = \prod_{k=1}^{m} \mathbb{E}[y_k | y_{k-1} = 1, \ldots, y_1 = 1]$$

Substituting this into (7) yields

$$\begin{aligned} P_{hit}(0,T) &= \prod_{k=1}^{m} \mathbb{E}[y_k | y_{k-1} = 1, \ldots, y_1 = 1] \\ &= \prod_{k=1}^{m} \mathbb{P}(\tau_k \leq T | \tau_{k-1} \leq T, \ldots, \tau_1 \leq T) \\ &= \prod_{k=1}^{m} \mathbb{P}(\tau_k \leq T | \tau_{k-1} \leq T) \end{aligned} \tag{8}$$

This means that (8) characterizes the probability $P_{hit}(0,T)$ of the rare event as a product of conditional probabilities of intermediate "less rare" events leading to it. Thus, if we estimate the conditional probabilities

$$\gamma_k \stackrel{\triangle}{=} \mathbb{P}(\tau_k \leq T | \tau_{k-1} \leq T) \text{ for } k = 1, \ldots, m$$

and insert this in (8) then we get for $P_{hit}(0,T)$:

$$P_{hit}(0,T) = \prod_{k=1}^{m} \gamma_k \tag{9}$$

The estimation of the probabilities γ_k is in the sequel studied as a nonlinear filtering problem.

3.3 Particle filter approach of [Cerou et al 02]

The process $\{x_t\}$ is a strong Markov process. Because of this, the filtering problem considered is to evaluate the conditional distribution of $\xi_k \stackrel{\triangle}{=} x_{\tau_k \wedge T}$ given $y_{1:k}$, which we

denote as $P_{\xi_k|y_{1:k}}(B)$ for $B \in \mathcal{E}'$, i.e. the Borel σ-algebra of $E' = \mathbb{R}^n$. Because $\{\varkappa_k\}$ is a strong Markov process, $\{\xi_k\}$ is an $\mathbb{R}^n$-valued Markov sequence, and filtering for $\{\xi_k\}$ is accomplished through the following sequence of transformations

$$P_{\xi_{k-1}|y_{1:k-1}}(\cdot) \xrightarrow{\text{prediction}} P_{\xi_k|y_{1:k-1}}(\cdot) \xrightarrow{\text{measurement}} P_{\xi_k|y_{1:k}}(\cdot).$$

with the prediction satisfying:

$$P_{\xi_k|y_{1:k-1}}(B) = \int_{E'} P_{\xi_k|\xi_{k-1}}(B|\xi) P_{\xi_{k-1}|y_{1:k-1}}(d\xi),\ B \in \mathcal{E}',$$

and the measurement update:

$$\frac{dP_{\xi_k|y_{1:k}}}{dP_{\xi_k|y_{1:k-1}}}(\xi) = \frac{\mathbf{1}_{\{\xi \in D_k\}}}{\int_{E'} \mathbf{1}_{\{\xi' \in D_k\}} P_{\xi_k|y_{1:k-1}}(d\xi')}$$

where $\frac{dP_{\xi_k|y_{1:k}}}{dP_{\xi_k|y_{1:k-1}}}$ is the Radon-Nikodym derivative and $P_{\xi_k|\xi_{k-1}}$ is the transition measure of the Markov sequence $\{\xi_k\}$. Next we introduce the notation

$$\begin{aligned} p_k(B) &\triangleq P_{\xi_k|y_{1:k-1}}(B|1\ldots1), \\ \pi_k(B) &\triangleq P_{\xi_k|y_{1:k}}(B|1\ldots1). \end{aligned}$$

This yields

$$\mathbb{E}[y_k|y_{1:k-1} = (1,1,\ldots,1)] = \int_{E'} \mathbf{1}_{\{\xi \in D_k\}} p_k(d\xi).$$

Similarly as in (8), this can be written as:

$$\gamma_k = P(\tau_k \le T | \tau_{k-1} \le T) = \int_{E'} \mathbf{1}_{\{\xi \in D_k\}} p_k(d\xi). \tag{10}$$

With this each of the m terms γ_k in (9) is characterized as a solution of a sequence of filtering problems.

We denote by $\gamma_k^{N_p}$, $p_k^{N_p}$ and $\pi_k^{N_p}$ the numerical approximations of γ_k, p_k and π_k respectively. At $t = 0$ we start with an empirical measure of the form

$$\pi_0^{N_p} \triangleq \sum_{i=1}^{N_p} \omega_0^i \delta_{\{\xi_0^i\}},$$

where $\omega_0^i = 1/N_p$, and ξ_0^i are independent samples from initial distribution $P_{\xi_0}(\cdot)$. Hence π_0^{1,N_p} is defined by a system of N_p particles $\{t_0^i, \xi_0^i, \omega_0^i\}_{i=1}^{N_p}$, where each i-th particles is a

triplet consisting of current time, state and weight. From $t = 0$ to τ_1, and from τ_{k-1} to τ_k each particle evolves stochastically according to equation (2) with initial condition (t^i_{k-1}, ξ^i_{k-1}) (prediction step) until it reaches the next set D_k or the final time T. Let $\{\hat{t}^i_k, \hat{\xi}^i_k, \omega^i_{k-1}\}_{i=1}^{N_p}$ denote the values of the particles after the k-th prediction. Then the empirical distribution $p_k^{N_p}$ associated with the predicted cloud of particles is:

$$p_k^{N_p} = \sum_{i=1}^{N_p} \omega^i_{k-1} \delta_{\{\hat{\xi}^i_k\}}.$$

The particles which do not reach the set D_k before time T are deleted, i.e. we set $\hat{\omega}^i_k = 0$ if $\hat{t}^i_k = T$, else we set $\hat{\omega}^i_k = \omega^i_{k-1}$. If all particles become deleted, the algorithm is stopped and $\gamma_m^{N_p} = 0$. Otherwise

$$\gamma_k \approx \gamma_k^{N_p} = \sum_{i=1}^{N_p} \hat{\omega}^i_k.$$

The empirical distribution $\pi_k^{N_p}$ associated with the measurement updated cloud of particles is:

$$\pi_k^{N_p} = \sum_{i=1}^{N_p} \tilde{\omega}^i_k \delta_{\{\tilde{\xi}^i_k\}}. \tag{11}$$

with

$$\tilde{\omega}^i_k = \frac{\hat{\omega}^i_k}{\sum_{j=1}^{N_p} \hat{\omega}^j_k}, \ \tilde{\xi}^i_k = \hat{\xi}^i_k, \quad i = 1, \ldots, N_p,$$

Particles having reached the set D_k are used for a resampling with replacement of empirical measure (11) step. After this step we again have N_p particles $\{t^i_k, \xi^i_k, \omega^i_k\}_{i=1}^{N_p}$ at level D_k. For $k = m$ we have $P_{hit}(0,T) \approx \prod_{k=1}^m \gamma_k^{N_p}$.

In [Cerou et al 02] it is proven that the particle estimates are unbiased, i.e.

$$\mathbb{E}[\prod_{k=1}^m \gamma_k^{N_p}] = \mathbb{P}(\tau_m \leq T) = P_{hit}(0,T)$$

and also that:

$$(\mathbb{E}(\prod_{k=1}^m \gamma_k^{N_p} - \prod_{k=1}^m \gamma_k)^p)^{\frac{1}{p}} \leq \frac{a_p b_m}{\sqrt{N_p}},$$

for some finite constant a_p which depends only on the parameter p, and for some finite constant b_m which depends only on the parameter m.

The above leads to an IPS algorithm version to estimate rare event probability for a diffusion process.

Numerical Integration To implement this IPS model in practice one should be able to sample random variables $x_0 \sim P_{x_0}(\cdot)$ and simulate paths of $\{x_t\}$ from $x_{\tau_{k-1}}$ to the k-th level ($k = 1, \ldots, m$). The stochastic processes are commonly simulated through a discrete-time approximation. We use the Euler approximation:

$$X^h_{t_i} = X^h_{t_{i-1}} + a(X^h_{t_{i-1}})(t_i - t_{i-1}) + b(X^h_{t_{i-1}})(W_{t_i} - W_{t_{i-1}}),\ X^h_0 \sim P_{x_0}(\cdot),$$

where independent increments of Brownian motion $\Delta W_{t_i} = (W_{t_i} - W_{t_{i-1}})$ are distributed according to n-dimensional normal distribution with parameters $\mu = (0, \ldots, 0)^T, \Sigma = I \cdot \sqrt{t_i - t_{i-1}}$, where I is a unit matrix. Superscript $h = t_i - t_{i-1}$ - denotes the discrete step size. The important question is whether the discretized problem converges to original continuous time problem. In [Krystul & Bagchi 03] we prove that under certain condition the discretized problem weakly converges to continuous one as the discretization step decreases:

$$|P(\tau^h_D \le T) - P(\tau_D \le T)| \underset{h \to 0}{\longrightarrow} 0,$$

where τ^h_D and τ_D are the respective stopping times of the discrete and continuous time models.

Next section presents a detailed description of the two versions of the algorithm. The difference between them is the resampling method.

3.4 Algorithms 1.1 and 1.2

The following algorithm provides recursive approximation for $P_{hit}(0, T) = \mathbb{P}(\tau_m \le T)$.
Step 0. Level sets

- Choose appropriate nested sequence of closed subsets of $\mathbb{R}^n$:

$$D = D_m \subset D_{m-1} \subset \cdots \subset D_1.$$

Step 1. Initial sampling: $k = 0$

- For $i = 1, \ldots, N_p$ generate initial state value outside D_1:
  $x^i_0 \sim P_{x_0}(\cdot)$ and set $\xi^i_0 = x^i_0$.
- For $i = 1, \ldots, N_p$ set the initial weights: $\omega^i_0 = 1/N_p$ and initial time: $t^i_0 = 0$.

- Then
$$\pi_0^{N_p} = \sum_{i=1}^{N_p} \omega_0^i \delta_{\{\xi_0^i\}}.$$

Iteration k; $k = 1, \ldots, m$ over step 2 (prediction) and step 3 (resampling)
Step 2. Prediction step: $\pi_{k-1} \longrightarrow p_k$

- For $i = 1, \ldots, N_p$ simulate a new path starting at (t_{k-1}^i, ξ_{k-1}^i) until the k-th set D_k is hit or until t^i reaches T.
- This yields new particles $\{\hat{t}_k^i, \hat{\xi}_k^i, \omega_{k-1}^i\}_{i=1}^{N_p}$.
- $p_k^{N_p}$ is the empirical distribution associated with the new cloud of particles:
$$p_k^{N_p} = \sum_{i=1}^{N_p} \omega_{k-1}^i \delta_{\{\hat{\xi}_k^i\}}.$$
- The particles which do not reach the set D_k before time T are killed, i.e. we set $\hat{\omega}_k^i = 0$ if $\hat{t}_k^i = T$ and $\hat{\omega}_k^i = \omega_{k-1}^i$ if $\hat{t}_k^i < T$.
- The new set of particles is $\{\hat{t}_k^i, \hat{\xi}_k^i, \hat{\omega}_k^i\}_{i=1}^{N_p}$.
- Approximation of γ_k:
$$\gamma_k \approx \gamma_k^{N_p} = \sum_{i=1}^{N_p} \hat{\omega}_k^i.$$
If all particles are killed, i.e. $\gamma_k^{N_p} = 0$, then the algorithm is stopped and we set $P_{hit}(0,T) \approx 0$.

Step 3. Resampling step: $p_k \longrightarrow \pi_k$
"Random resampling" (Algorithm 1.1):

- For $i = 1, \ldots, N_p$ set $\tilde{\xi}_k^i = \hat{\xi}_k^i$ and
$$\tilde{\omega}_k^i = \frac{\hat{\omega}_k^i}{\sum_{j=1}^{N_p} \hat{\omega}_k^j}, \ i = 1, \ldots, N_p,$$

- Resample with replacement N_p particles ξ_k^i according to the empirical measure

$$\pi_k^{N_p} = \sum_{i=1}^{N_p} \tilde{\omega}_k^i \delta_{\{\tilde{\xi}_k^i\}}.$$

- The new set of particles is $\{t_k^i, \xi_k^i, \omega_k^i\}_{i=1}^{N_p}$, with $\omega_k^i = 1/N_p$.
- If $k < m$ then repeat step 2 and 3 for $k := k + 1$.
- Otherwise, algorithm stops with $P_{hit}(0,T) \approx \prod_{k=1}^{m} \gamma_k^{N_p}$.

