On a new class of score functions to estimate tail probabilities of some stochastic processes with Adaptive Multilevel Splitting

Charles-Edouard Bréhier, Tony Lelièvre

▶ **To cite this version:**

Charles-Edouard Bréhier, Tony Lelièvre. On a new class of score functions to estimate tail probabilities of some stochastic processes with Adaptive Multilevel Splitting. Chaos: An Interdisciplinary Journal of Nonlinear Science, 2019, 29, pp.033126. 10.1063/1.5081440 . hal-01923385

HAL Id: hal-01923385
https://hal.science/hal-01923385v1

Submitted on 15 Nov 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

On a new class of score functions to estimate tail probabilities of some stochastic processes with Adaptive Multilevel Splitting

Charles-Edouard Bréhier[1, a)] and Tony Lelièvre[2, b)]

[1)] *Univ Lyon, CNRS, Université Claude Bernard Lyon 1, UMR5208, Institut Camille Jordan, F-69622 Villeurbanne, France*

[2)] *Universiteé Paris-Est, CERMICS (ENPC), INRIA, 6-8-10 avenue Blaise Pascal, 77455 Marne-la-Valleée, France*

(Dated: 14 November 2018)

We investigate the application of the Adaptive Multilevel Splitting algorithm for the estimation of tail probabilities of solutions of Stochastic Differential Equations evaluated at a given time, and of associated temporal averages.

We introduce a new, very general and effective family of score functions which is designed for these problems. We illustrate its behavior on a series of numerical experiments. In particular, we demonstrate how it can be used to estimate large deviation rate functionals for the longtime limit of temporal averages.

[a)] brehier@math.univ-lyon1.fr

[b)] lelievre@cermics.enpc.fr

I. INTRODUCTION

Fast and accurate estimation of rare event probabilities, and the effective simulation of these events, is a challenging computational issue, which appears in many fields of science and engineering. Since rare events are often the ones that matter in complex systems, designing efficient and easily implementable algorithms is a crucial question which has been the subject of many studies in the recent years.

Since the pionnering works on Monte-Carlo methods, several classes of algorithms have been developed, see for instance the monographs[3,15,41]. The most popular strategies are importance sampling and splitting. On the one hand, importance sampling consists in changing the probability distribution, such that under the new probability distribution the events of interest are not rare anymore. Appropriate reweighting then yields consistent estimators. This strategy has for instance been applied recently to simulate rare events in climate models[37]. On the other hand, splitting techniques, consist in writing the rare event probability as a product of conditional probabilities which are simpler to estimate, and in using interacting particle systems in order to estimate these conditional probabilities.

In this manuscript, a class of splitting algorithms is considered. Splitting techniques have been introduced in the 1950s[29], and have been studied extensively in the last two decades[16,20,25,26]. Many variants have appeared in the literature: Generalized multilevel splitting[6,7], RESTART[47,48], Subset simulation[4], Nested sampling[43,44], Reversible shaking transformations with interacting particle systems[1,27], genealogical particle analysis[21,49], etc...

The Adaptive Multilevel Splitting (AMS) algorithm[18] is designed to estimate rare event probabilities of the type $\mathbb{P}(\tau_B < \tau_A)$, where τ_A and τ_B are stopping times associated with a Markov process X, typically the entrance times of X in regions A and B of the state space. In many applications, A and B are metastable states for the process. The algorithm is based on selection and mutation mechanisms, which leads to the evolution of a system of interacting replicas. The selection is performed using a score function, which is often referred to as a reaction coordinate when dealing with metastable systems.

The objective of this article is to design and test new score functions, using the AMS strategy, to estimate probabilities of the type $\mathbb{P}(\Phi(X_T) > a)$ or $\mathbb{P}\left(\frac{1}{T}\int_0^T \phi(X_t)dt > a\right)$, where a is a threshold, Φ and ϕ are real-valued functions, and $\left(X_t\right)_{0\leq t\leq T}$ is a Markov process. In fact, as will be explained below, the probabilities of interest can be rewritten as $\mathbb{P}(\tau_B < \tau_A)$,

associated with an auxiliary Markov process. Our main contribution is the identification of appropriate score functions related to this interpretation, and which return a non-zero value for the estimator of the probability of interest. By using then an AMS algorithm which fits in the Generalized Adaptive Multilevel Splitting framework[11], one can construct unbiased estimators of the probability (and possibly of other quantities of interest).

The efficiency of the approach is investigated with numerical experiments, using several test cases taken from the literature on rare events. First, validation is performed on one-dimensional Gaussian models (Brownian Motion[46], Ornstein-Uhlenbeck process[49]). More complex test cases then illustrate the efficiency of the approach and of the new score functions introduced in this article: drifted Brownian Motion, three-dimensional Lorenz model[5,32]. Estimations of probabilities depending on temporal averages $\frac{1}{T}\int_0^T \phi(X_s)ds$ are also considered for two models[24]: the one-dimensional Ornstein-Uhlenbeck process and a driven periodic diffusion. In these examples, values of large deviations rate functionals for the longtime limit $T \to \infty$ are estimated.

In the last decade, many works have been devoted to the analysis and applications of AMS algorithms. A series of work has been devoted to the analysis of the so-called ideal case[9,13,14,28], namely when the AMS algorithm is applied with the optimal score function (namely the so-called committor function). In practice, this optimal score function is unkown. Beyond the ideal case, consistency[11] (unbiasedness of the small probability estimator) and efficiency[17] (variance of the small probability estimator) have been studied. Moreover, the adaption of the original algorithm to the discrete-in-time setting has been studied in details in[11]. It can be used to compute transition times between metastable states[19], return times[30], or other observables associated with the rare event of interest[34]. The AMS algorithm has been successfully applied in many contexts: the Allen-Cahn stochastic partial differential equation[12,39], the simulation of Bose-Einstein condensates[36], molecular dynamics and computational chemistry[2,19,31,45], nuclear physics[33–35] and turbulence[8,38], for example.

This article is organized as follows. Section II presents the precise mathematical setting, in particular the rare event probability of interest is defined by (7). A general formulation of the AMS algorithm designed to estimate this quantity is provided in Section III, in particular see Section III B for the full algorithmic description. Examples of appropriate score functions are discussed in Section IV. To overcome the limitations of a vanilla strategy, Section IV A, our main contribution is the construction of the score functions presented in Section IV B.

Finally, numerical experiments are reported in Section V.

II. SETTING

We consider stochastic processes, with values in $\mathbb{R}^d$, in dimension $d \in \mathbb{N}$, which are solutions of Stochastic Differential Equations (SDEs) of the type: for $0 \leq t_0 \leq t \leq T$ and $x_0 \in \mathbb{R}^d$,

$$dX_t^{t_0,x_0} = f(t, X_t^{t_0,x_0})dt + \sigma(t, X_t^{t_0,x_0})dW(t) \tag{1}$$

where $X_t^{t_0,x_0} \in \mathbb{R}^d$, with initial condition,

$$X_{t_0}^{t_0,x_0} = x_0. \tag{2}$$

The noise $\big(W(t)\big)_{t\geq 0}$ is given by a standard Wiener process with values in $\mathbb{R}^D$, for some $D \in \mathbb{N}$. The coefficients $f : [0,T] \times \mathbb{R}^d \to \mathbb{R}^d$ and $\sigma : [0,T] \times \mathbb{R}^d \to \mathbb{R}^{d\times D}$ are assumed to be sufficiently smooth to ensure global well-posedness of the SDE.

In this work, two types of rare events associated with $(X_t^{t_0,x_0})_{0\leq t\leq T}$ are considered. Let $a \in \mathbb{R}$ denote a threshold, and let $\Phi, \phi : \mathbb{R}^d \to \mathbb{R}$ be two measurable functions. First, we are interested in tail probabilities for the random variable $\Phi(X_T^{t_0,x_0})$, namely in

$$\mathbb{P}\big(\Phi(X_T^{t_0,x_0}) > a\big). \tag{3}$$

Second, we are interested in tail probabilities for temporal averages, defined as

$$\mathbb{P}\left(\frac{1}{T-t_0}\int_{t_0}^{T} \phi(X_t^{t_0,x_0})dt > a\right). \tag{4}$$

We will investigate numerically the performance of AMS estimators for both (3) and (4) on various examples. In particular, we will consider the regime $T \to \infty$ for (4) in order to estimate large deviation rate functionals.