"Uniform resampling" (Algorithm 1.2):

- If the particle has succeeded in reaching the k-th level before time T, we keep it, so $\{t_k^i, \xi_k^i, \omega_k^i\} = \{\hat{t}_k^i, \hat{\xi}_k^i, \hat{\omega}_k^i\}$, otherwise, $\{t_k^i, \xi_k^i, \omega_k^i\}$ is chosen randomly and uniformly in the set of particles having succeeded to reach the set D_k before time T.
- Normalize the weights.
- If $k < m$ then repeat step 2 and 3 for $k := k + 1$.
- Otherwise, algorithm stops with $P_{hit}(0,T) \approx \prod_{k=1}^{m} \gamma_k^{N_p}$.

3.5 Numerical evaluation

We will show in this section how the IPS algorithm (see section 3.4) for simulating rare events works in a simple case.

We consider the Geometric Brownian motion process driven by the following SDE:

$$dx_t = (\mu + \frac{\sigma^2}{2})x_t dt + \sigma x_t dW_t, \ x_0 = x \tag{12}$$

where μ and σ are strictly positive constants. We want to estimate the probability that process $\{x_t\}$ will hit barrier d before time T, i.e. $P(\tau_d(x) \leq T)$ where $\tau_d(x) \stackrel{\triangle}{=} \inf\{t > 0 : x_t \in [d, +\infty); \ x_0 = x\}$. Although the above model is rather simple one, it allows us to check the IPS method accuracy. This is possible thanks to the following analytical formula:

$$P(\tau_d(x) \leq t) = \int_0^t \frac{\ln(d/x)}{\sqrt{2\pi\sigma^3 s^3}} \exp\{\frac{-(\ln(d/x) - \mu s)^2}{2\sigma^2 s}\} ds. \tag{13}$$

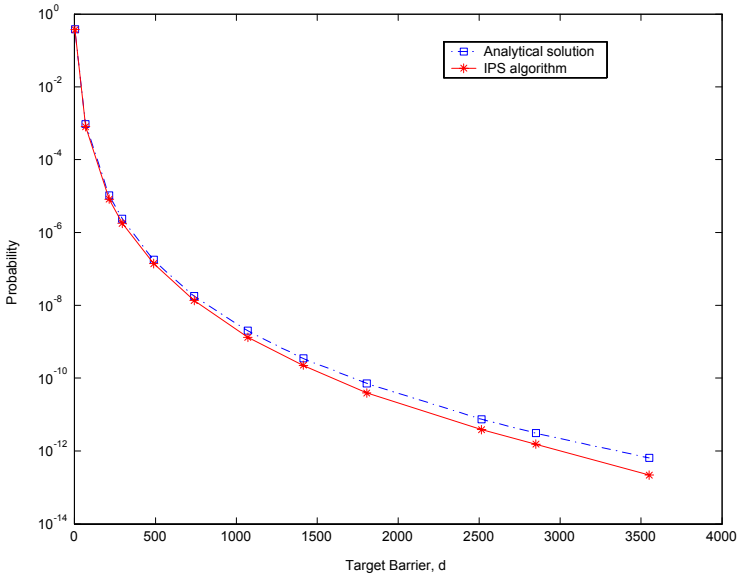

Figure 1: Probability to hit barrier d before time $T = 1$. 1000 runs with 1000 particles

Before starting the simulations we should define the decreasing nested sequence of sets: $D = D_m \subset \cdots \subset D_1$. For $\{D_j,\ j = 1, \ldots, m\}$ we choose an increasing sequence of real numbers $\{d_j,\ j = 1, \ldots, m\}$, with $0 < d_1 < \cdots < d_{m-1} < d_m = d$ and take $D_j = [d_j, +\infty)$. In our example we choose numbers d_j experimentally so that approximately $40\% - 50\%$ of particles started at level D_{j-1} manage to reach level D_j, $j = 1, \ldots, m$.

In figure 1 we see the probability $P(\tau_d(x) \leq T)$ as a function of d. The red curve with asterisks is the numerically computed theoretical value, and the blue curve with squares is the Monte-Carlo simulation result, with 1000 runs of 1000 particles each:

$$P_{hit}(0, T) \approx \frac{1}{1000} \sum_{i=1}^{1000} (\prod_{k=1}^{m} \gamma_k^{N_p})^{(i)}.$$

We use the Euler scheme with 500 discretization time steps. The parameters of Geometric Brownian motion process are $\mu = 1$, $\sigma = 1$ and $x_0 = 1$. The largest value of d was 3550. This means that the probability for the process started at $x_0 = 1$ to reach the desired level is approximately $6.48 \cdot 10^{-13}$, so a standard Monte Carlo approach would not be practical.

From Table 2 we can see that the method works quite well; the relative error is less than 60%.

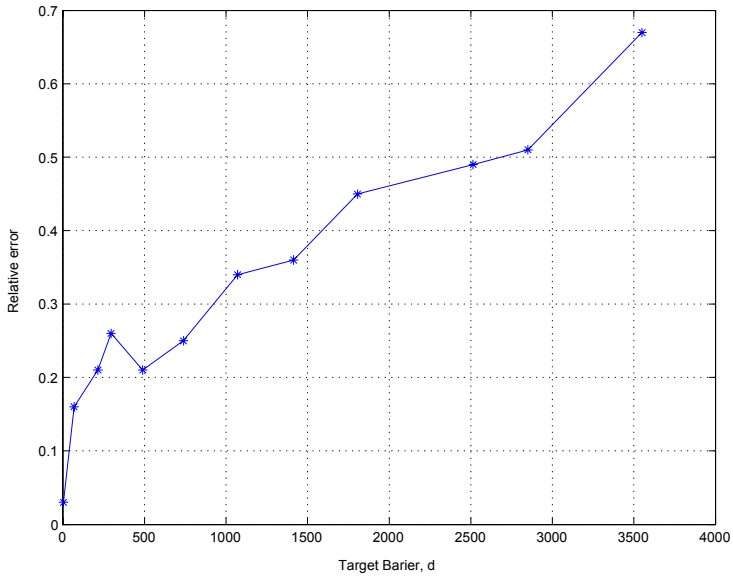

Figure 2: Relative error in figure 1

Table 2. Results of simulations.

Target Level (d)	Number of Levels (dj)	Probability to hit barrier d. Analytical	Probability to hit barrier d. IPS algorithm	Relative Error	Speed-up factor	Computer Time (min.)
5	2	0.384481	0.372872	0.03	2.82E-05	3.97
70	12	0.000956	8.01E-04	0.16	6.82E-03	8.08
215	19	1.05E-05	8.29E-06	0.21	5.78E-01	9.35
295	21	2.38E-06	1.77E-06	0.26	2.63E+00	9.77
490	26	1.78E-07	1.41E-07	0.21	3.15E+01	10.51
740	31	1.80E-08	1.36E-08	0.25	3.18E+02	10.64
1070	36	2.02E-09	1.32E-09	0.34	3.19E+03	10.79
1415	40	3.51E-10	2.25E-10	0.36	1.85E+04	11.32
1805	45	7.19E-11	3.92E-11	0.45	1.05E+05	11.03
2515	51	7.55E-12	3.87E-12	0.49	1.05E+06	11.15
2850	54	3.14E-12	1.55E-12	0.51	2.62E+06	11.15
3550	58	6.48E-13	2.17E-13	0.67	1.86E+07	11.49

We can still get quite satisfactory results running fewer simulations. We repeat the above example with the same parameters, but instead of 1000 runs with 1000 particles now we run only 100 with 1000 particles. The results can be seen in figures 5-6 and in

Table 3. We also calculate the speed up factor to get an idea how fast the IPS method is in comparison with standard Monte Carlo (see Table 3).

Table 3. Results of simulations

Target Level (d)	Number of Levels (dj)	Probability to hit barrier d. Analytical	Probability to hit barrier d. IPS algorithm	Relative Error	Speed-up factor	Computer Time (min.)
5	2	3.84E-01	0.374263	0.03	2.81E-04	0.33
70	12	9.56E-04	7.99E-04	0.16	6.84E-02	0.62
215	19	1.05E-05	8.43E-06	0.20	5.70E+00	0.70
295	21	2.38E-06	1.88E-06	0.21	2.48E+01	0.73
490	26	1.78E-07	1.27E-07	0.28	3.52E+02	0.74
740	31	1.80E-08	1.01E-08	0.44	4.32E+03	0.82
1070	36	2.02E-09	1.21E-09	0.40	3.49E+04	0.84
1415	40	3.51E-10	1.05E-10	0.70	3.98E+05	0.83
1805	45	7.19E-11	7.30E-11	0.02	5.65E+05	0.85
2515	51	7.55E-12	1.16E-11	0.54	3.50E+06	0.82
2850	54	3.14E-12	4.67E-13	0.85	8.67E+07	0.83
3550	58	6.48E-13	2.52E-13	0.61	1.59E+08	0.84

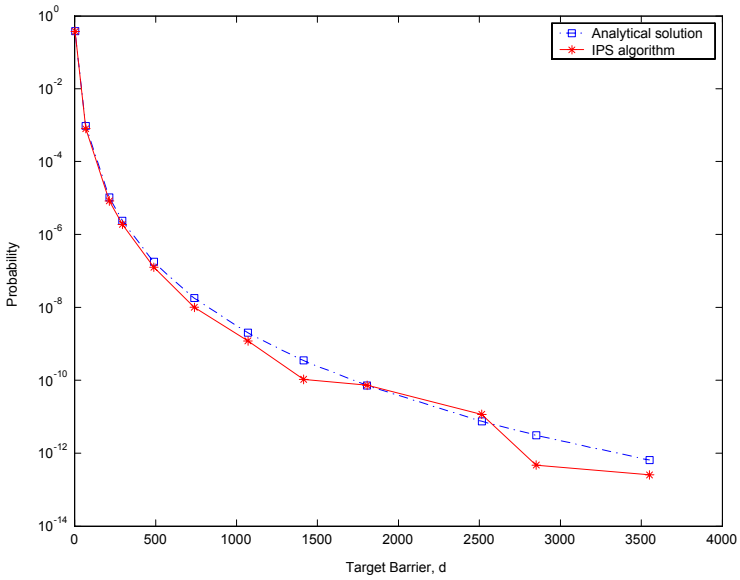

Figure 3: Probability to hit barrier d before time $T = 1$, 100 runs with 1000 particles

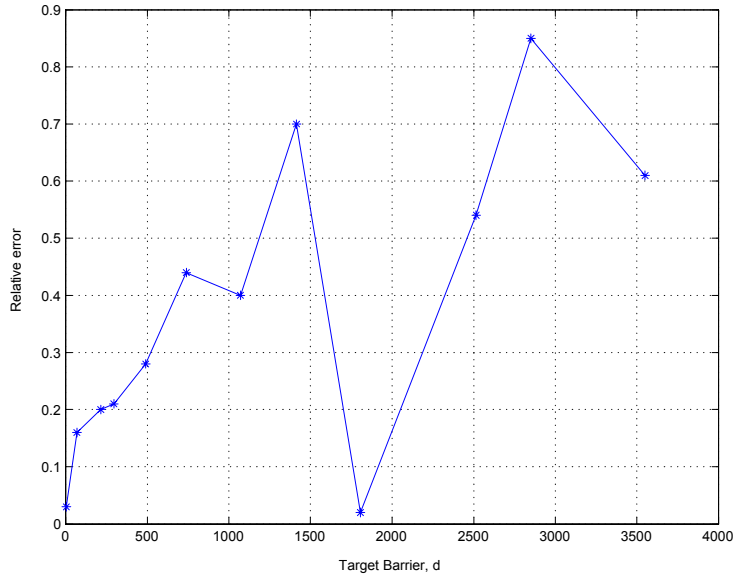

Figure 4: Relative error in figure 3

4 Rare event Monte Carlo simulation for switching diffusion

4.1 Stochastic hybrid model

In this section we recall definitions and results needed for the study of rare event simulations for switching diffusion processes.

Let $\{x_t, \theta_t\}$ be the switching diffusion taking its values in $\mathbb{R}^n \times \mathbb{M}$ defined by

$$dx_t = a(\theta_t, x_t)dt + b(\theta_t, x_t)dW_t, \tag{14}$$

$$P_{\theta_{t+\delta}|\theta_t, x_t}(\theta|\eta, x) = \lambda_{\eta\theta}(x)\delta + o(\delta),\ \eta \neq \theta, \tag{15}$$

where $\mathbb{M}$ is a finite set of modes and $(W_t)_{t\geq 0}$ is a Brownian motion in $\mathbb{R}^n$. We set $\tau_D \stackrel{\triangle}{=} \inf\{t > 0 : x_t \in D\}$ for the first passage time of $\{x_t\}$ to a closed connected Borel set D. We want to estimate the probability $P_{hit}(0, T)$ that $\{x_t\}$ will hit the set D on the time interval $(0, T]$, $T < \infty$:

$$P_{hit}(0, T) = P(\tau_D \leq T). \tag{16}$$

Let $\mathbb{M} = \{e_1, e_2, \ldots, e_N\}$ be a finite set of unit vectors, i.e. $e_i \in \mathbb{M}$ is a i-th unit vector in $\mathbb{R}^N$. Then the $(\mathbb{R}^n \times \mathbb{M})$-valued switching diffusion process (14,15) can be represented as a solution of the stochastic differential equation of the following form [Ghosh et al 93]:

$$dx_t = a(x_t, \theta_t)dt + b(x_t, \theta_t)dW_t \tag{17}$$

$$d\theta_t = \int_{\mathbb{R}} c(x_t, \theta_{t-}, z)p(dt, dz) \tag{18}$$

for $t \geq 0$, with (x_0, θ_0) a prescribed $(\mathbb{R}^n \times \mathbb{M})$-valued random variable; $p(dt, dz)$ is a Poisson random measure with intensity $dt \cdot dz$; (W_t) is an n-dimensional Wiener process independent of (x_0, θ_0) and $p(dt, dz)$. Under assumption on W_t, $p(dt, dz)$, (x_0, θ_0), and on functions a, b and c, equation (17,18) admits an a.s. pathwise unique solution [Ghosh et al 93] and which is a strong Markov process (see [Blom et al 03a]). Sufficient assumptions on a, b and c are as follows [Blom et al 03a], [Ghosh et al 93]:

Let

$$\begin{aligned} a &: \mathbb{R}^n \times \mathbb{M} \to \mathbb{R}^n \\ b &: \mathbb{R}^n \times \mathbb{M} \to \mathbb{R}^{n\times n} \\ \lambda_{ij} &: \mathbb{R}^n \to \mathbb{R},\ i, j = 1, 2, \ldots, N \end{aligned}$$

For each $\theta \in \mathbb{M}$, $a(\cdot, \theta)$ and $b(\cdot, \theta)$ are assumed to be continuous, Lipschitz and to satisfy a linear growth condition. For all $i, j \in \{1, \ldots, N\}$, $\lambda_{ij}(\cdot)$ is assumed to be bounded, continuous and Lipschitz, $\lambda_{ij}(\cdot) \geq 0$ for $i \neq j$ and $\sum_{j=1}^{N} \lambda_{ij}(\cdot) = 0$ for any $i \in \{1, \ldots, N\}$. For $i, j \in \{1, \ldots, N\}$, $i \neq j$, $x \in \mathbb{R}^n$ we construct the intervals $\Delta_{ij}(x)$ of the real line in the following manner [Ghosh et al 93]:

$$\begin{aligned}
\Delta_{12}(x) &= [0, \lambda_{12}(x)) \\
\Delta_{13}(x) &= [\lambda_{12}(x), \lambda_{12}(x) + \lambda_{13}(x)) \\
&\vdots \\
\Delta_{1N}(x) &= \Big[\textstyle\sum_{j=2}^{N-1} \lambda_{1j}(x), \sum_{j=2}^{N} \lambda_{1j}(x)\Big) \\
\Delta_{21}(x) &= \Big[\textstyle\sum_{j=2}^{N} \lambda_{1j}(x), \sum_{j=2}^{N} \lambda_{1j}(x) + \lambda_{21}(x)\Big)
\end{aligned}$$

and so on. Thus, in general,

$$\Delta_{ij}(x) = [\sum_{i'=1}^{i-1} \sum_{\substack{j'=1 \\ j' \neq i'}}^{N} \lambda_{i'j'}(x) + \sum_{\substack{j'=1 \\ j' \neq i}}^{j-1} \lambda_{ij'}(x), \sum_{i'=1}^{i-1} \sum_{\substack{j'=1 \\ j' \neq i'}}^{N} \lambda_{i'j'}(x) + \sum_{\substack{j'=1 \\ j' \neq i}}^{j} \lambda_{ij'}(x)).$$

For fixed x these are disjoint intervals, and the length of $\Delta_{ij}(x)$ is $\lambda_{ij}(x)$. Now we define the function $c : \mathbb{R}^n \times \mathbb{M} \times \mathbb{R} \to \mathbb{M}$ as follows:

$$c(x, e_i, z) = \begin{cases} e_j - e_i & \text{if } z \in \Delta_{ij}(x) \\ 0 & \text{otherwise.} \end{cases} \tag{19}$$

4.2 Monte Carlo simulation

We define the following interval:

$$U(x) \stackrel{\triangle}{=} \bigcup_{i=1}^{N} \Big(\bigcup_{\substack{j=1 \\ j \neq i}}^{N} \Delta_{ij}(x) \Big),$$

it includes all intervals $\Delta_{ij}(x), i, j = 1, \ldots, N$, $i \neq j$. Since the length of each interval $\Delta_{ij}(x)$ is $\lambda_{ij}(x)$, and this is continuous and bounded function for $i, j = 1, \ldots, N$, $i \neq j$, then it follows that the length of interval $U(x)$ (denote $l(U(x))$) is bounded and continuous function of x. Therefore, it has a maximum at some point $x^\star$:

$$l(U(y)) \leq l(U(x^\star)) \text{ for all } y \in \mathbb{R}^n.$$

Let $U_{\max} \triangleq U(x^\star)$ denote the interval of maximum length and let $\lambda_{\max} \triangleq l(U_{\max})$ denote the length of $U_{\max}$. Equation (18) can be rewritten as follows

$$d\theta_t = \int_{U_{\max}} c(X_t, \theta_{t-}, z) p(dt, dz).$$

Let $N(t)$ be a standard Poisson process with intensity $\lambda_{\max}$. Let τ_n^θ, $n = 1, 2, \ldots$ denote the jump times of $N(t)$. Let $U_{\max}$ be the "mark" space, and $(Z_n)_{n\geq 1}$ be a sequence of i.i.d. random variables with uniform distribution on $U_{\max}$, independent of $N(t)$. In this special case we can represent the random Poisson measure $p(dt, dz)$ with intensity $dt \cdot dz$ as a random counting measure associated to the marked point process $(\tau_n^\theta, Z_n)_{n\geq 0}$, i.e. for each $A \subset U_{\max}$

$$p((0,t], A) = \sum_{n\geq 1} \mathbf{1}_{\{\tau_n \leq t\}} \cdot \mathbf{1}_{\{Z_n \in A\}}, \tag{20}$$

and

$$\mathbb{E}[p((0,t], A)] = \lambda_{\max} \cdot t \cdot \mathbb{P}(Z_n \in A) = \lambda_{\max} \cdot t \cdot \frac{l(A)}{\lambda_{\max}} = t \cdot l(A).$$

The representation (20) is very convenient for practical problems. We see that $p(dt, dz)$ can be easily generated just by generating a sequence of independent random variables τ_k^θ and Z_k, $k = 1, 2, \ldots$.

Stochastic processes are commonly simulated through a discrete-time approximations (e.g. Euler scheme). The solution of SDE (17)-(18) is simulated through the discrete-time approximation $(X_{t_i}^h, \theta_{t_i}^h)$ that is iteratively computed from the initial condition (x_0, θ_0) using[1]:

$$X_{t_i}^h = X_{t_{i-1}}^h + a(X_{t_{i-1}}^h, \theta_{t_{i-1}}^h)(t_i - t_{i-1}) + b(X_{t_{i-1}}^h, \theta_{t_{i-1}}^h)(W_{t_i} - W_{t_{i-1}}), \tag{21}$$

$$\theta_{t_i}^h = \theta_{t_{i-1}}^h + \int_{U_{\max}} c(X_{t_i}^h, \theta_{t_{i-1}}^h, z) p(\{t_i\}, dz). \tag{22}$$

At the grid point[2], (22) computes the jump of θ^h exactly, conditional on

$$(X_{t_i}^h, \theta_{t_i^-}^h) = (X_{t_i}^h, \theta_{t_{i-1}}^h),$$

[1]This method is similar to what has been used before at NLR within Monte Carlo simulation of accident risk models

[2]here the grid is the superposition of the random jump times τ_n of the component θ_t on interval $[0, T]$ and a deterministic grid

if t_i is indeed a point of the Poisson random measure. Otherwise, the jump term is zero. (The integral in equation (22) entails at most a single evaluation of the function c because $p(\{t_i\}, dz)$ is a point mass at the mark z that arrives at t_i if t_i is a jump time.)

The important question is whether the discretized problem converges to original continuous time problem. In [Krystul & Bagchi 03] it is shown that under certain condition the discretized problem weakly converges to continuous one as discretization step decreases:

$$|P(\tau_D^h \leq T) - P(\tau_D \leq T)| \underset{h\to 0}{\longrightarrow} 0,$$

where τ_D^h and τ_D are the stopping times of discrete and continuous time model correspondingly.

4.3 Product of Conditional Probabilities

For a diffusion, in subsection 3.2 a sequence of gateway states has been used to characterize the rare event probability as a product of conditional probabilities. Here we extend this product form characterization to a switching diffusion. We assume that switching diffusion (17),(18) starts at $t = 0$ in a Borel set $\bar{D}_0 \subset \mathbb{R}^n \times \mathbb{M}$ with a known initial probability distribution $P_{x_0,\theta_0}(\cdot)$. As in [Cerou et al 02] we assume a sequence of nested Borel sets, $\bar{D} = \bar{D}_m \subset \cdots \subset \bar{D}_1$ which are defined as follows:

$$\bar{D}_k \stackrel{\triangle}{=} D_k \times \mathbb{M},\ k = 1, \ldots, m \tag{23}$$

where D_k is a closed Borel set of $\mathbb{R}^n$, and $\bar{D}_1$ such that $\bar{D}_1 \cap \bar{D}_0 = \varnothing$. Hence $\bar{D}_k$ are cylinder sets. The first moment that $\{x_t, \theta_t\}$ hits a set $\bar{D}_k$ is defined as the stopping time:

$$\tau_k \stackrel{\triangle}{=} \inf\{t \geq 0 : (x_t, \theta_t) \in \bar{D}_k\} = \inf\{t \geq 0 : x_t \in D_k\},$$

$\tau_k = \infty$ if this set is empty. We want to estimate $\mathbb{P}(\tau_m \leq T)$, for some $T < \infty$, i.e. the probability that switching diffusion $\{x_t, \theta_t\}$ will hit the rare event set $\bar{D}$ before time T. The process $\{x_t, \theta_t\}$, before hitting $\bar{D}$, passes through a sequence of nested cylinders (23). To capture the behavior of $\{x_t, \theta_t\}$ between the first cylinder hittings, we introduce the stochastic sequence $\{\bar{\varkappa}_k\}$ representing the paths of $\{x_t, \theta_t\}$ between the successive cylinder hittings.

$$\bar{\varkappa}_k = \{(x_t, \theta_t) : \tau_{k-1} \wedge T \leq t \leq \tau_k \wedge T\},\ \bar{\varkappa}_0 = (x_0, \theta_0).$$

By the strong Markov property of $\{x_t, \theta_t\}$ the stochastic sequence $\{\bar{\varkappa}_k\}$ forms a Markov process [Cerou et al 02] with value in the Borel path-space $(E, \mathcal{E})$ of $\{x_t, \theta_t\}$. Following

[Cerou et al 02] we also introduce the $\{0,1\}$-valued variables $\{y_k, k = 1,\ldots,m\}$ defined as follows:

$$y_k(\omega) \stackrel{\triangle}{=} \mathbf{1}_{\{\omega: x_{\tau_k \wedge T}(\omega) \in D_k\}} = \mathbf{1}_{\{\omega:\tau_k(\omega)\leq T\}}. \tag{24}$$