Notice that the case of temporal averages (4) can be rewritten in the form of (3). Indeed, the probability (4) may be written as (3) for the auxiliary process defined by $\tilde{X}_t^{t_0,x_0} = \big(X_t^{t_0,x_0}, Y_t^{t_0,x_0}\big)$, where $Y_t^{t_0,x_0} = \phi(x_0)$, and for $t > t_0$

$$Y_t^{t_0,x_0} = \frac{1}{t-t_0}\int_{t_0}^{t} \phi(X_s^{t_0,x_0})ds,$$

and with $\tilde{\Phi}(x,y) = y$. The process $\big(Y_t^{t_0,x_0}\big)_{t_0\leq t\leq T}$ is solution of the following ODE,

$$dY_t^{t_0,x_0} = \frac{1}{t-t_0}\big(\phi(X_t^{t_0,x_0}) - Y_t^{t_0,x_0}\big),\ t > t_0 \quad ,\ Y_{t_0}^{t_0,x_0} = \phi(x_0),$$

coupled with the SDE for the diffusion process $\left(X_t^{t_0,x_0}\right)_{t_0\leq t\leq T}$. This trick will be used for our numerical experiments below. Therefore, in the following, we present the AMS algorithm and discuss its theoretical properties only for (3).

For future purposes, observe that the target probability (3) may be written as $u_a(t_0,x_0)$, where

$$u_a(t,x)=\mathbb{P}\big(\Phi(X_T^{t,x})>a\big) \tag{5}$$

is the solution (under appropriate regularity assumptions) of the backward Kolmogorov equation

$$\begin{cases}\frac{\partial u_a(t,x)}{\partial t}+\mathcal{L}_t u_a(t,x)=0 \text{ for } t\in[0,T] \text{ and } x\in\mathbb{R}^d,\\ u_a(T,x)=\mathbb{1}_{\Phi(x)>a} \text{ for } x\in\mathbb{R}^d,\end{cases} \tag{6}$$

where the infinitesimal generator $\mathcal{L}_t$ is defined by: for all test functions φ, $\mathcal{L}_t\varphi(x)=f(t,x)\cdot\nabla\varphi(x)+\frac{1}{2}\sigma(t,x)\sigma(t,x)^\star:\nabla^2\varphi(x)$. Approximating the solutions of PDEs of this type using deterministic methods is in general possible only when the dimension d is small. Instead, Monte Carlo methods may be used. However, naive Monte Carlo algorithms are not efficient in the rare event regime, *e.g.* when $a\to\infty$ or when the diffusion coefficient is of the type $\sigma_\epsilon=\sqrt{\epsilon}\sigma$ and $\epsilon\to 0$.

In practice, discrete-time approximations are implemented. Let $\Delta t>0$ denote the time-step size of the integrator (for instance the standard Euler-Maruyama method), with $T=N\Delta t$ and $t_0=n_0\Delta t$, where $n_0\in\mathbb{N}_0, N\in\mathbb{N}$, $n_0\leq N$. With a slight abuse of notation, let us denote the discrete-time process obtained after discretization of (1) by $\left(X_n^{n_0,x_0}\right)_{n_0\leq n\leq N}$. The time-discrete counterpart of (3) is then

$$\mathbb{P}\big(\Phi(X_N^{n_0,x_0})>a\big). \tag{7}$$

The algorithms presented below are used to estimate probabilities of the type (7).

Remark 1. *It is assumed that the initial condition is deterministic: $X_{n_0}^{n_0,x_0}=x_0$. The adaptation of the algorithms presented below to the case of a random initial condition is straightforward, by simply using the Markov property: $\mathbb{P}\big(\Phi(X_N)>a\big)=\displaystyle\int\mathbb{P}\big(\Phi(X_N^{n_0,x_0})>a\big)d\mu_0(x_0)$ where μ_0 denotes the law of X_{n_0}.*

III. GENERAL FORMULATION OF THE ADAPTIVE MULTILEVEL SPLITTING ALGORITHM

A. Context

The goal is to estimate the probability p given by (7), in the regime where p is small, which is for example the case when a is large.

It is convenient to introduce an auxiliary process $\big(Z_n\big)_{n_0\leq n\leq N}$, such that $Z_n = \big(n\Delta t, X_n^{n_0,x_0}\big)$. Indeed, let

$$A = \{(T,x);\ \Phi(x)\leq a\}\,, B = \{(T,x);\ \Phi(x) > a\}\,,$$

and define the associated stopping times

$$\tau_A = \inf\{n\geq n_0, n\in\mathbb{N}_0;\ Z_n\in A\}\,, \tau_B = \inf\{n\geq n_0, n\in\mathbb{N}_0;\ Z_n\in B\}\,.$$

Then the probability p given by (7) can be rewritten as

$$p = \mathbb{P}\big(\Phi(X_N^{n_0,x_0}) > a\big) = \mathbb{P}(\tau_B < \tau_A). \tag{8}$$

We are then in position to build algorithms which fit in the Generalized Adaptive Multilevel Splitting framework developed in[11], which ensure that the obtained estimators of the probability (8) are unbiased.

For that, a score function, or reaction coordinate, ξ, needs to be given. Following the interpretation above, ξ may depend on $z = (n\Delta t, x)$.

To run the algorithm and define simple unbiased estimators of p, only one requirement is imposed on the function ξ: there exists $\xi_{\max}$ such that

$$B\subset\{z;\ \xi(z) > \xi_{\max}\}\,,$$

which in the context of this article is rephrased as

$$\Phi(x) > a \implies \xi(T,x) > \xi_{\max}. \tag{9}$$

The principle of splitting algorithm is then to write

$$\mathbb{P}(\tau_B < \tau_A) = \mathbb{P}(\tau_{\zeta_1} < \tau_A)\mathbb{P}(\tau_{\zeta_2} < \tau_A|\tau_{\zeta_1} < \tau_A)\mathbb{P}(\tau_{\zeta_3} < \tau_A|\tau_{\zeta_2} < \tau_A)\ldots\mathbb{P}(\tau_B < \tau_A|\tau_{\xi_{\max}} < \tau_A)$$

for an increasing sequence of levels $(\zeta_q)_{q\geq 1}$, where $\tau_\zeta = \inf\{n\geq n_0;\ \xi(Z_n) > \zeta\}$. If the levels are well chosen, then the successive conditional probabilities $\mathbb{P}(\tau_{\zeta_{q+1}} < \tau_A|\tau_{\zeta_q} < \tau_A)$ are easy

to compute. The principle of the adaptive multilevel splitting algorithm[18] is to choose the levels adaptively, so that the successive conditional probabilities $\mathbb{P}(\tau_{\zeta_{q+1}} < \tau_A | \tau_{\zeta_q} < \tau_A)$ are constant and fixed. The levels constructed in the algorithm are then random.

B. The Adaptive Multilevel Splitting algorithm

Before giving the detailed algorithm, let us roughly explain the main steps (we also refer to[11] for more details and intuition on the algorithm). In the initialization, one samples $n_{\rm rep}$ trajectories following (1)-(2) and compute the score of each trajectory, namely the maximum of ξ attained along the path. Then the algorithm proceeds as follows: one discards the trajectory which has the smallest score and in order to keep the number of trajectories fixed, a new one is created by choosing one of the remaining trajectories at random, copying it up to the score of the killed trajectory, and sampling the end of trajectory independently from the past. This is called the partial resampling. One thus obtains a new ensemble of $n_{\rm rep}$ trajectories on which one can iterate by again discarding the the trajectory which has the smallest score. As the iteration goes, one thus obtains trajectories with largest and largest scores, and an estimate of the probability of interest is obtained as $(1 - 1/n_{\rm rep})^{Q_{\rm iter}} P(\tau_B < \tau_A | \tau_{\xi_{\max}} < \tau_A)$ (notice that $(1 - 1/n_{\rm rep})$ is an estimate of the conditional probability to reach level ζ_{q+1} conditionally to the fact that level ζ_q has been reached), where $Q_{\rm iter}$ is the number of iterations required to reach the maximum level $\xi_{\max}$. In practice, $P(\tau_B < \tau_A | \tau_{\xi_{\max}} < \tau_A)$ is estimated by the proportion of trajectories which reach B before A at the last iteration of the algorithm, namely when all the trajectories satisfy $\tau_{\xi_{\max}} < \tau_A$.

Actually, the algorithm has to be adapted in order to take into account situations when more than one particle has the smallest score, which happens with non zero probability for Markov chains. Let us now give the details of the AMS algorithm.

To simplify notation, in the sequel, the initial condition x_0 and the time n_0 are omitted in the notation of the replicas.

a. Input

- $n_{\rm rep} \in \mathbb{N}$, the number of replicas,
- a score function $z = (n\Delta t, x) \mapsto \xi(n\Delta t, x) \in \mathbb{R}$ and a stopping level $\xi_{\max} \in \mathbb{R}$ such that (9) is satisfied.

b. Initialization

- Sample $n_{\rm rep}$ independent realizations of the Markov process

$$X^j = \left(X_m^j\right)_{n_0 \le m \le N},\ 1 \le j \le n_{\rm rep}$$

  following the dynamics (1)–(2).

- Compute the score of each replica, $M^j = \max\limits_{n_0 \le m \le N} \xi(m\Delta t, X_m^j)$.

- Compute the level $\mathcal{Z} = \min\limits_{1 \le j \le n_{\rm rep}} M^j$.

- Define $\mathcal{K} = \{j \in \{1, \dots, n_{\rm rep}\}\ ;\ M^j = \mathcal{Z}\}$.

- Set $q = 0$, $\hat{p} = 1$, $\mathcal{B} = 1$.

c. Stopping criterion **If** $\mathcal{Z} \ge \xi_{\max}$ or $\mathrm{card}(\mathcal{K}) = n_{\rm rep}$, **then** set $\mathcal{B} = 0$.

d. While $\mathcal{B} = 1$

- **Update**

  - $q \leftarrow q + 1$ and $\hat{p} \leftarrow \hat{p} \cdot \left(1 - \frac{\mathrm{card}(\mathcal{K})}{n}\right)$.

- **Splitting**

  - Reindex the replicas, such that

$$\begin{cases} M^j = \mathcal{Z} & \text{if } j \in \{1, \dots, \mathrm{card}(\mathcal{K})\} \\ M^j > \mathcal{Z} & \text{if } j \in \{\mathrm{card}(\mathcal{K}) + 1, \dots, n_{\rm rep}\}\,. \end{cases}$$

  - For replicas with index $j \in \{1, \dots, \mathrm{card}(\mathcal{K})\}$, sample labels $\ell_1, \dots, \ell_{\mathrm{card}(\mathcal{K})}$, independently and uniformly in $\{\mathrm{card}(\mathcal{K}) + 1, \dots, n_{\rm rep}\}$.

- **Partial resampling**

  - Remove the replicas with label $j \in \{1, \dots, \mathrm{card}(\mathcal{K})\}$.
  - For $j \in \{1, \dots, \mathrm{card}(\mathcal{K})\}$, define $m_j = \inf\left\{m \in \{n_0, \dots, N\}\ ;\ \xi(X_m^{\ell_j}) > \mathcal{Z}\right\}$.
  - For $m \in \{n_0, \dots, m_j\}$, set $X_m^j = X_m^{\ell_j}$.
  - Sample a new trajectory $\left(X_m^j\right)_{m_j \le m \le N}$ with the Markov dynamics (1) driven by independent realizations of the Brownian motion.

- **Level computation**
  - Compute the scores $M^j = \max_{n_0 \le m \le N} \xi(m\Delta t, X_m^j)$.
  - Compute the level $\mathcal{Z} = \min_{1 \le j \le n_{\text{rep}}} M^j$.
  - Define the set $\mathcal{K} = \{j \in \{1, \dots, n_{\text{rep}}\} \ ; \ M^j = \mathcal{Z}\}$.

e. Stopping criterion **If** $\mathcal{Z} \ge \xi_{\max}$ or $\text{card}(\mathcal{K}) = n_{\text{rep}}$, **then** set $\mathcal{B} = 0$.

f. End while

g. Update: $\hat{p} \leftarrow \hat{p} \frac{1}{n_{\text{rep}}} \sum_{j=1}^{n_{\text{rep}}} \mathbb{1}_{\Phi(X_N^j) \ge a}$.

h. Output: $\hat{p}$ and $Q_{\text{iter}} = q$.

Remark 2. *We presented the algorithm in its simplest form. There are many variants, see*[11]*. For example, the killing level* $\mathcal{Z}$ *can be defined as* $\mathcal{Z} = M^{(k)}$ *where* $M^{(1)} \le M^{(2)} \le \dots \le M^{(n_{\text{rep}})}$ *denotes an increasing relabelling of the scores* $(M^j)_{1 \le j \le n_{\text{rep}}}$ *(order statistics).*

C. Consistency result

Let $\hat{p}$ and Q_{iter} be the outputs of the realization of the algorithm above. We quote the following result[11], which states that the output $\hat{p}$ of the algorithm described in Section III B above is an unbiased estimator of the probability given by (7).

Theorem 1. *Let* ξ *be a score function and* $\xi_{\max} \in \mathbb{R}$ *be such that* (9) *is satisfied. Let* $n_{\text{rep}} \in \mathbb{N}$ *be a given number of replicas. Assume that almost surely the algorithm stops after a finite number of iterations:* $Q_{\text{iter}} < \infty$ *almost surely.*

Then $\hat{p}$ *is an unbiased estimator of the probability* p *given by* (8)*:*

$$\mathbb{E}[\hat{p}] = \mathbb{P}\big(\Phi(X_N^{n_0,x_0}) > a\big).$$

Note that if $\psi : \mathbb{R}^d \to \mathbb{R}$ is a function with support included in $\{x;\ \Phi(x) > a\}$, *i.e.* $\psi(x) = 0$ if $\Phi(x) \le a$, then an unbiased estimator of $\mathbb{E}[\psi(X_N^{n_0,x_0})]$ is given by replacing the final update in the algorithm above (see step *g.*), by

$$\hat{p} \leftarrow \hat{p} \frac{1}{n_{\text{rep}}} \sum_{j=1}^{n_{\text{rep}}} \psi(X_N^j).$$

The unbiasedness property is crucial in practice for the following two reasons. First, it is very easy to parallelize the estimation of rare events using this property. Indeed, since the

estimator is unbiased whatever the value of $n_{\rm rep}$, to get a convergent estimation, one has simply to fix $n_{\rm rep}$ to a value which enables the computation of $\hat{p}$ on a single CPU, and then to sample M independent realizations of this $\hat{p}$, run in parallel. In the large M limit, one obtains a convergent estimator of the quantity of interest by simply considering the average of the realizations of $\hat{p}$. Second, the practical interest of Theorem 1 is that since $\mathbb{E}(\hat{p})$ is the same whatever the choice of the numerical parameters (namely $n_{\rm rep}$ and ξ), one can compare the results obtained with different choices to get confidence in the result. For example, one can consider the confidence intervals obtained with M independent realizations of $\hat{p}$ for two different choices of ξ, and check whether these confidence intervals overlap or not.

Remark 3. *In the algorithm described in Section III B, the set $\mathcal{K}$ defined in the initialization and in the level computation steps may have a cardinal strictly larger than* 1*, even if the level $\mathcal{Z}$ is defined as the minimum of the scores over the replicas. This simply means than more than one replica has a score which is the smallest among the replicas. In the discrete-time setting (namely for Markov chains), this happens with non zero probability, and it requires an appropriate modification of the original AMS algorithm, as described above, see*[11] *for more details.*

Notice that in particular, there is a possibility that the algorithm stops if $\mathrm{card}(\mathcal{K}) = n_{\rm rep}$*, in which case there is an extinction of the system of replicas.*

IV. CHOICES OF THE SCORE FUNCTION

Let us now describe the various score functions we will consider in order to estimate (7).

A. Vanilla score function and limitations

The simplest choice consists in choosing the score function as given by

$$\xi_{\rm std}(n\Delta t, x) = \Phi(x), \tag{10}$$

with $\xi_{\max} = a$. In this case, the score function does not depend on the time variable.

If the conditional probability

$$q = \mathbb{P}\big(\Phi(X_N^{n_0,x_0}) > a \;\big|\; \max_{n_0 \le n \le N} \Phi(X_n^{n_0,x_0}) > a\big)$$

is small, the performance of this vanilla strategy may be poor. Indeed, only a small proportion of the replicas have a non-zero contribution to the value of the estimator of the probability (7). It may even happen that all the replicas satisfy $\max_{n_0\le n\le N} \Phi(X_n^{n_0,x_0}) > a$ but that none of them satisfies $\Phi(X_N^{n_0,x_0}) > a$. In that situation, the algorithm returns $\hat{p} = 0$ to estimate $p > 0$. One of the goals of this work is to construct score functions which circumvent that issue: with the score functions introduced below, almost surely $\hat{p} \neq 0$.

B. Time-dependent score functions

As discussed above, it is natural to design score functions ξ, which satisfy the following condition:

$$\{\Phi(X_N^{n_0,x_0}) > a\} = \left\{ \max_{n_0\le n\le T} \xi(n\Delta t, X_n^{n_0,x_0}) > 1 \right\}. \tag{11}$$

The choice of the value $\xi_{\max} = 1$ on the right-hand side above is arbitrary, but no generality is lost. Indeed, if a score function ξ satisfying (11) is used in the AMS algorithm above, with $\xi_{\max} = 1$, at the last update, the ratio $\frac{1}{n_{\rm rep}} \sum_{j=1}^{n_{\rm rep}} \mathbb{1}_{\Phi(X_N^j)\ge a}$ is identically equal to 1, since all replicas satisfy $\max_{n_0\le n\le T} \xi(n\Delta t, X_n^{n_0,x_0}) > 1$. In particular, by construction $\hat{p} \neq 0$ (provided $Q_{\rm iter} < \infty$).

As will be seen below, in practice it is more natural to identify functions $\tilde{\xi}$ taking values in $(-\infty, 1]$, which satisfy the condition

$$\{\Phi(X_N^{n_0,x_0}) > a\} = \left\{ \max_{n_0\le n\le T} \tilde{\xi}(n\Delta t, X_n^{n_0,x_0}) = 1 \right\}, \tag{12}$$

instead of (11). To justify the use of the algorithm in this case, observe that $\bar{\xi}(t,x) = \tilde{\xi}(t,x) + \mathbb{1}_{\tilde{\xi}(t,x)=1}$ then satisfies (11). In addition, when running the algorithm, choosing either $\bar{\xi}$ or $\tilde{\xi}$ exactly yields the same result. We are thus in the setting where the unbiasedness result Theorem 1 applies. The score functions presented below will satisfy (12) instead of (11).