Hence, for each k we have

$$y_k(\omega) = \mathbf{1}_{\{\omega:\tau_k(\omega)\leq T\}} = \prod_{i=1}^{k} \mathbf{1}_{\{\omega:\ \tau_i(\omega)\leq T\}} = \prod_{i=1}^{k} y_i(\omega). \tag{25}$$

Next we characterize $P_{hit}(0,T)$ in terms of the sequence $\{y_k\}$. By its definition,

$$P_{hit}(0,T) = \mathbb{P}(\tau_m \leq T) = \mathbb{E}[\mathbf{1}_{\{\tau_m \leq T\}}]$$

Subsequent substitution of (24) and (25) yields:

$$P_{hit}(0,T) = \mathbb{E}[y_m] = \mathbb{E}[\prod_{k=1}^{m} y_k]. \tag{26}$$

Since y_k assumes values from $\{0,1\}$,

$$\mathbb{E}[\prod_{k=1}^{m} y_k] = \prod_{k=1}^{m} \mathbb{E}[y_k | y_{k-1} = 1, \ldots, y_1 = 1]$$

Substituting this into (26) yields

$$\begin{aligned} P_{hit}(0,T) &= \prod_{k=1}^{m} \mathbb{E}[y_k | y_{k-1} = 1, \ldots, y_1 = 1] \\ &= \prod_{k=1}^{m} \mathbb{P}(\tau_k \leq T | \tau_{k-1} \leq T, \ldots, \tau_1 \leq T) \\ &= \prod_{k=1}^{m} \mathbb{P}(\tau_k \leq T | \tau_{k-1} \leq T) \end{aligned} \tag{27}$$

This means that (27) characterizes the probability $P_{hit}(0,T)$ of the rare event as a product of conditional probabilities of intermediate "less rare" events leading to it. Thus, if we estimate the conditional probabilities

$$\gamma_k \stackrel{\triangle}{=} \mathbb{P}(\tau_k \leq T | \tau_{k-1} \leq T) \ \text{ for } \ k = 1, \ldots, m$$

and insert this in (27) then we get for $P_{hit}(0,T)$:

$$P_{hit}(0,T) = \prod_{k=1}^{m} \gamma_k \tag{28}$$

The estimation of the probabilities γ_k is in the sequel studied as a nonlinear filtering problem.

4.4 Particle Filter Approach of [Cerou et al 02]

The process $\{x_t, \theta_t\}$ is a strong Markov process. Because of this, the filtering problem considered is to evaluate the conditional distribution of $\xi_k \stackrel{\triangle}{=} (x_{\tau_k \wedge T}, \theta_{\tau_k \wedge T})$ given $y_{1:k}$, which we denote as $P_{\xi_k|y_{1:k}}(B)$ for $B \in \mathcal{E}'$, i.e. the Borel σ-algebra of $E' = \mathbb{R}^n \times \mathbb{M}$. Because $\{\bar{\varkappa}_k\}$ is a strong Markov process, $\{\xi_k\}$ is a $\mathbb{R}^n \times \mathbb{M}$-valued Markov sequence, and filtering for $\{\xi_k\}$ is accomplished through the following sequence of transformations

$$P_{\xi_{k-1}|y_{1:k-1}}(\cdot) \xrightarrow{\text{prediction}} P_{\xi_k|y_{1:k-1}}(\cdot) \xrightarrow{\text{measurement}} P_{\xi_k|y_{1:k}}(\cdot).$$

with the prediction satisfying:

$$P_{\xi_k|y_{1:k-1}}(B) = \int_{E'} P_{\xi_k|\xi_{k-1}}(B|\xi) P_{\xi_{k-1}|y_{1:k-1}}(d\xi), \ B \in \mathcal{E}',$$

and the measurement update:

$$\frac{dP_{\xi_k|y_{1:k}}}{dP_{\xi_k|y_{1:k-1}}}(\xi) = \frac{\mathbf{1}_{\{\xi \in \bar{D}_k\}}}{\int_{E'} \mathbf{1}_{\{\xi' \in \bar{D}_k\}} P_{\xi_k|y_{1:k-1}}(d\xi')}$$

where $\frac{dP_{\xi_k|y_{1:k}}}{dP_{\xi_k|y_{1:k-1}}}$ is the Radon-Nikodym derivative and $P_{\xi_k|\xi_{k-1}}$ is the transition measure of the Markov sequence $\{\xi_k\}$. Next we introduce the notation

$$\begin{aligned} p_k(B) &\stackrel{\triangle}{=} P_{\xi_k|y_{1:k-1}}(B|1\ldots1), \\ \pi_k(B) &\stackrel{\triangle}{=} P_{\xi_k|y_{1:k}}(B|1\ldots1). \end{aligned}$$

This yields

$$\mathbb{E}[y_k|y_{1:k-1} = (1,1,\ldots,1)] = \int_{E'} \mathbf{1}_{\{\xi \in \bar{D}_k\}} p_k(d\xi).$$

Similarly as in (27), this can be written as:

$$\gamma_k = P(\tau_k \leq T | \tau_{k-1} \leq T) = \int_{E'} \mathbf{1}_{\{\xi \in \bar{D}_k\}} p_k(d\xi). \tag{29}$$

With this each of the m terms γ_k in (28) is characterized as a solution of a sequence of filtering problems.

We denote by $\gamma_k^{N_p}$, $p_k^{N_p}$ and $\pi_k^{N_p}$ the numerical approximations of γ_k, p_k and π_k respectively. At $t = 0$ we start with an empirical measure of the form

$$\pi_0^{N_p} \triangleq \sum_{i=1}^{N_p} \omega_0^i \delta_{\{\xi_0^i\}},$$

where $\omega_0^i = 1/N_p$, and ξ_0^i are independent samples from initial distribution $P_{\xi_0}(\cdot)$. Hence π_0^{1,N_p} is defined by a system of N_p particles $\{t_0^i, \xi_0^i, \omega_0^i\}_{i=1}^{N_p}$, where each i-th particles is a triplet consisting of current time, state and weight. From $t = 0$ to τ_1, and from τ_{k-1} to τ_k each particle evolves stochastically according to equations (17, 18) with initial condition (t_{k-1}^i, ξ_{k-1}^i) (prediction step) until it reaches the next cylinder set $\bar{D}_k$ or the final time T. Let $\{\hat{t}_k^i, \hat{\xi}_k^i, \omega_{k-1}^i\}_{i=1}^{N_p}$ denote the values of the particles after the k-th prediction. Then the empirical distribution $p_k^{N_p}$ associated with the predicted cloud of particles is:

$$p_k^{N_p} = \sum_{i=1}^{N_p} \omega_{k-1}^i \delta_{\{\hat{\xi}_k^i\}}.$$

The particles which do not reach the set $\bar{D}_k$ before time T are deleted, i.e. we set $\hat{\omega}_k^i = 0$ if $\hat{t}_k^i = T$, else we set $\hat{\omega}_k^i = \omega_{k-1}^i$. If all particles become deleted, the algorithm is stopped and $\gamma_m^{N_p} = 0$. Otherwise

$$\gamma_k \approx \gamma_k^{N_p} = \sum_{i=1}^{N_p} \hat{\omega}_k^i.$$

The empirical distribution $\pi_k^{N_p}$ associated with the measurement updated cloud of particles is:

$$\pi_k^{N_p} = \sum_{i=1}^{N_p} \tilde{\omega}_k^i \delta_{\{\tilde{\xi}_k^i\}}. \tag{30}$$

with

$$\tilde{\omega}_k^i = \frac{\hat{\omega}_k^i}{\sum_{j=1}^{N_p} \hat{\omega}_k^j}, \ \tilde{\xi}_k^i = \hat{\xi}_k^i, \quad i = 1, \ldots, N_p,$$

Particles having reached the set $\bar{D}_k$ are used for a resampling with replacement of empirical measure (30) step. After this step we again have N_p particles $\{t_k^i, \xi_k^i, \omega_k^i\}_{i=1}^{N_p}$ at level $\bar{D}_k$. For $k = m$ we have $P_{hit}(0, T) \approx \prod_{k=1}^{m} \gamma_k^{N_p}$.

In [Cerou et al 02] it is proven that the particle estimates are unbiased, i.e.

$$\mathbb{E}[\prod_{k=1}^{m} \gamma_k^{N_p}] = \mathbb{P}(\tau_m \leq T) = P_{hit}(0,T)$$

and also that:

$$(\mathbb{E}(\prod_{k=1}^{m} \gamma_k^{N_p} - \prod_{k=1}^{m} \gamma_k)^p)^{\frac{1}{p}} \leq \frac{a_p b_m}{\sqrt{N_p}},$$

for some finite constant a_p which depends only on the parameter p, and for some finite constant b_m which depends only on the parameter m.

The above leads to an IPS algorithm version to estimate rare event probability for a switching diffusion process.

4.5 Algorithms 1.1 and 1.2 for switching diffusions

The following algorithm provides recursive approximation for $P_{hit}(0,T) = \mathbb{P}(\tau_m \leq T)$.
Step 0. Level sets

- Choose appropriate nested sequence of closed subsets of $\mathbb{R}^n$: $D = D_m \subset D_{m-1} \subset \cdots \subset D_1$, and define $\bar{D}_k = D_k \times \mathbb{M}$, $k = 1, \ldots, m$.

Step 1. Initial sampling: $k = 0$

- For $i = 1, \ldots, N_p$ generate initial state value outside $\bar{D}_1$:
  $(x_0^i, \theta_0^i) \sim P_{x_0,\theta_0}(\cdot)$ and set $\xi_0^i = (x_0^i, \theta_0^i)$.
- For $i = 1, \ldots, N_p$ set the initial weights: $\omega_0^i = 1/N_p$ and initial time: $t_0^i = 0$.
- Then

$$\pi_0^{N_p} = \sum_{i=1}^{N_p} \omega_0^i \delta_{\{\xi_0^i\}}.$$

Iteration k; $k = 1, \ldots, m$ over step 2 (prediction) and step 3 (resampling)
Step 2. Prediction step: $\pi_{k-1} \longrightarrow p_k$

- For $i = 1, \ldots, N_p$ simulate a new path (see [Krystul & Bagchi 03]) starting at (t_{k-1}^i, ξ_{k-1}^i) until the k-th cylinder set $\bar{D}_k$ is hit or until t^i reaches T.

- This yields new particles $\{\hat{t}_k^i, \hat{\xi}_k^i, \omega_{k-1}^i\}_{i=1}^{N_p}$.

- $p_k^{N_p}$ is the empirical distribution associated with the new cloud of particles:

$$p_k^{N_p} = \sum_{i=1}^{N_p} \omega_{k-1}^i \delta_{\{\hat{\xi}_k^i\}}.$$

- The particles which do not reach the set $\bar{D}_k$ before time T are killed, i.e. we set $\hat{\omega}_k^i = 0$ if $\hat{t}_k^i = T$ and $\hat{\omega}_k^i = \omega_{k-1}^i$ if $\hat{t}_k^i < T$.

- The new set of particles is $\{\hat{t}_k^i, \hat{\xi}_k^i, \hat{\omega}_k^i\}_{i=1}^{N_p}$.

- Approximation of γ_k:

$$\gamma_k \approx \gamma_k^{N_p} = \sum_{i=1}^{N_p} \hat{\omega}_k^i.$$

  If all particles are killed, i.e. $\gamma_k^{N_p} = 0$, then the algorithm is stopped and we set $P_{hit}(0,T) \approx 0$.

Step 3. Resampling step: $p_k \longrightarrow \pi_k$
"Random resampling" (Algorithm 1.1):

- For $i = 1, \ldots, N_p$ set $\tilde{\xi}_k^i = \hat{\xi}_k^i$ and

$$\tilde{\omega}_k^i = \frac{\hat{\omega}_k^i}{\sum_{j=1}^{N_p} \hat{\omega}_k^j}, \ i = 1, \ldots, N_p.$$

- Resample with replacement N_p particles ξ_k^i according to the empirical measure

$$\pi_k^{N_p} = \sum_{i=1}^{N_p} \tilde{\omega}_k^i \delta_{\{\tilde{\xi}_k^i\}}.$$

- The new set of particles is $\{t_k^i, \xi_k^i, \omega_k^i\}_{i=1}^{N_p}$, with $\omega_k^i = 1/N_p$.

- If $k < m$ then repeat step 2 and 3 for $k := k + 1$.

- Otherwise, algorithm stops with $P_{hit}(0,T) \approx \prod_{k=1}^{m} \gamma_k^{N_p}$.

"Uniform resampling" (Algorithm 1.2):

- If the particle has succeeded in reaching the k-th set, we keep it, so $\{t_k^i, \xi_k^i, \omega_k^i\} = \{\hat{t}_k^i, \hat{\xi}_k^i, \hat{\omega}_k^i\}$, otherwise, $\{t_k^i, \xi_k^i, \omega_k^i\}$ is chosen randomly and uniformly in the set of particles having succeeded to reach the set $\bar{D}_k$.
- Normalize the weights.
- If $k < m$ then repeat step 2 and 3 for $k := k + 1$.
- Otherwise, algorithm stops with $P_{hit}(0,T) \approx \prod_{k=1}^{m} \gamma_k^{N_p}$.

4.6 Algorithms 1.1^θ and 1.2^θ (Sampling per Mode)

The initial sampling step 1 and the resampling step 3 of the IPS algorithm do not work well for hybrid state processes the conditional mode probabilities of which may be small. Therefore, we propose "sampling per mode" for the initial sampling step 1 and for the resampling step 3. This idea of sampling per mode was successfully introduced in [Blom & Bloem 03a] for target tracking problems.

If the initial probabilities of some particular modes are very small then it is highly unlikely to draw particles in these modes. To avoid this, at the initial sampling step we start with a fixed number of particles in each mode whatever small the initial probability is, i.e.:

Step 1H. Initial sampling: $k = 0$

- Choose for each mode $e_i \in \mathbb{M} = \{e_1, \ldots, e_N\}$, $i = 1, \ldots, N$ an integer N_p^i, so that $N_p = \sum_{i=1}^{N} N_p^i$.
- For each $i = 1, \ldots, N$ sample N_p^i initial state values outside $\bar{D}_1$: $x_0^j \sim P_{x_0|\theta_0}(\cdot|e_i)$, $\theta_0^j = e_i$, set $\xi_0^j = (x_0^j, \theta_0^j)$ and assign initial weight: $\omega_0^j = \frac{P_{\theta_0}(e_i)}{N_p^i}$ and initial time $t_0^i = 0$, for $j \in \{\sum_{k=0}^{i-1} N_p^k + 1, \ldots, \sum_{k=0}^{i-1} N_p^k + N_p^i\}$, with convention $N_p^0 = 0$.
- Then

$$\pi_0^{N_p} = \sum_{i=1}^{N_p} \omega_0^i \delta_{\{\xi_0^i\}},$$

and

$$\gamma_0^{N_p} = 1.$$

Similarly, for the resampling step we also keep a fixed number of particles per mode, i.e.:

Step 3H. Resampling step: $p_k \longrightarrow \pi_k$

"Random resampling per mode" (Algorithm 1.1$^\theta$):

- For each mode $\theta \in \mathbb{M}$ evaluate the unnormalized weights:

$$\tilde{\omega}_k^{\theta,i} = \hat{\omega}_k^i \cdot \mathbf{1}_{\{\theta^i_{\tau_k}=\theta\}},\ i = 1, \ldots, N_p.$$

  For $i = 1, \ldots, N_p$ set $\tilde{\xi}_k^i = \hat{\xi}_k^i$, then we have the empirical unnormalized distribution per mode:

$$\pi_k^{\theta,N_p} \triangleq \sum_{i=1}^{N_p} \tilde{\omega}_k^{\theta,i} \delta_{\{\tilde{\xi}_k^i\}}, \tag{31}$$

  and the total weight per mode equals $\sum_{j=1}^{N_p} \tilde{\omega}_k^{\theta,j}$.

- For each $\theta \in \mathbb{M}$, resample with replacement N_p^θ values ξ_k^i according to the empirical measure (31), and assign new weights to particles in mode θ:

$$\omega_k^\theta = \frac{\sum_{j=1}^{N_p} \tilde{\omega}_k^{\theta,j}}{N_p^\theta}.$$

- If $k < m$ then repeat step 2H and 3H for $k := k + 1$.
- Otherwise algorithm stops, with $P_{hit}(0,T) \approx \prod_{k=1}^m \gamma_k^{N_p}$.

"Uniform resampling" (Algorithm 1.2$^\theta$):

- For each mode $\theta \in \mathbb{M}$ the particles that have succeeded to reach the set $\bar{D}_k$ before time T are kept, i.e. $\{t_k^i, \xi_k^i, \omega_k^i\} = \{\hat{t}_k^i, \hat{\xi}_k^i, \hat{\omega}_k^i\}$, those that do not reach $\bar{D}_k$ are selected randomly and uniformly in the set of particles having succeeded to reach the set $\bar{D}_k$ in mode θ.
- Normalize the weights.
- If $k < m$ then repeat step 2 and 3 for $k := k + 1$.
- Otherwise, algorithm stops with $P_{hit}(0,T) \approx \prod_{k=1}^m \gamma_k^{N_p}$.

4.7 Rare switching

The possibility of small mode probabilities is covered well by resampling per mode. However there remains another problem to be solved; the problem of rare transitions (switches) of switching diffusion (17-18).

In the above algorithms during prediction step 2 we generate random path $(\xi^i_{k-1:k})_{i=1}^{N_p} = (x^i_{\tau_{k-1}:\tau_k}, \theta^i_{\tau_{k-1}:\tau_k})_{i=1}^{N_p}$ to approximate the distribution p_k $(k = 1, \dots, m)$. If the probability of some transitions (switches) is very small then, most probably, there will be few switches observed during the generation of these random paths. This may affect the accuracy of estimation of p_k and thus the accuracy of estimation of $P_{hit}(0, T)$ as well. To improve the quality of our estimates we will change the way the random trajectories are being generated. The idea is to use a kind of sequential importance sampling technique. We briefly explain the idea.

Let us consider a random trajectory $u_{t_0:t_N} = (u_{t_0}, \dots, u_{t_N})$ with the density

$$p_{u_{t_0},\dots,u_{t_N}}(u_0, \dots, u_N) = p_{u_{t_0}}(u_0) p_{u_{t_1}|u_{t_0}}(u_1|u_0) \dots p_{u_{t_N}|u_{t_0},\dots,u_{t_{N-1}}}(u_N|u_0, \dots, u_{N-1}).$$

The u_{t_k} is a state of some studied system at time t_k. The standard sequential sampling Monte Carlo works as follows: we pick $u_{t_0} = u_0$ according to the density $p_{u_{t_0}}$ and sequentially generate quantities $u_{t_k} = u_k$ according to the conditional density $p_{u_{t_k}|u_{t_0},\dots,u_{t_{k-1}}}$. In order to provoke more rare events during simulation we replace the densities $p_{u_{t_0}}$, $p_{u_{t_k}|u_{t_0},\dots,u_{t_{k-1}}}$ $(k = 2, \dots N)$ by some known densities $q_{u_{t_k}|u_{t_0},\dots,u_{t_{k-1}}}$, $q_{u_{t_k}|u_{t_0},\dots,u_{t_{k-1}}}$ $(i = 2, \dots N)$ with respect to which the rare events are more likely to occur. In such a way the new density of random trajectory $u_{t_0:t_N}$ is given by

$$q_{u_{t_0},\dots,u_{t_N}}(u_0, \dots, u_N) = q_{u_{t_0}}(u_0) q_{u_{t_1}|u_{t_0}}(u_1|u_0) \dots q_{u_{t_N}|u_{t_0},\dots,u_{t_{N-1}}}(u_N|u_0, \dots, u_{N-1}).$$

To compensate for this change (i.e. to keep the estimates unbiased) the importance weight for each random i-th realization of random path should be evaluated:

$$\omega^i_N = \frac{p_{u_{t_0},\dots,u_{t_N}}(u^i_0, \dots, u^i_N)}{q_{u_{t_0},\dots,u_{t_N}}(u^i_0, \dots, u^i_N)}.$$

This can be done recursively in time:

$$\omega^i_k = \omega^i_{k-1} \frac{p_{u_{t_k}|u_{t_0},\dots,u_{t_{k-1}}}(u^i_k|u^i_0, \dots, u^i_{k-1})}{q_{u_{t_k}|u_{t_0},\dots,u_{t_{k-1}}}(u^i_k|u^i_0, \dots, u^i_{k-1})}, \quad k = 1, \dots, N.$$

Now, we are going to apply the above sequential importance sampling technique to switching diffusion. But before this, we would like to remind that in order to conduct simulations the continuous time stochastic processes are commonly approximated by certain

approximating continuous time processes which are easy to simulate (for example Euler or Milstein approximations) and then the values of the approximating process are determined recursively from the corresponding discretization schemes at discretization time points. In our case we use the Euler scheme (21-22). Therefore, all further derivations are done for the stochastic process $\{X_t^h, \theta_t^h\}$. It is important to point out that on time interval $[0,T]$ the process $\{\theta_t^h\}$ is a continuous time Markov chain with piecewise constant switching rates, i.e. on each individual interval $[t_{k-1}, t_k)$ (where t_k, $k = 0, 1, \ldots$ are the discretization points) it is the homogeneous continuous time Markov chain with constant switching rate matrix $Q = (\lambda_{ij}(X^h_{t_{k-1}}))_{i,j=1}^N$.

Coefficient $c(\cdot)$ in equation (22) is responsible for switchings. To make the rare switches less rare we replace $c(\cdot)$ with $\hat{c}(\cdot)$, and denote the changed process by $\{\hat{X}_t^h, \hat{\theta}_t^h\}$ and its values at discretization points are evaluated from the following discretization scheme:

$$\hat{X}^h_{t_i} = \hat{X}^h_{t_{i-1}} + a(\hat{X}^h_{t_{i-1}}, \hat{\theta}^h_{t_{i-1}})(t_i - t_{i-1}) + b(\hat{X}^h_{t_{i-1}}, \hat{\theta}^h_{t_{i-1}})(W_{t_i} - W_{t_{i-1}}), \tag{32}$$

$$\hat{\theta}^h_{t_i} = \hat{\theta}^h_{t_{i-1}} + \int_{U_{\max}} \hat{c}(\hat{X}^h_{t_i}, \hat{\theta}^h_{t_{i-1}}, z) p(\{t_i\}, dz),\ t_i \in I,\ i = 1, 2, \ldots \tag{33}$$

Then, for $n > k$, $n, k = 1, 2, \ldots$:

$$\begin{aligned}
&P_{X^h_{t_n}, \theta^h_{t_n} | X^h_{t_k}, \theta^h_{t_k}}(A, B | x_k, \theta_k) \\
&\quad = \sum_{\theta_n \in B} \int_A \cdots \sum_{\theta_{k+1} \in \mathbb{M}} \int_{\mathbb{R}^n} \prod_{i=k+1}^n P_{X^h_{t_i}, \theta^h_{t_i} | X^h_{t_{i-1}}, \theta^h_{t_{i-1}}}(dx_i, \theta_i | x_{i-1}, \theta_{i-1}) \\
&\quad = \sum_{\theta_n \in B} \int_A \cdots \sum_{\theta_{k+1} \in \mathbb{M}} \int_{\mathbb{R}^n} \prod_{i=k+1}^n \frac{P_{X^h_{t_i}, \theta^h_{t_i} | X^h_{t_{i-1}}, \theta^h_{t_{i-1}}}(dx_i, \theta_i | x_{i-1}, \theta_{i-1})}{P_{\hat{X}^h_{t_i}, \hat{\theta}^h_{t_i} | \hat{X}^h_{t_{i-1}}, \hat{\theta}^h_{t_{i-1}}}(dx_i, \theta_i | x_{i-1}, \theta_{i-1})} \times \\
&\qquad \times P_{\hat{X}^h_{t_i}, \hat{\theta}^h_{t_i} | \hat{X}^h_{t_{i-1}}, \hat{\theta}^h_{t_{i-1}}}(dx_i, \theta_i | x_{i-1}, \theta_{i-1}) \\
&\quad = \sum_{\theta_n \in B} \int_A \cdots \sum_{\theta_{k+1} \in \mathbb{M}} \int_{\mathbb{R}^n} \prod_{i=k+1}^n \frac{P_{X^h_{t_i} | \theta^h_{t_{i-1}}, X^h_{t_{i-1}}, \theta^h_{t_{i-1}}}(dx_i | \theta_i, x_{i-1}, \theta_{i-1})}{P_{\hat{X}^h_{t_i} | \hat{\theta}^h_{t_{i-1}}, \hat{X}^h_{t_{i-1}}, \hat{\theta}^h_{t_{i-1}}}(dx_i | \theta_i, x_{i-1}, \theta_{i-1})} \times \\
&\qquad \times \frac{P_{\theta^h_{t_{i-1}} | X^h_{t_{i-1}}, \theta^h_{t_{i-1}}}(\theta_i | x_{i-1}, \theta_{i-1})}{P_{\hat{\theta}^h_{t_{i-1}} | \hat{X}^h_{t_{i-1}}, \hat{\theta}^h_{t_{i-1}}}(\theta_i | x_{i-1}, \theta_{i-1})} P_{\hat{X}^h_{t_i}, \hat{\theta}^h_{t_i} | \hat{X}^h_{t_{i-1}}, \hat{\theta}^h_{t_{i-1}}}(dx_i, \theta_i | x_{i-1}, \theta_{i-1}) \\
&= \sum_{\theta_n \in B} \int_A \cdots \sum_{\theta_{k+1} \in \mathbb{M}} \int_{\mathbb{R}^n} \prod_{i=k+1}^n L_{t_i | t_{i-1}}(\theta_i | x_{i-1}, \theta_{i-1}) P_{\hat{X}^h_{t_i}, \hat{\theta}^h_{t_i} | \hat{X}^h_{t_{i-1}}, \hat{\theta}^h_{t_{i-1}}}(dx_i, \theta_i | x_{i-1}, \theta_{i-1}).
\end{aligned}$$