One of the novelties of this article is the introduction of the following score function:

$$\xi_{\rm new}(n\Delta t, x) = \big(\Phi(x) - a\big)\mathbb{1}_{\Phi(x)\le a} + \frac{n\Delta t}{N\Delta t}\mathbb{1}_{\Phi(x)>a}. \tag{13}$$

Observe that $\xi_{\rm new}$ takes values in $(-\infty, 1]$, and that $\xi_{\rm new}(n\Delta t, x) = 1$ if and only if $n = N$ and $\Phi(x) \ge a$. Thus the condition (12) is satisfied. We refer to Figure 1 for a schematic representation of this score function.

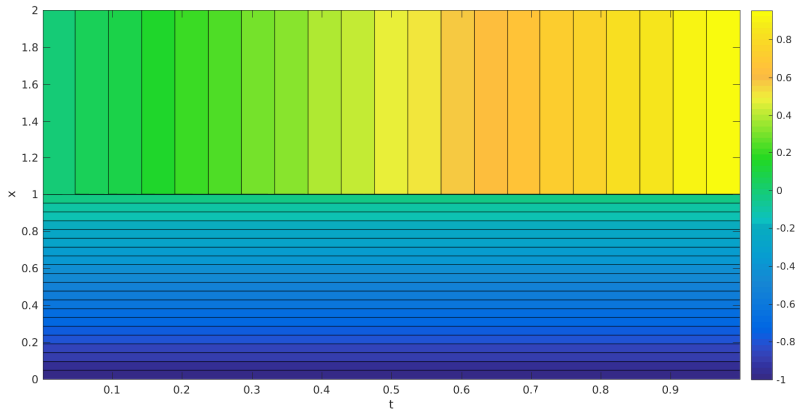

FIG. 1. Level lines of the score function $(t,x) \mapsto \xi_{\text{new}}(t,x)$, with $\Phi(x) = x$, $a = 1$, $T = 1$.

Note that the score function defined by (13) only depends on the function Φ, on the threshold a, and on the final time $T = N\Delta t$. It may thus be applied in any situation, but in some cases better score functions may be built upon using more information on the dynamics. The practical implementation is very simple.

Let us explain how the AMS algorithm proceeds when used with the score function (13). Observe that

$$\left\{ \max_{n_0 \le n \le N} \xi_{\text{new}}(n\Delta t, X_n^{n_0,x_0}) \ge 0 \right\} = \left\{ \max_{n_0 \le n \le N} \Phi(X_n^{n_0,x_0}) > a \right\}.$$

The first iterations of the algorithm, up to reaching level 0, are thus devoted to construct n_{rep} replicas which satisfy the weaker condition $\left\{ \max_{n_0 \le n \le N} \Phi(X_n^{n_0,x_0}) > a \right\}$. In other words, if the stopping level $\xi_{\max}$ in the algorithm is set equal to 0 instead of 1, one thus recovers the vanilla AMS algorithm described above, applied with the score function $\xi(t,x) = \Phi(x)$ (independent of time t).

Compared with the vanilla score function, the AMS algorithm the new score function does not stop when $\left\{ \max_{n_0 \le n \le N} \Phi(X_n^{n_0,x_0}) > a \right\}$. In terms of splitting, observe that this consists in writing

$$\mathbb{P}\big(\Phi(X_N^{n_0,x_0}) > a\big) = \mathbb{P}\big(\Phi(X_N^{n_0,x_0}) > a \mid \max_{n_0 \le n \le N} \Phi(X_n^{n_0,x_0}) > a\big)\mathbb{P}\big(\max_{n_0 \le n \le N} \Phi(X_n^{n_0,x_0}) > a\big),$$

and the remaining effort consists in estimating the conditional probability above.

More generally, observe that for every $n_1 \in \{n_0, \dots, N\}$,

$$\left\{ \max_{n_0 \le n \le N} \xi_{\text{new}}(n\Delta t, X_n^{n_0,x_0}) \ge \frac{n_1}{N} \right\} = \left\{ \max_{n_1 \le n \le N} \Phi(X_n^{n_0,x_0}) > a \right\}.$$

The construction of the score function (13) is associated to the following family of nested events:

$$\left\{\max_{n_0\le n\le N}\xi_{\text{new}}(n\Delta t, X_n^{n_0,x_0})\ge 1\right\}\subset\ldots\subset\left\{\max_{n_0\le n\le N}\xi_{\text{new}}(n\Delta t, X_n^{n_0,x_0})\ge \frac{n_1}{N}\right\}$$
$$\subset\ldots\subset\left\{\max_{n_0\le n\le N}\xi_{\text{new}}(n\Delta t, X_n^{n_0,x_0})\ge 0\right\},$$

which equivalently may be rewritten as

$$\{\Phi(X_N^{n_0,x_0})>a\}\subset\ldots\subset\left\{\max_{n_1\le n\le N}\Phi(X_n^{n_0,x_0})>a\right\}$$
$$\subset\ldots\subset\left\{\max_{n_0\le n\le N}\Phi(X_n^{n_0,x_0})>a\right\}.$$

In the selection procedure, the intervals $[n_1\Delta t, N\Delta t]$ are iteratively reduced (by increasing the left end point of the interval), until they ultimately contain only the point $N\Delta t$ (the right end point of the interval which remains fixed).

To conclude, we mention that the construction given by (13) can be generalized as follows. Let $\mathbf{a}:[0,T]\to\mathbb{R}$ be a non-decreasing function, such that $\mathbf{a}(T)=a$. Define

$$\xi_{\text{new}}^{\mathbf{a}}(n\Delta t,x)=\big(\Phi(x)-\mathbf{a}(n\Delta t)\big)\mathbb{1}_{\Phi(x)\le\mathbf{a}(n\Delta t)}+\frac{n\Delta t}{N\Delta t}\mathbb{1}_{\Phi(x)>\mathbf{a}(n\Delta t)}. \tag{14}$$

The score function ξ_{new} defined by (13) is a particular case of (14), with $\mathbf{a}(t)=a$. Optimizing the choice of the function $\mathbf{a}$ may help improve the efficiency of the algorithm. Notice that condition (12) is satisfied with $\xi=\xi_{\text{new}}^{\mathbf{a}}$. We refer to Figure 2 for a schematic representation of this score function.

C. The optimal score function: the committor function

For the general setting presented in Section III A where ones want to estimate $\mathbb{P}(\tau_B<\tau_A)$, the committor function is defined as $z\mapsto\mathbb{P}^z(\tau_B<\tau_A)$, where the upperscript z refers to the initial condition of the process Z. In[17], it is shown that, in a continous-time setting, the asymptotic variance (as the number of replicas n_{rep} goes to infinity) of AMS algorithm is minimized when using the committor function as the score function. It is thus interesting to look at what the committor function looks like in our context.

In our context, the committor function is given by

$$\xi_{\text{com}}(n\Delta t,x)=\mathbb{P}\big(\Phi(X_N^{n,x})>a\big). \tag{15}$$

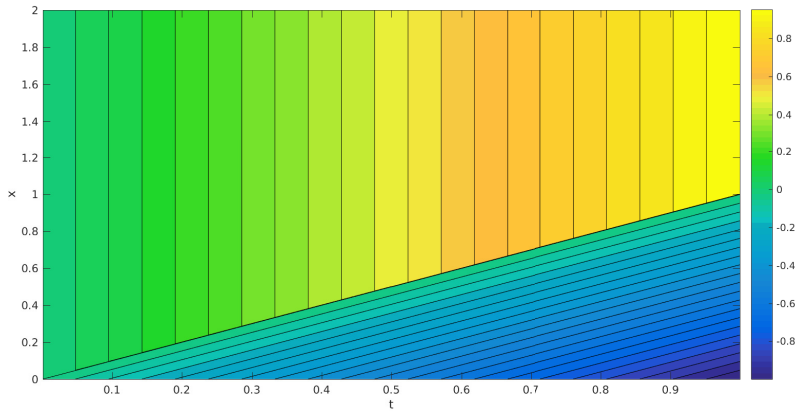

FIG. 2. Level lines of the score function $(t,x) \mapsto \xi^{\mathbf{a}}_{\text{new}}(t,x)$, with $\mathbf{a}(t) = atT^{-1}$, and $\Phi(x) = x$, $a = 1$, $T = 1$.

For the discussion, it is more convenient to consider the continuous-time version $\xi_{\text{com}}(t,x) = \mathbb{P}\big(\Phi(X_T^{t,x}) > a\big)$, for $t \in [0,T]$, that we still denote ξ_{com} with a slight abuse of notation. Recall that $u_a(t,x) = \mathbb{P}\big(\Phi(X_T^{t,x}) > a\big) = \xi_{\text{com}}(t,x)$ satisfies the Kolmogorov backward equation (6), as explained in Section II.