To get the last equality we used the fact that

$$P_{X^h_{t_i}|\theta^h_{t_{i-1}},X^h_{t_{i-1}},\theta^h_{t_{i-1}}}(dx_i|\theta_i,x_{i-1},\theta_{i-1}) = P_{\hat{X}^h_{t_i}|\hat{\theta}^h_{t_{i-1}},\hat{X}^h_{t_{i-1}},\hat{\theta}^h_{t_{i-1}}}(dx_i|\theta_i,x_{i-1},\theta_{i-1})$$

for $i = 1, 2, \ldots$ and $L_{t|s}(\theta|x',\theta') \stackrel{\triangle}{=} \frac{P_{\theta^h_t|X^h_s,\theta^h_s}(\theta|x',\theta')}{P_{\hat{\theta}^h_t|\hat{X}^h_s,\hat{\theta}^h_s}(\theta|x',\theta')}$ denotes the likelihood ratio. This means, an unbiased estimates of distribution p_k $(k = 1, \ldots, m)$ in the algorithm described in the previous section can be obtained by generating random trajectories of the process (32,33) (i.e. sampling according to $P_{\hat{x}_t,\hat{\theta}_t|\hat{x}_s,\hat{\theta}_s}(\cdot|x',\theta')$) and adjusting the weight of each particle recursively in time:

$$\omega^i_{t_j} = \omega^i_{t_{j-1}} \cdot L_{t_j|t_{j-1}}(\theta^i_j|x^i_{j-1},\theta^i_{j-1}).$$

The key is picking a good importance switching distribution $P_{\hat{X}^h_t,\hat{\theta}^h_t|\hat{X}^h_s,\hat{\theta}^h_s}(\cdot|x',\theta')$, i.e. the coefficient $\hat{c}(\cdot)$. Roughly speaking, we should pick it so as to make the rare switch more likely to occur.

4.8 Algorithms 2.1$^\theta$ and 2.2$^\theta$ (Importance Switching for a Markov chain $\{\theta_t\}$)

We use the sequential importance switching method described above to improve step 2 by forcing rare switchings in the discrete component for the case that $\{\theta_t\}$ is a homogeneous continuous time Markov chain, i.e. $L_{t|s}(\theta|x,\theta')$ is x invariant.

Step 2H. Prediction step: $\pi_{k-1} \longrightarrow p_k$;

- For $i = 1, \ldots, N_p$ use (32, 33) to simulate a new path (see [Krystul & Bagchi 03]) starting at $(x^i_{t_{k-1}}, \theta^i_{t_{k-1}})$ to the k-th cylinder set $\bar{D}_k$ is reached or until $t^i = T$;
- This yields a new set of particles $\{\hat{t}^i_k, \hat{\xi}^i_k, \hat{\omega}^i_k\}_{i=1}^{N_p}$.
- For the i-th particle evaluate the likelihood ratio

$$L_{\hat{t}^i_k|t^i_{k-1}}(\theta^i_k|\theta^i_{k-1}) = \frac{P_{\theta^h_{\hat{t}^i_k}|\theta^h_{t^i_{k-1}}}(\theta^i_k|\theta^i_{k-1})}{P_{\hat{\theta}^h_{\hat{t}^i_k}|\hat{\theta}^h_{t^i_{k-1}}}(\theta^i_k|\theta^i_{k-1})}.$$

- The particles, which do not reach the set $\bar{D}_k$ before time T are killed, i.e. we set $\hat{\omega}^i_k = 0$ if $\hat{t}^i_k = T$ and otherwise $\hat{\omega}^i_k = \omega^i_{k-1} \cdot L_{\hat{t}^i_k|t^i_{k-1}}(\theta^i_k|\theta^i_{k-1})$.

- The new set of particles is $\{\hat{t}_k^i, \hat{\xi}_k^i, \hat{\omega}_k^i\}_{i=1}^{N_p}$.
- Approximation of γ_k:

$$\gamma_k \approx \gamma_k^{N_p} = \sum_{i=1}^{N_p} \hat{\omega}_k^i.$$

- If all particles are killed, i.e. $\gamma_k^{N_p} = 0$, then the algorithm stops, and we set $P_{hit}(0,T) \approx 0$.

Algorithm 2.2^θ is almost the same as 2.1^θ only with "Uniform resampling" as described in Algorithm 1.2^θ.

5 Numerical evaluations

5.1 Switching diffusion example

Next we illustrate the performance of a Monte Carlo (MC) approach, the IPS algorithm of [Cerou et al 02] and the effect of the alternative steps 1H, 2H and 3H for a switching diffusion. Table 4 presents the list of tested algorithms. There Algorithms 1.1 and 1.2 are two versions of algorithm of [Cerou et al 02] in case of a switching diffusion (section 4.5); Algorithm 1.1^{θ} is Algorithm 1.1 with improved initial sampling step 1H and with resampling per mode step 3H (section 4.6); and Algorithm 2.1^{θ} is Algorithm 1.1^{θ} plus importance switching (section 4.8).

Table 4. Tested algorithms

Algorithm	Particle filtering	Initial sampling per mode	Resampling per mode	Uniform Resampling	Random Resampling	Importance Switching
MC	—	Yes (1H)	—	—	—	—
Algorithm 1.1	Yes	—	—	—	Yes	—
Algorithm 1.2	Yes	—	—	Yes	—	—
Algorithm 1.1^{θ}	Yes	Yes (1H)	Yes (3H)	—	Yes	—
Algorithm 2.1^{θ}	Yes	Yes (1H)	Yes (3H)	—	Yes	Yes (2H)

For example we extend the diffusion $\{x_t\}$ of (12) to a Markovian switching diffusion $(x_t, \theta_t) \in \mathbb{R} \times \{e_1, e_2, e_3\}$, the evolution of which is governed by the following SDE

$$\begin{aligned} dx_t &= (\mu(\theta_t) + \frac{\sigma(\theta_t)^2}{2})x_t dt + \sigma(\theta_t)x_t dW_t, \\ d\theta_t &= \int_{\mathbb{R}} c(\theta_{t-}, z)p(dt, dz). \end{aligned}$$

Initial conditions:

$$\begin{aligned} x_0 &= 1 \\ P_{\theta_0}(e_1) &= 1 - 10^{-7} - 10^{-9} \\ P_{\theta_0}(e_2) &= 10^{-7} \\ P_{\theta_0}(e_3) &= 10^{-9} \end{aligned}$$

Parameters:

$$\begin{aligned} \mu(e_1) &= 1, \quad \mu(e_2) = 4, \quad \mu(e_3) = 3, \\ \sigma(e_1) &= 1, \quad \sigma(e_2) = 0.9, \quad \sigma(e_3) = 1.7. \end{aligned}$$

$$c(e_i, z) = \begin{cases} e_j - e_i & \text{if } z \in \Delta_{ij} \\ 0 & \text{otherwise.} \end{cases}$$

here

$$\Delta_{ij}(x) = \Big[\sum_{i'=1}^{i-1} \sum_{\substack{j'=1 \\ j' \neq i'}}^{3} \lambda_{i'j'}(x) + \sum_{\substack{j'=1 \\ j' \neq i}}^{j-1} \lambda_{ij'}(x), \sum_{i'=1}^{i-1} \sum_{\substack{j'=1 \\ j' \neq i'}}^{3} \lambda_{i'j'}(x) + \sum_{\substack{j'=1 \\ j' \neq i}}^{j} \lambda_{ij'}(x) \Big), \ i, j = 1, 2, 3$$

$$\begin{array}{rclrcl} \lambda_{12} & = & 1 \cdot 10^{-4}, & \lambda_{13} & = & 1 \cdot 10^{-6}, \\ \lambda_{21} & = & 5 \cdot 10^{-5}, & \lambda_{23} & = & 1 \cdot 10^{-5}, \\ \lambda_{31} & = & 5 \cdot 10^{-5}, & \lambda_{32} & = & 1 \cdot 10^{-4}. \end{array}$$

We use the same decreasing sequence of levels $d = d_m \geq \cdots \geq d_1$, as in example for diffusion, and we run 1000 simulations with 1000 particles (500 for Mode 1, 300 for Mode 2 and 200 for Mode 3). For algorithm 2.1^θ the importance switching function $\hat{c}(\cdot)$ is constructed as $c(\cdot)$ but with $\hat{\lambda}_{ij} = \frac{1}{30}$ for $i \neq j$. The likelihood ratio $L_{t|s}(\theta|\theta')$ can be numerically evaluated as quotient of the transition probabilities of $\{\theta_t\}$ and $\{\hat{\theta}_t\}$. For this we numerically evaluate the matrix exponentials $e^{Q \cdot (t-s)}$ and $e^{\hat{Q} \cdot (t-s)}$, where Q and $\hat{Q}$ are the transition rate matrices of the continuous time Markov chains $\{\theta_t\}$ and $\{\hat{\theta}_t\}$ respectively.

As in the preceding example, we want to estimate the probability to hit barrier d before end time T, i.e. $P(\tau_d(x) \leq T)$ and also

$$P(\tau_d(x) \leq T, \theta_{\tau_d(x)} = \eta), \tag{34}$$

$\eta \in \mathbb{M}$, where $\tau_d(x) \stackrel{\triangle}{=} \inf\{t > 0 : x_t \in [d, +\infty);\, x_0 = x\}$. But in this case the evolution of continuous component x_t is more complicated, since it switches from one diffusion path to another as the discrete component θ_t jumps from one state to another. Due to this further complication, there is no analytical solution available for this case. Nevertheless, we can still make valuable estimates of these probabilities just by looking at conditional probabilities $P(\tau_d(x) \leq T,\ \theta_{\tau_d(x)} = \eta | \theta_s = \eta,\ s \in [0, \tau_d(x))\ ),\ \eta \in \mathbb{M}$, for which the analytical solution is know, i.e. equation (13). Note that in each fixed mode x_t is governed by the same equation as in example with diffusion, and with the same parameters when the mode is 1 (i.e. $\theta_t = e_1$).