As mentioned above, the asymptotic variance (as the number of replicas n_{rep} goes to infinity) of AMS algorithm is minimized when using the committor function as the score function[17]. The asymptotic variance is then $\frac{-p^2\log(p)}{n_{\text{rep}}}$, where p is the probability which is estimated. The analysis of the AMS algorithm in the ideal case, *i.e.* when using the committor function as a the score function, has been performed in many works[9,13,14,28].

Of course, in practice, the committor function is unknown and the asymptotic variance depends on the chosen score function. It has been proved[17] that the asymptotic variance is always bounded from above by $\frac{2p(1-p)}{n_{\text{rep}}}$, for any choice of the score function, where we recall that the asymptotic variance of the vanilla Monte-Carlo method is $\frac{p(1-p)}{n_{\text{rep}}}$. This can be seen as a sign of the robustness of the AMS approach to estimate rare event probability (contrary to importance sampling method which may result in a dramatic increase of the asymptotic variance compared with the vanilla Monte-Carlo method).

For simple Gaussian models, namely when X is a Brownian Motion, an Ornstein-Uhlenkeck process, or a drifted Brownian Motion, it is possible to compute analytically the committor function. This is useful to validate algorithms on test cases, as will be illustrated in Section V. Figure 3 represents the level lines of the committor function for a

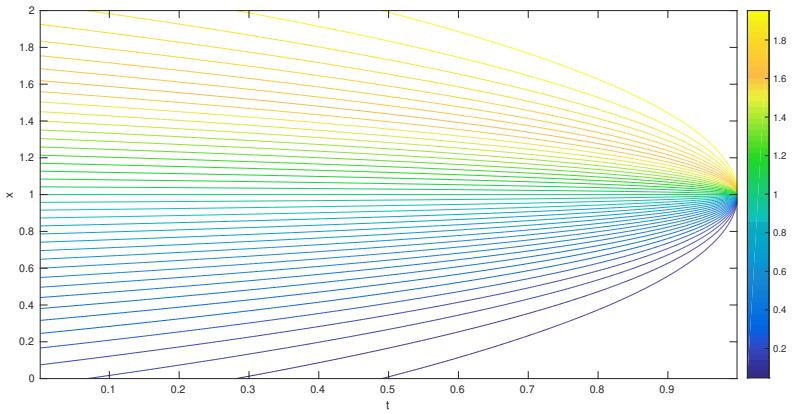

FIG. 3. Level lines of the committor function $(t,x) \mapsto \xi_{\mathrm{com}}(t,x)$, in the Brownian Motion case $X(t) = B(t)$, with $a = 1$ and $T = 1$.

one-dimensional Brownian Motion (with $T = 1$ and $a = 1$). In that case,

$$\xi_{\mathrm{com}}(t,x) = 1 - F\left(\frac{a-x}{\sqrt{T-t}}\right),$$

where F is the cumulative distribution function of the standard Gaussian distribution $\mathcal{N}(0,1)$, to be compared to the level sets of ξ_{new} and $\xi_{\mathrm{new}}^{\mathbf{a}}$ on Figures 1 and 2. This form leads to define other families of appropriate score functions:

$$\xi(t,x) = 1 - F\big(\phi(t,x)\big)$$

where $\phi(t,x) \underset{t\to\infty}{\to} (-\infty)\mathbb{1}_{\Phi(x)>a} + (+\infty)\mathbb{1}_{\Phi(x)<a}$. But the efficiency depends a lot on the choice of ϕ. In practice, we did not observe much gain in our numerical experiments, compared to the score function ξ_{new} introduced in the previous section.

Let us mention that various techniques have been proposed in order to approximate the committor function, in particular in the context of importance sampling techniques for rare events, since the committor function also gives the optimal change of measure. If diffusions with vanishing noise are considered[22,23,46], solutions of associated Hamilton-Jacobi equations are good candidates to estimate the committor function. See also[42] for approximations based on coarse-grained models. Whether such constructions are possible when considering temporal averages, instead of the terminal value of the process is unclear.

V. NUMERICAL SIMULATIONS

Let p denote the rare event probability of interest. An estimator of p is calculated as the empirical average over independent realizations of the AMS algorithm, given a choice of score function ξ. The main objective of this section is to investigate the behavior of the algorithm when choosing $\xi = \xi_{\text{new}}$ given by (13). A comparison with the vanilla score function $\xi = \xi_{\text{std}}$ given by (10) is provided.

Let $n_{\text{rep}} \in \mathbb{N}$ denote the number of replicas, and let $M \in \mathbb{N}$, and $\left(\hat{p}_m\right)_{1 \leq m \leq M}$ be the output probabilities of M independent realizations of the AMS algorithm. We report the values of the empirical average

$$\hat{p} = \frac{1}{M} \sum_{m=1}^{M} \hat{p}_m,$$

and of the empirical variance $\hat{\sigma}^2 = \frac{1}{M-1} \sum_{m=1}^{M} (\hat{p}_m - \hat{p})^2$. Confidence intervals are computed as

$$\left[\hat{p} - \frac{1.96\hat{\sigma}}{\sqrt{M}}, \hat{p} + \frac{1.96\hat{\sigma}}{\sqrt{M}} \right],$$

assuming that the number of realizations M is sufficiently large to use the Gaussian, Central Limit Theorem, regime.

Recall that $\mathbb{E}[\hat{p}] = \mathbb{E}[\hat{p}_1] = p$, whatever the choice of the score function ξ and of the number of replicas n_{rep}, thanks to Theorem 1. The variance of the estimator and thus the efficiency strongly depends on ξ. In the experiments below, the empirical variance $\hat{\sigma}^2$ is compared with the optimal asymptotic variance $\frac{-p^2 \log(p)}{n_{\text{rep}}}$ for (adaptive) multilevel splitting algorithms, which is obtained in the regime $n_{\text{rep}} \to \infty$, when choosing the (unknown in general) committor function $\xi = \xi_{\text{com}}$ as the score function. The difference between the empirical variance and the optimal one can be seen as a measure of how far the chosen score function is from the committor.

In some of the numerical experiments below, the conditional probability

$$q = \mathbb{P}\left(\Phi(X_N) > a \;\middle|\; \max_{0 \leq n \leq N} \Phi(X_n) > a \right), \tag{16}$$

is also estimated by

$$\hat{q} = \frac{\hat{p}}{\hat{p}_{\max}},$$

where $\hat{p}_{\max} = \frac{1}{M} \sum_{m=1}^{M} \hat{p}_{\max,\text{m}}$ is the estimator of the probability

$$p_{\max} = \mathbb{P}\left(\max_{0 \leq n \leq N} \Phi(X_n) > a \right), \tag{17}$$

which is estimated using the vanilla score function $\xi = \xi_{\text{std}}$.

A. Validation using two Gaussian models

In this section, we validate the AMS alogorithm with various score functions on simple models for which the probability of the rare event is known with arbitrary precision.

1. Brownian Motion

We follow here numerical experiments from[46]. Let $d = 1$, and consider the diffusion process given by

$$dX(t) = \sqrt{2\beta^{-1}} dW(t),\ X(0) = 0.1,$$

where $\big(W(t)\big)_{t\geq 0}$ is a standard real-valued Wiener process.

The dynamics is discretized using the explicit Euler-Maruyama method, with time-step size $\Delta t = 10^{-3}$ (notice that the numerical scheme gives here the exact solution):

$$X_{n+1} = X_n + \sqrt{2\beta^{-1}\Delta t}\zeta_n,$$

with $X_0 = X(0) = 0.1$, where $\big(\zeta_n\big)_{0\leq n\leq N}$ are independent standard Gaussian random variables.

The goal is to estimate the probability

$$p = \mathbb{P}\big(|X_N| > 1\big).$$

This corresponds with the choice $\Phi(x) = |x|$, $a = 1$, $T = 1$ so that $N = T/\Delta t = 10^3$.

Since $X(t) = X(0) + W(t)$, the law of $X(t)$ is a Gaussian distribution, and the value of $\mathbb{P}(|X(1)| \geq 1)$ can be computed exactly in terms of the cumulative distribution function of the standard Gaussian distribution.

Two numerical experiments are reported below, using $\xi = \xi_{\text{new}}$. First, in Table I, the number of replicas is set equal to $n_{\text{rep}} = 10^2$, and the empirical average is computed over $M = 10^4$ independent realizations of the algorithm. Second, in Table II, the number of replicas is set equal to $n_{\text{rep}} = 10^3$, and the empirical average is computed over $M = 10^3$ independent realizations of the algorithm.