Figure 5 illustrates the conditional probabilities $P(\tau_d(x) \leq T,\ \theta_{\tau_d(x)} = \eta | \theta_s = \eta,\ s \in [0, \tau_d(x))\ ),\ \eta \in \mathbb{M}$ as functions of d. We see that reaching the target level is more likely in mode 2 or 3 and less likely in mode 1. It is also obvious that higher number of particles will survive in modes 2 and 3 than in mode 1, but their weights can be relatively

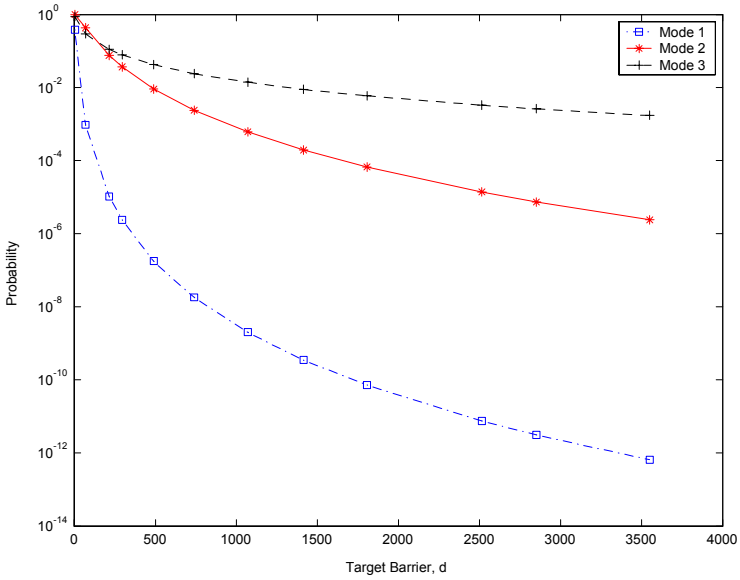

Figure 5: Probability to hit barrier d before time T=1.

small because of initial conditions. Table 5 shows the conditional probabilities to hit target d before time $T = 1$

$$P(\tau_d(x) \leq T, \theta_{\tau_d(x)} = \eta | \theta_s = \eta, s \in (0, \tau_d(x))) \cdot P(\theta_0 = \eta), \ \eta \in \mathbb{M} \tag{35}$$

for 12 different values of d. We can use this as a coarse reference for our simulations.

Table 5. Conditional probability to hit target level d before T=1; eq. (35)

Target Level (d)	Total Probability	Conditional Probability to hit in mode 1	Conditional Probability to hit in mode 2	Conditional Probability to hit in mode 3
5	3,84E-01	3,84E-01	9,98E-08	8,88E-10
70	9,56E-04	9,56E-04	4,33E-08	2,99E-10
215	1,06E-05	1,05E-05	7,58E-09	1,11E-10
295	2,38E-06	2,38E-06	3,68E-09	7,86E-11
490	1,79E-07	1,78E-07	9,17E-10	4,23E-11
740	1,83E-08	1,80E-08	2,40E-10	2,42E-11
1070	2,09E-09	2,02E-09	6,11E-11	1,40E-11
1415	3,80E-10	3,51E-10	1,95E-11	8,97E-12
1805	8,46E-11	7,19E-11	6,70E-12	5,97E-12
2515	1,23E-11	7,55E-12	1,40E-12	3,33E-12
2850	6,53E-12	3,14E-12	7,46E-13	2,64E-12
3550	2,63E-12	6,48E-13	2,38E-13	1,75E-12

In the following sections the performance of several algorithms is demonstrated and their relative merits are discussed.

5.2 Standard Monte Carlo

To begin with, we want to present the results obtained by running weighted standard Monte Carlo simulation. The simulation starts with 5000000 particles in Mode 1, 3000000 in Mode 2 and 2000000 in Mode 3, and each trial is assigned its weight (see step 1H in section 4.6). Then, each trial is independently simulated until it reaches barrier d or the end time T. Table 6 presents the results of 10^7 runs.

Table 6. Probability to hit barrier d
Standard Monte Carlo, 10,000,000 runs

Target Level (d)	Total Probability	Probability to hit in mode 1	Probability to hit in mode 2	Probability to hit in mode 3
5	3.74E-01	3.74E-01	6.35E-05	1.40E-06
70	8.39E-04	8.29E-04	9.64E-06	2.89E-10
215	1.02E-05	9.00E-06	1.21E-06	1.05E-10
295	2.60E-06	2.20E-06	4.03E-07	7.38E-11
490	4.01E-07	4.00E-07	7.82E-10	3.91E-11
740	2.18E-10	0.00E+00	1.96E-10	2.20E-11
1070	5.87E-11	0.00E+00	4.62E-11	1.25E-11
1415	2.18E-11	0.00E+00	1.38E-11	7.94E-12
1805	1.01E-11	0.00E+00	4.80E-12	5.27E-12
2515	3.38E-12	0.00E+00	5.00E-13	2.88E-12
2850	2.60E-12	0.00E+00	3.00E-13	2.30E-12
3550	1.57E-12	0.00E+00	6.67E-14	1.51E-12

This shows that it is not realistic to estimate rare events using the standard Monte Carlo (MC) method (compare tables 4 and 6). 10^7 runs of MC enable reasonable estimates to be made for events with probabilities of order 10^{-6}. The probability to reach the highest barrier $d = 3550$ in modes 2 and 3, given that we start in one of these modes, is higher than 10^{-6} (see figure 5). That is why MC shows rather better performance in these modes than in mode 1. When switching rates are low, an MC simulation also needs numerous trials to achieve good statistic. We conclude that MC method is very expensive in time and not appropriate for estimating event probabilities of less than 10^{-6}.

5.3 Switching Diffusion IPS algorithms 1.1 and 1.2

Next, Tables 7 and 8 present the results obtained by running Algorithms 1.1 & 1.2 for switching diffusion.

Table 7. Probability to hit barrier d
Algorithm 1.1. 1000 simulations of 1000 particles each

Target Level (d)	Total Probability	Probability to hit in mode 1	Probability to hit in mode 2	Probability to hit in mode 3
5	3.74E-01	3.74E-01	5.24E-05	0.00E+00
70	8.18E-04	8.15E-04	3.45E-06	1.57E-08
215	8.78E-06	8.57E-06	2.08E-07	6.22E-10
295	1.99E-06	1.92E-06	6.98E-08	1.90E-10
490	1.42E-07	1.34E-07	8.47E-09	6.28E-11
740	1.34E-08	1.27E-08	6.74E-10	1.26E-11
1070	1.34E-09	1.31E-09	2.66E-11	1.75E-12
1415	2.17E-10	2.15E-10	1.28E-12	2.34E-13
1805	3.33E-11	3.31E-11	9.11E-14	1.89E-14
2515	3.15E-12	3.15E-12	3.10E-15	8.74E-18
2850	1.21E-12	1.21E-12	7.06E-16	0.00E+00
3550	2.85E-13	2.85E-13	3.01E-17	0.00E+00

Table 8. Probability to hit barrier d
Algorithm 1.2. 1000 simulations of 1000 particles each

Target Level (d)	Total Probability	Probability to hit in mode 1	Probability to hit in mode 2	Probability to hit in mode 3
5	3.73E-01	3.73E-01	8.78E-05	0.00E+00
70	8.19E-04	8.03E-04	1.64E-05	0.00E+00
215	1.00E-05	8.27E-06	1.77E-06	0.00E+00
295	2.58E-06	1.85E-06	7.38E-07	0.00E+00
490	2.62E-07	1.30E-07	1.32E-07	0.00E+00
740	3.88E-08	1.23E-08	2.65E-08	0.00E+00
1070	6.08E-09	1.28E-09	4.80E-09	0.00E+00
1415	1.29E-09	2.12E-10	1.08E-09	0.00E+00
1805	2.44E-10	3.46E-11	2.10E-10	0.00E+00
2515	2.50E-11	3.46E-12	2.15E-11	0.00E+00
2850	8.31E-12	1.28E-12	7.03E-12	0.00E+00
3550	1.30E-12	2.67E-13	1.03E-12	0.00E+00

The results of both algorithms suggest that few if any particles are sampled in modes 2 and 3 at initial time $t = 0$, and after resampling steps they die very quickly. The sampling per mode can help to solve this problem.

5.4 IPS model and random resampling per mode Algorithm 1.1^{θ}

Our next results (Algorithm 1.1^{θ}) show how the situation can be improved by using "sampling per mode" for the initial sampling step and for the resampling step.

Table 9. Probability to hit barrier d
Algorithm 1.1^{θ}. 1000 simulations of 1000 particles each

Target Level (d)	Total Probability	Probability to hit in mode 1	Probability to hit in mode 2	Probability to hit in mode 3
5	3.73E-01	3.73E-01	7.75E-05	8.86E-10
70	8.12E-04	8.03E-04	9.49E-06	2.27E-08
215	9.25E-06	8.60E-06	6.55E-07	3.21E-10
295	2.15E-06	1.94E-06	2.16E-07	1.15E-10
490	1.76E-07	1.38E-07	3.81E-08	4.06E-11
740	1.83E-08	1.27E-08	5.57E-09	2.18E-11
1070	1.88E-09	1.31E-09	5.61E-10	1.23E-11
1415	3.16E-10	2.17E-10	9.11E-11	7.78E-12
1805	5.67E-11	3.68E-11	1.52E-11	4.77E-12
2515	7.99E-12	3.85E-12	1.54E-12	2.60E-12
2850	3.97E-12	1.51E-12	5.13E-13	1.95E-12
3550	1.78E-12	3.54E-13	1.15E-13	1.31E-12

Resampling per mode allows us to avoid loss of light particles in "light" modes, e.g. in mode 3, and helps to maintain fixed number of particles in each mode. Compare Tables 5 and 9. The results for mode 1 in these two tables are almost the same. But the probability to hit barrier d (especially for low values of d) in mode 2 differs from the conditional probabilities presented in Table 5. This means there were some switches from mode 1 to mode 2, i.e. heavy particles from mode 1 "arrived" to mode 2 before they have hit the level. It is important to understand that even rare switches may have considerable effect on final result. In our example switching rates are very low, so we should not expect many switches during simulation. We therefore believe that the probabilities of hitting the target barrier in some modes are underestimated.

5.5 IPS model and Importance Sampling Algorithm 2.1^{θ}

Next, we will see how the results change if we use algorithm 1.1^{θ} with the combination of Importance Switching technique (Algorithm 2.1^{θ}), i.e. when we make rare switches more

frequent. Table 10 presents results of Algorithm 2.1^{θ}.

The increase of the frequencies of switches has a considerable effect on probabilities of all modes and thus on total probability. This is what one should expect. First of all, at the beginning heavy particles from mode 1 can switch to mode 2 or mode 3, and thus, to increase the weights of these modes, which in turn increases the hitting probabilities in these modes. Second, particles in modes 2 and 3 approach the rare set faster than in mode 1, and if the switching to mode 1 occurs, then there is high chance that the particle is close to the target level and there is high chance that it will hit that level. Thus, there is no big surprise that we observe higher hitting probabilities in modes 1,2 and 3 after increasing frequencies of switchings.

This example shows how it is important to take care of rare switches, since it may have a substantial effect on total result. But it is not obvious how much we can change the frequencies (rates of θ's) without introducing perceptible distortion of the results. The tests of the Algorithm 2.1^{θ} had shown that it is stable with different choices of switching rates $\hat{\lambda}_{ij}$.

Table 10. Probability to hit barrier d
Algorithm 2.1^{θ}. 1000 simulations of 1000 particles each

Target Level (d)	Total Probability	Probability to hit in mode 1	Probability to hit in mode 2	Probability to hit in mode 3
5	4.04E-01	4.04E-01	4.82E-05	4.32E-07
70	1.47E-03	1.45E-03	1.23E-05	1.27E-07
215	2.34E-05	2.14E-05	1.93E-06	4.78E-08
295	6.52E-06	5.51E-06	9.74E-07	3.59E-08
490	8.14E-07	5.22E-07	2.70E-07	2.18E-08
740	1.63E-07	6.58E-08	8.25E-08	1.46E-08
1070	4.34E-08	8.65E-09	2.46E-08	1.02E-08
1415	1.87E-08	1.72E-09	9.29E-09	7.70E-09
1805	1.00E-08	3.96E-10	3.67E-09	5.94E-09
2515	5.54E-09	7.53E-11	1.18E-09	4.29E-09
2850	4.75E-09	4.04E-11	8.57E-10	3.85E-09
3550	3.67E-09	1.41E-11	4.75E-10	3.18E-09

5.6 Discussion of numerical results

Figures 6 through 8 present the estimated values of probabilities (34), obtained by running algorithms listed in Table 4. The estimated values of total hitting probability are given in figure 9.