β	$\hat{p}$	p ($\Delta t = 0$)	confidence interval	$\hat{\sigma}^2$	$\frac{-p^2 \log(p)}{n_{\text{rep}}}$
2	$3.199\ 10^{-1}$	$3.197\ 10^{-1}$	$[3.192\ 10^{-1}, 3.205\ 10^{-1}]$	$1.256\ 10^{-3}$	$1.166\ 10^{-3}$
4	$1.613\ 10^{-1}$	$1.614\ 10^{-1}$	$[1.608\ 10^{-1}, 1.617\ 10^{-1}]$	$5.275\ 10^{-4}$	$4.751\ 10^{-4}$
8	$4.978\ 10^{-2}$	$4.983\ 10^{-2}$	$[4.958\ 10^{-2}, 4.998\ 10^{-2}]$	$1.030\ 10^{-4}$	$7.447\ 10^{-5}$
16	$6.395\ 10^{-3}$	$6.386\ 10^{-3}$	$[6.353\ 10^{-3}, 6.436\ 10^{-3}]$	$4.449\ 10^{-6}$	$2.061\ 10^{-6}$
32	$1.631\ 10^{-4}$	$1.645\ 10^{-4}$	$[1.611\ 10^{-4}, 1.651\ 10^{-4}]$	$1.052\ 10^{-8}$	$2.358\ 10^{-9}$
64	$1.787\ 10^{-7}$	$1.782\ 10^{-7}$	$[1.721\ 10^{-7}, 1.854\ 10^{-7}]$	$1.139\ 10^{-13}$	$4.935\ 10^{-15}$
128	$2.501\ 10^{-13}$	$3.011\ 10^{-13}$	$[1.766\ 10^{-13}, 3.235\ 10^{-13}]$	$1.403\ 10^{-23}$	$2.614\ 10^{-26}$

TABLE I. Brownian Motion, $n_{\text{rep}} = 10^2$, $M = 10^4$.

β	$\hat{p}$	p ($\Delta t = 0$)	confidence interval	$\hat{\sigma}^2$	$\frac{-p^2 \log(p)}{n_{\text{rep}}}$
2	$3.196\ 10^{-1}$	$3.197\ 10^{-1}$	$[3.188\ 10^{-1}, 3.203\ 10^{-1}]$	$1.362\ 10^{-4}$	$1.166\ 10^{-4}$
4	$1.617\ 10^{-1}$	$1.614\ 10^{-1}$	$[1.612\ 10^{-1}, 1.621\ 10^{-1}]$	$5.456\ 10^{-5}$	$4.751\ 10^{-5}$
8	$4.983\ 10^{-2}$	$4.983\ 10^{-2}$	$[4.963\ 10^{-2}, 5.002\ 10^{-2}]$	$9.815\ 10^{-6}$	$7.447\ 10^{-6}$
16	$6.411\ 10^{-3}$	$6.386\ 10^{-3}$	$[6.371\ 10^{-3}, 6.451\ 10^{-3}]$	$4.242\ 10^{-7}$	$2.061\ 10^{-7}$
32	$1.634\ 10^{-4}$	$1.645\ 10^{-4}$	$[1.614\ 10^{-4}, 1.655\ 10^{-4}]$	$1.063\ 10^{-9}$	$2.358\ 10^{-10}$
64	$1.800\ 10^{-7}$	$1.782\ 10^{-7}$	$[1.740\ 10^{-7}, 1.860\ 10^{-7}]$	$9.360\ 10^{-15}$	$4.935\ 10^{-16}$
128	$3.045\ 10^{-13}$	$3.011\ 10^{-13}$	$[2.426\ 10^{-13}, 3.664\ 10^{-13}]$	$9.986\ 10^{-25}$	$2.614\ 10^{-27}$

TABLE II. Brownian Motion, $n_{\text{rep}} = 10^3$, $M = 10^3$.

These numerical experiments validate the algorithm using $\xi = \xi_{\text{new}}$ in the case of a one-dimensional Brownian Motion. The empirical variance $\hat{\sigma}^2$ is much smaller than $\frac{p(1-p)}{n_{\text{rep}}}$ which would be obtained using a naive Monte-Carlo strategy (using n_{rep} independent replicas). It is observed that the ratio between the empirical and the optimal variances increase when p decreases, but in practice this increase only has a limited impact and the new algorithm remains effective.

2. *Ornstein-Uhlenbeck*

We consider here an example taken from[49]. Let $d = 1$, and consider the diffusion process given by

$$dX(t) = -X(t)dt + dW(t),\ X(0) = 0.$$

The dynamics is discretized using the explicit Euler-Maruyama method, with time-step size $\Delta t = 10^{-3}$:

$$X_{n+1} = X_n - \Delta t X_n + \sqrt{\Delta t}\zeta_n,$$

with $X_0 = X(0) = 0$, where $\big(\zeta_n\big)_{0\le n\le N}$ are independent standard Gaussian random variables.

The goal is to estimate the probability

$$p = \mathbb{P}\big(X_N > a\big),$$

for different values of a. This corresponds with the choice $\Phi(x) = x$. The value of T is set to $T = 2$ so that $N = T/\Delta t = 2000$.

In this numerical experiment, the number of replicas is set equal to $n_{\rm rep} = 10^2$, and the empirical average is computed over $M = 10^4$ independent realizations of the algorithm. The estimator $\hat{p}_{\rm new}$ of p and the empirical variance $\hat{\sigma}^2_{\rm new}$ are obtained using the score function $\xi = \xi_{\rm new}$. The estimator $\hat{p}_{\rm std}$ and the empirical variance $\hat{\sigma}^2_{\rm std}$ are obtained using the vanilla splitting strategy, with reaction coordinate $\xi = \xi_{\rm std}$. The value of the probability p for the continuous time process, and the optimal variance $\frac{-p^2\log(p)}{n_{\rm rep}}$ are also reported for comparison.

a	$\hat{p}_{\rm new}$	$\hat{p}_{\rm std}$	p $(\Delta t = 0)$	$\hat{\sigma}^2_{\rm new}$	$\hat{\sigma}^2_{\rm std}$	$\frac{-p^2\log(p)}{n_{\rm rep}}$
2.8	$3.216\ 10^{-5}$	$3.252\ 10^{-5}$	$3.213\ 10^{-5}$	$2.377\ 10^{-10}$	$3.327\ 10^{-10}$	$1.068\ 10^{-10}$
2.9	$1.756\ 10^{-5}$	$1.728\ 10^{-5}$	$1.742\ 10^{-5}$	$7.917\ 10^{-10}$	$1.009\ 10^{-10}$	$3.325\ 10^{-11}$
3.0	$9.341\ 10^{-6}$	$9.300\ 10^{-6}$	$9.260\ 10^{-6}$	$2.411\ 10^{-11}$	$5.832\ 10^{-11}$	$9.937\ 10^{-12}$
3.1	$4.857\ 10^{-6}$	$4.826\ 10^{-6}$	$4.827\ 10^{-6}$	$7.011\ 10^{-12}$	$8.482\ 10^{-12}$	$2.852\ 10^{-12}$
3.2	$2.486\ 10^{-6}$	$2.449\ 10^{-6}$	$2.468\ 10^{-6}$	$1.984\ 10^{-12}$	$2.475\ 10^{-12}$	$7.864\ 10^{-13}$

TABLE III. Ornstein-Uhlenbeck, $T = 2$. Comparison of the new and of the vanilla splitting algorithms. $n_{\rm rep} = 10^2$, $M = 10^4$.

The results of numerical experiments with $T = 4$ and $T = 8$ are reported below. The quantity

$$\hat{r} = \frac{1}{M} \sum_{m=1}^{M} \mathbb{1}_{\hat{p}_m > 0}$$

is also reported, when the vanilla score function is used. This is the proportion of the independent realizations of the algorithm which contribute in the empirical average. This proportion depends on the conditional probability q (see (16)): it may happen that the n_{rep} replicas obtained at the final iteration all satisfy $X_N \leq 1$, even if by construction they all satisfy $\max_{0 \leq n \leq N} X_n > 1$. However, by construction (except if extinction of the system of replicas happens, which has not been observed in this experiment), if the new score function is used, $\hat{r}$ is identically equal to 1. Observe that, when T goes to infinity, by ergodicity of the process, $p_{\max} \to 1$ (see (17) for the definition of $p_{\max}$), whereas q and p converge to a non trivial probability. Thus, when T goes to infinity, it is expected that $\hat{r}$ will be equal to 0 if M is too small, when using the vanilla strategy with $\xi = \xi_{\text{std}}$. The conditional probability q is also estimated: $\hat{q} = 0.07$ when $T = 2$, $\hat{q} = 0.02$ when $T = 4$, $\hat{q} = 0.01$ when $T = 8$.

a	$\hat{p}_{\text{new}}$	$\hat{p}_{\text{std}}$	p ($\Delta t = 0$)	$\hat{\sigma}^2_{\text{new}}$	$\hat{\sigma}^2_{\text{std}}$	$\frac{-p^2 \log(p)}{n_{\text{rep}}}$	$\hat{r}$
2.8	$3.743\ 10^{-5}$	$3.789\ 10^{-5}$	$3.740\ 10^{-5}$	$7.106\ 10^{-10}$	$1.089\ 10^{-9}$	$1.426\ 10^{-10}$	0.85
2.9	$2.052\ 10^{-5}$	$2.069\ 10^{-5}$	$2.049\ 10^{-5}$	$2.438\ 10^{-10}$	$3.484\ 10^{-10}$	$4.532\ 10^{-11}$	0.83
3.0	$1.104\ 10^{-5}$	$1.124\ 10^{-5}$	$1.101\ 10^{-5}$	$7.540\ 10^{-11}$	$1.147\ 10^{-10}$	$1.384\ 10^{-11}$	0.81
3.1	$5.822\ 10^{-6}$	$5.856\ 10^{-6}$	$5.805\ 10^{-6}$	$2.412\ 10^{-11}$	$3.584\ 10^{-11}$	$4.062\ 10^{-12}$	0.78
3.2	$3.022\ 10^{-6}$	$3.016\ 10^{-6}$	$3.002\ 10^{-6}$	$7.452\ 10^{-12}$	$9.964\ 10^{-12}$	$1.146\ 10^{-12}$	0.75

TABLE IV. Ornstein-Uhlenbeck, $T = 4$. Comparison of the new and of the vanilla splitting algorithms. $n_{\text{rep}} = 10^2$, $M = 10^4$.