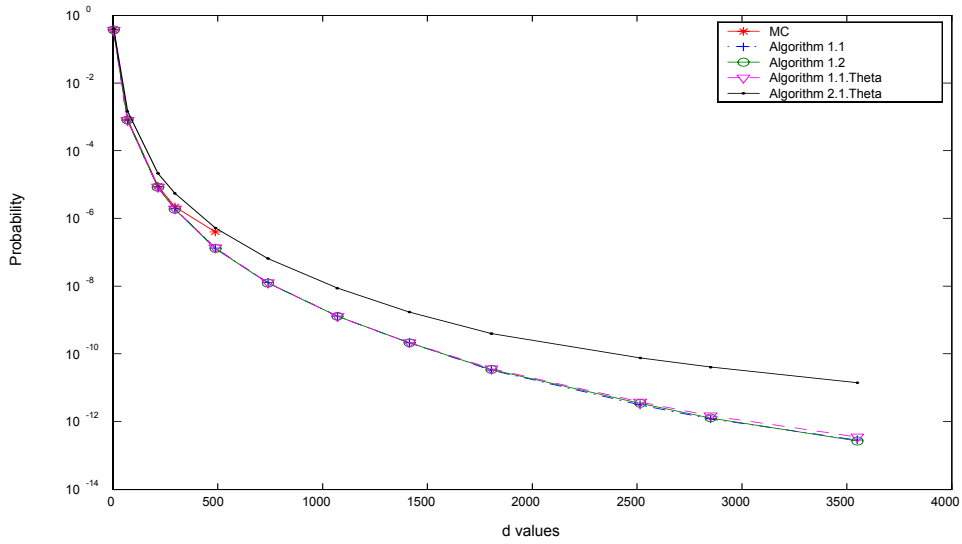

Figure 6: Probability to hit level d in Mode 1. MC stops at $d = 490$.

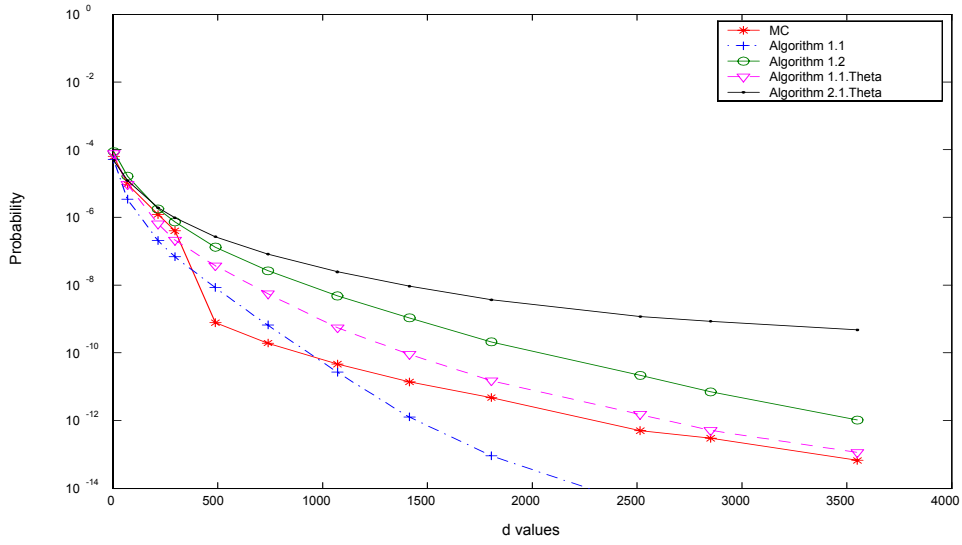

Figure 7: Probability to hit level d in Mode 2

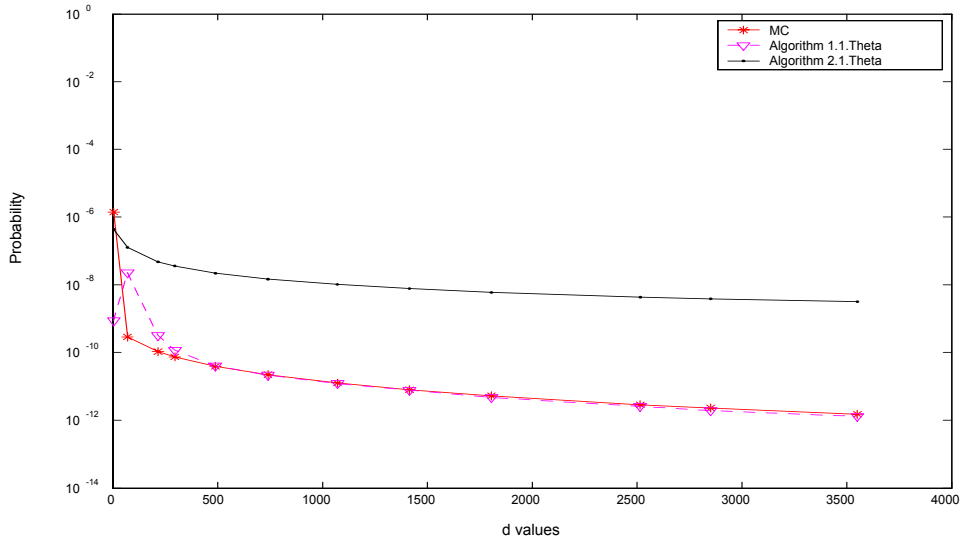

Figure 8: Probability to hit level d in Mode 3. Algorithms 1.1 and 1.2 stop at the beginning.

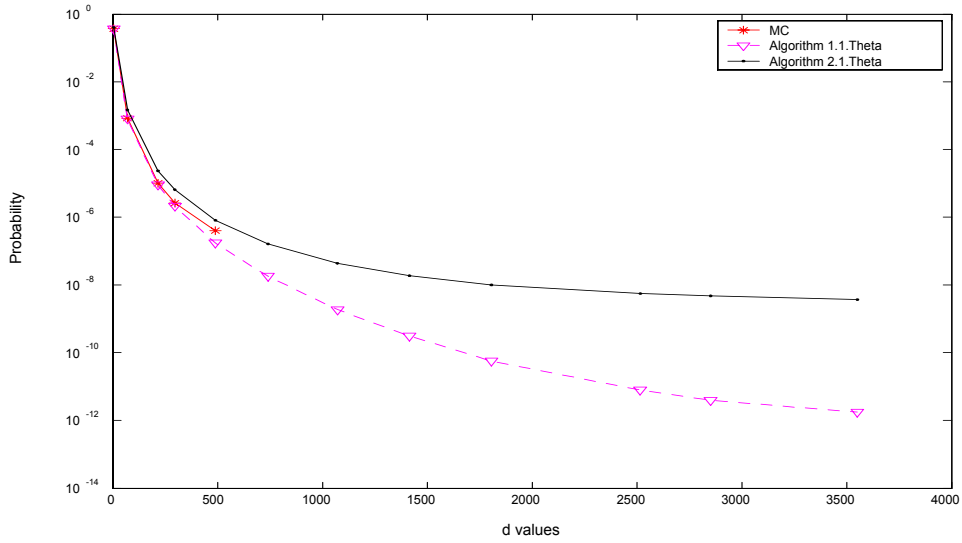

Figure 9: Total hitting probability of level d. MC stops at $d = 490$. Algorithms 1.1 and 1.2 stop at the beginning.

The results in figure 6 show that MC stops at $d = 490$ in Mode 1. This implies the same for the total probability in figure 9. For this example, Algorithms 1.1 and 1.2 stop immediately in Mode 3; particles with low weights simply died out immediately (figure 8).

The Algorithm 1.1^{θ} allows to avoid loss of light particles in "light" modes, e.g. in mode 3, and helps to maintain fixed number of particles in each mode. However, it is not able to cope with rare switches. It may happen that the switch itself is a rare event but it may have a significant influence on the total result. If this is the case (as in our example), then it is unlikely that there will be any switches during the simulation.

The Algorithm 2.1^{θ} is the best in handling both the problem of rare switches and problem of "light" modes. It forces interaction between the modes by making rare switches more frequent and properly adjusting the weights of particles. The results in figures 6 through 8 show that the increase of the frequencies of switches has a considerable effect on probabilities of mode 2 and 3 and thus on total probability (see figure 9). This is what one should expect because the heavy particles can leave the mode 1, although very rarely, and have a great influence on modes 2 and 3. The interaction between modes 2 and 3 is not really noticeable but they have an influence on mode 1.

6 Concluding Remarks

In this report we tried to extend the rare event sequential Monte Carlo approach of [Cerou et al 02] to switching diffusion situation. First, we have reformulated this approach as one of risk decomposition and filtering to estimate rare event probability for a switching diffusion. Then we have developed two extensions: sampling per mode to cope with large differences in mode weights, and importance switching to cope with rare mode switching. Next we evaluated the algorithms for two simple examples. First, we took a test example with analytical solution, i.e. geometric Brownian motion. The numerical test shows high efficiency and sufficient accuracy of the algorithm. However, as expected, it does not suffice when applied to switching diffusion. Two main pitfalls have been identified: rare switches and large difference of particle weights in different modes. The algorithms tested on switching diffusion are summarized in Table 4a. The obtained results show that all the proposed extensions are in fact needed for estimating rare events for a rarely switching diffusion. The best performing algorithm (Algorithm 2.1^{θ}) is able to cope with differences in weights (sampling per mode), rare switches of discrete component (importance switching) and rare visits of continuous component to the rare target set (decomposition of rare event probability).

There are many interesting questions left, such as the optimal selection of the nested cylinder sets, the number of these sets, the number of particles, proof of convergence and how the number of independent particle simulations influence the accuracy.

References

[Blom & Bloem 03a] Blom, H.A.P., Bloem, E.A., Joint IMMPDA Particle filter, Proceedings 6th Int. Conf on Information Fusion, Cairns, Queensland, Australia, 8-11 July 2003, pp. 785-792.

[Blom & Bloem 03b] Blom, H.A.P., Bloem, E.A., Tracking Multiple Maneuvering Targets by Joint Combinations of IMM and PDA, Preprint Proceedings 42^{nd} IEEE Conference on Decision and Control, December 9-12, 2003, Maui, Hawaii USA.

[Blom et al 03a] Blom, H.A.P., Bakker, G.J., Everdij M.H.C., M.N.J. van der Park, Stochastic analysis background of accident risk assessment for Air Traffic Management, Report Hybridge D2.2, July 29, 2003

[Blom et al 03b] Blom, H.A.P., Bakker, G.J., Everdij M.H.C., M.N.J. van der Park, Collision risk modeling of Air Traffic, Proceedings European Control Conference, Cambridge, UK, 2003

[Cerou et al 02] F. Cerou, P.D. Moral, F. LeGland, P. Lezaud, Genetic genealogical models in rare event analysis, Publications du Laboratoire de Statistiques et Probabilites, Toulous III, 2002.

[Del Moral 03] P. Del Moral, Feynman-Kac Formulae. Genealogical and Interacting Particle Systems with Applications, Springer, 2004.

[Ghosh et al 93] Ghosh, M.K., Arapostathis, A., Marcus, S.I., Optimal control of switching diffusions with application to flexible manufacturing systems, SIAM J. Control Optimization, Vol. 31, 1993, pp. 1183-1204.

[Ghosh et al 97] Ghosh, M.K., Arapostathis, A., Marcus, S.I., Ergodic control of switching diffusions, SIAM J. Control Optim., 35, 1997, pp. 1952-1988.

[Glasserman 03] Glasserman, P., Monte Carlo Methods in Financial Engineering, Stochastic Modelling and Applied Probability, Vol. 53, Springer, 2003.

[Hu et al 03] J.Hu, M.Prandini and S.Sastry, Probabilistic safety analysis in three-dimensional aircraft flight, 42nd Conf. on Decision and Control, Maui, Hawaii, USA, Dec. 2003.

[Kloeden & Platen 92] Kloeden, P.E., Platen, E., Numerical solution of stochastic differential equations, Springer-Verlag Berlin Heidelberg, 1992, 632p.

[Krystul et al 03a] J. Krystul, A.Bagchi and H.Blom, Risk decomposition and assessment methods, Report Hybridge D8.1, May, 2003.

[Krystul & Bagchi 03] J.Krystul and A.Bagchi, Weak approximation of first exit time of switching diffusion, working paper, 2003.

[Lezaud et al 03] P.Lezaud, J.Krystul and H.Blom, Accident Risk Assessment and Monte Carlo Simulation Methods, Report Hybridge D8.2, 2003.

[Reiss & Thomas 97] Reiss, R. D. & Thomas, M. *Statistical Analysis of Extreme Values.* Birkhäuser, 1997.

[Townsend et al 98] Townsend, J. D. & Haraszti, Z. & Freebersyser, J. A. & Devetsikiotis, M. *Simulation of Rare Events in Communications Networks.* in *IEEE Communications Magazine, August 1998*