To conclude this section, note that on this example, the AMS algorithms applied with the vanilla and the new score functions have a similar quantitative behavior in terms of asymptotic variance. However, their qualitative properties are different. When the conditional probability q gets small, the advantage of the new score function is the fact that the output $\hat{p}$ is always positive, so that even with a few realizations, one gets a rough but informative approximation of the target probability.

a	$\hat{p}_{\text{new}}$	$\hat{p}_{\text{std}}$	p $(\Delta t = 0)$	$\hat{\sigma}^2_{\text{new}}$	$\hat{\sigma}^2_{\text{std}}$	$\frac{-p^2 \log(p)}{n_{\text{rep}}}$	$\hat{r}$
2.8	$3.792\ 10^{-5}$	$3.714\ 10^{-5}$	$3.751\ 10^{-5}$	$1.696\ 10^{-9}$	$2.435\ 10^{-9}$	$1.434\ 10^{-10}$	0.54
2.9	$2.036\ 10^{-5}$	$2.071\ 10^{-5}$	$2.055\ 10^{-5}$	$5.439\ 10^{-10}$	$8.289\ 10^{-10}$	$4.557\ 10^{-11}$	0.51
3.0	$1.103\ 10^{-5}$	$1.128\ 10^{-5}$	$1.104\ 10^{-5}$	$1.843\ 10^{-10}$	$2.745\ 10^{-10}$	$1.392\ 10^{-11}$	0.49
3.1	$5.915\ 10^{-6}$	$5.968\ 10^{-6}$	$5.824\ 10^{-6}$	$5.599\ 10^{-11}$	$8.457\ 10^{-11}$	$4.089\ 10^{-12}$	0.47
3.2	$2.978\ 10^{-6}$	$3.022\ 10^{-6}$	$3.013\ 10^{-6}$	$1.639\ 10^{-11}$	$2.368\ 10^{-11}$	$1.154\ 10^{-12}$	0.43

TABLE V. Ornstein-Uhlenbeck, $T = 8$. Comparison of the new and of the vanilla splitting algorithms. $n_{\text{rep}} = 10^2$, $M = 10^4$.

B. Drifted Brownian Motion

We here considers numerical examples taken from [10,40]. Let $d = 1$, and consider the diffusion process given by

$$dX_t = -\alpha dt + \sqrt{2\beta^{-1}} dW(t),\ X(0) = 0.$$

The dynamics is discretized using the explicit Euler-Maruyama method, with time-step size $\Delta t = 10^{-2}$ (which gives again the exact solution in this simple case):

$$X_{n+1} = X_n - \alpha \Delta t + \sqrt{2\beta^{-1}\Delta t}\zeta_n,$$

with $X_0 = X(0) = 0$, where $\big(\zeta_n\big)_{0 \le n \le N}$ are independent standard Gaussian random variables.

The goal is to estimate the probability

$$p = \mathbb{P}\big(X_N > 1\big),$$

thus $\Phi(x) = x$, $a = 1$. One considers the final time $T = 1$, so that $N = T/\Delta t = 10^2$. The value of α is set equal to $\alpha = 4$. As in the previous example, the value of p is easy to get using the fact that X_N is Gaussian.

In this numerical experiment, see Table VI, three choices of score functions are considered. The number of replicas is $n_{\text{rep}} = 10^3$. First, the estimator $\hat{p}_{\text{new}}$ and the empirical variance $\hat{\sigma}^2_{\text{new}}$ are obtained using $\xi = \xi_{\text{new}}$, with a sample size $M = 4.10^4$. Second, the estimator $\hat{p}_{\text{new},\mathbf{a}}$ and the empirical variance $\hat{\sigma}^2_{\text{new},\mathbf{a}}$ are obtained using $\xi = \xi^{\mathbf{a}}_{\text{new}}$ with $\mathbf{a}(t) = \frac{at}{T}$, with a sample size $M = 4.10^5$. Finally, the estimator $\hat{p}_{\text{std}}$ and the empirical variance $\hat{\sigma}^2_{\text{std}}$ are

obtained using $\xi = \xi_{\text{std}}$, with a sample size $M = 4.10^4$. The sample sizes are chosen such that the total computational cost is of the same order for the three methods.

β	$\hat{p}_{\text{new}}$	$\hat{p}_{\text{new},\mathbf{a}}$	$\hat{p}_{\text{std}}$	p	$\hat{\sigma}^2_{\text{new}}$	$\hat{\sigma}^2_{\text{new},\mathbf{a}}$	$\hat{\sigma}^2_{\text{std}}$	$\frac{-p^2\log(p)}{n_{\text{rep}}}$	$\hat{r}$
1	$2.037\ 10^{-4}$	$2.036\ 10^{-4}$	$2.033\ 10^{-4}$	$2.035\ 10^{-4}$	$1.572\ 10^{-9}$	$5.734\ 10^{-9}$	$3.525\ 10^{-9}$	$3.519\ 10^{-10}$	0.99
2	$2.843\ 10^{-7}$	$2.870\ 10^{-7}$	$2.878\ 10^{-7}$	$2.867\ 10^{-7}$	$4.714\ 10^{-14}$	$2.348\ 10^{-12}$	$9.527\ 10^{-14}$	$1.238\ 10^{-15}$	0.69
3	$4.613\ 10^{-10}$	$4.325\ 10^{-10}$	$4.705\ 10^{-10}$	$4.571\ 10^{-10}$	$1.817\ 10^{-18}$	$2.197\ 10^{-17}$	$3.084\ 10^{-18}$	$4.493\ 10^{-21}$	0.11
4	$7.620\ 10^{-13}$	$6.975\ 10^{-13}$	$7.582\ 10^{-13}$	$7.687\ 10^{-13}$	$5.034\ 10^{-23}$	$4.388\ 10^{-22}$	$8.343\ 10^{-23}$	$1.648\ 10^{-26}$	0.01

TABLE VI. Drifted Brownian Motion, $\beta \in \{1, 2, 3, 4\}$, Comparison of two versions of the new splitting algorithm and of the vanilla splitting algorithm.

Since the sample size is not the same for the three examples of score functions in Table VI, the values of the empirical variances $\hat{\sigma}^2$ should be taken with care when comparing the methods. One would rather compare the values of $\frac{\hat{\sigma}^2}{M}$. Then one concludes that the best performance is obtained when using $\xi = \xi^{\mathbf{a}}_{\text{new}}$. The vanilla strategy, with $\xi = \xi_{\text{std}}$, seems to behave quantitatively the same as when $\xi = \xi_{\text{new}}$. However, the values of the proportion $\hat{r}$ of realizations such that $\hat{p}_m \neq 0$ is not zero is also reported, when $\xi = \xi_{\text{std}}$ (by construction, $\hat{r} = 1$ for the first two cases). This means that if M was decreased (for instance, M of the order 10^2 for $\beta = 4$), then the output of the experiment would be $\hat{p}_{\text{std}} = 0$.

As a consequence, the new algorithm clearly overcomes the limitation of the vanilla strategy when $\xi = \xi_{\text{std}}$. However, the score functions are far from being optimal, as revealed by the comparison with the optimal variance.

C. Temporal averages for an Ornstein-Uhlenbeck process

In this section, we consider an example taken from[24]. Consider the one-dimensional Ornstein-Uhlenbeck process X, which is the solution of the SDE

$$dX(t) = -X(t)dt + \sqrt{2\beta^{-1}}dW(t),\ X(0) = 0,$$

and define the temporal average

$$Y(T) = \frac{1}{T}\int_0^T X(s)ds.$$

More generally, set $Y(t) = \frac{1}{t}\int_0^t X(s)ds$, for $t \in (0, T]$ and $Y(0) = 0$.

The discretization is performed using the explicit Euler-Maruyama method, with time-step size $\Delta t = 5.\ 10^{-3}$: for $n \in \{0, \dots, N\}$ with $N\Delta t = T$,

$$X_{n+1} = (1 - \Delta t)X_n + \sqrt{2\beta^{-1}\Delta t}\zeta_n, \quad Y_n = \frac{1}{n}\sum_{m=1}^{n} X_m,$$

with $X_0 = Y_0 = 0$. Note that Y satisfies a recursion formula $Y_{n+1} = (1 - \frac{1}{n+1})Y_n + \frac{1}{n+1}X_{n+1}$.

The number of replicas is set equal to $n_{\text{rep}} = 10^3$ and the sample size to compute empirical averages is $M = 10^2$.

In this section, the probability which is estimated is

$$p(T, a) = \mathbb{P}(Y_N > a).$$

The associated estimator is denoted by $\hat{p}(T, a)$ and the empirical variance by $\hat{\sigma}^2(T, a)$.

In the large time limit $T \to \infty$, since the law of $Y(T)$ converges to a centered Gaussian with variance 1, $Y(T)$ satisfies a large deviation principle, with rate function I defined by: for all $a > 0$,

$$\lim_{T\to\infty} \ -\frac{1}{T}\log\Big(\mathbb{P}\big(Y(T) > a\big)\Big) = I(a) = \frac{a^2}{4}.$$

In the numerical experiment, we illustrate the potential of the AMS algorithm to estimate the large deviations rate function. Notice that in the large T limit, the probability is extremely small and in practice cannot be estimated by the vanilla splitting strategy. The estimate of the rate function $\hat{I}(a)$ is obtained by a regression procedure, see Figure 4. In addition to statistical error, two sources of numerical error are identified: values of T may not be sufficiently large, and the discretization of the dynamics and of the computation of temporal averages introduces a bias. The results, reported in Table VII, show the interest of this approach to estimate large deviations rate functionals.

D. Lorenz model

We consider the following stochastic version of the 3-dimensional Lorenz system, see[5,32] for similar numerical experiments:

$$\begin{cases} dX_1^\beta(t) = \sigma\big(X_2(t) - X_1(t)\big)dt + \sqrt{2\beta^{-1}}dW(t), \\ dX_2^\beta(t) = \big(rX_1(t) - X_2(t) - X_1(t)X_3(t)\big)dt, \\ dX_3^\beta(t) = X_1(t)X_2(t) - bX_3(t), \end{cases}$$

a	$\hat{p}(T=25,a)$	$\hat{p}(T=50,a)$	$\hat{p}(T=100,a)$	$\hat{p}(T=200,a)$	$\hat{I}(a)$	$\frac{a^2}{4}$
	$\hat{\sigma}^2(T=25,a)$	$\hat{\sigma}^2(T=50,a)$	$\hat{\sigma}^2(T=100,a)$	$\hat{\sigma}^2(T=200,a)$		
0.4	$7.28\ 10^{-2}$	$2.12\ 10^{-2}$	$2.22\ 10^{-3}$	$2.75\ 10^{-5}$	0.045	0.040
–	$2.02\ 10^{-5}$	$3.25\ 10^{-6}$	$1.27\ 10^{-7}$	$3.85\ 10^{-11}$	–	–
0.6	$1.45\ 10^{-2}$	$1.16\ 10^{-3}$	$1.16\ 10^{-5}$	$7.28\ 10^{-10}$	0.096	0.090
–	$1.36\ 10^{-6}$	$5.85\ 10^{-8}$	$2.21\ 10^{-10}$	$6.44\ 10^{-19}$	–	–
0.8	$1.76\ 10^{-3}$	$2.57\ 10^{-5}$	$6.12\ 10^{-9}$	$2.73\ 10^{-16}$	0.169	0.160
–	$8.01\ 10^{-8}$	$3.32\ 10^{-10}$	$1.44\ 10^{-16}$	$1.98\ 10^{-31}$	–	–
1.0	$1.37\ 10^{-4}$	$1.67\ 10^{-7}$	$3.06\ 10^{-13}$	$1.71\ 10^{-24}$	0.261	0.250
–	$1.47\ 10^{-9}$	$2.07\ 10^{-14}$	$1.16\ 10^{-25}$	$9.32\ 10^{-48}$	–	–
1.2	$6.21\ 10^{-6}$	$4.83\ 10^{-10}$	$2.71\ 10^{-18}$	$2.88\ 10^{-34}$	0.373	0.360
–	$1.40\ 10^{-11}$	$6.69\ 10^{-19}$	$3.49\ 10^{-35}$	$5.21\ 10^{-67}$	–	–

TABLE VII. Temporal averages for an Ornstein-Uhlenbeck process. $n_{\text{rep}} = 10^3$ and $M = 10^2$

which depends on parameters σ, r, b and β. The parameters are given the following values in this section: $\sigma = 3$, $r = 26$ and $b = 1$.

Consider first the deterministic case, *i.e.* $\beta = \infty$. Then the system admits three unstable equilibria, and one of them is

$$x^\star = \big(\sqrt{b(r-1)}, \sqrt{b(r-1)}, r-1\big) = (5, 5, 25).$$

Let the initial condition be given by $X^\infty(0) = x^\star + \frac{1}{2}(1,1,1)$. Then, one has the following stability result[5]: for all $t \geq 0$, $\Phi\big(X^\infty(t)\big) \leq 1$, where

$$\Phi(x) = \frac{x_1^2}{(r+\sigma)^2 \frac{b}{\sigma}} + \frac{x_2^2}{(r+\sigma)^2 b} + \frac{\big(x_3 - (r+\sigma)\big)^2}{(r+\sigma)^2}.$$

When noise is introduced in the system, *i.e.* $\beta < \infty$, we are interested in the estimation of the probability

$$\mathbb{P}\big(\Phi(X^\beta(T)) > 1\big),$$

with threshold $a = 1$.

In the numerical experiments, $\sqrt{2\beta^{-1}} = 3$, and the discretization is performed using the explicit Euler-Maruyama method, with time-step size $\Delta t = 10^{-2}$. The sample size is $M = 10^4$, and the number of replicas is $n_{\text{rep}} = 10^3$.

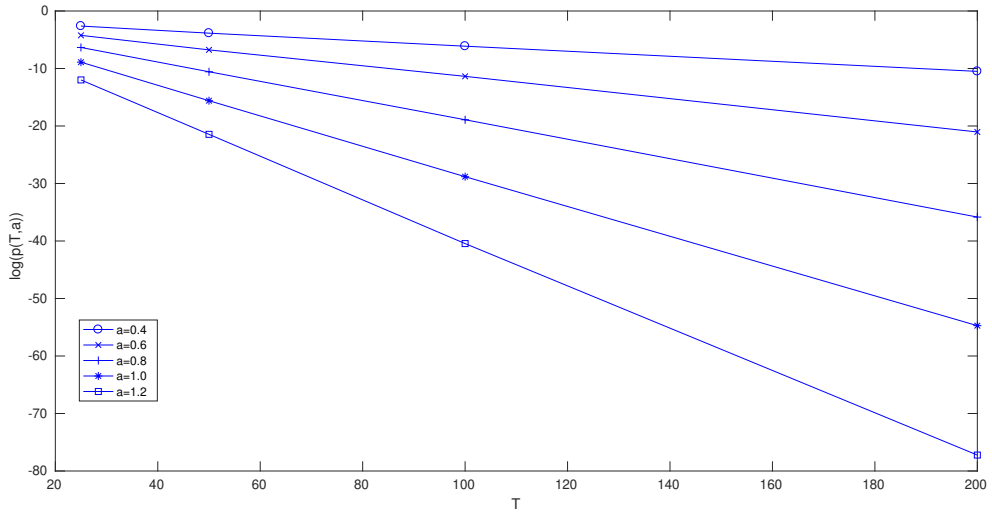

FIG. 4. Evolution of $\log\big(\hat{p}(T,a)\big)$ as a function of T, for different values of $a \in \{0.4, 0.6, 0.8, 1.0, 1.2\}$, see Table VII.

T	$\hat{p}$	confidence interval	$\hat{\sigma}^2$	$\frac{-\hat{p}^2 \log(\hat{p})}{n_{\rm rep}}$
5	$1.413\ 10^{-5}$	$[1.388\ 10^{-5}, 1.438\ 10^{-5}]$	$1.648\ 10^{-10}$	$2.230\ 10^{-12}$
10	$2.607\ 10^{-5}$	$[2.534\ 10^{-5}, 2.681\ 10^{-5}]$	$1.409\ 10^{-9}$	$7.174\ 10^{-12}$
15	$2.709\ 10^{-5}$	$[2.609\ 10^{-5}, 2.809\ 10^{-5}]$	$2.592\ 10^{-9}$	$7.718\ 10^{-12}$
20	$2.594\ 10^{-5}$	$[2.484\ 10^{-5}, 2.704\ 10^{-5}]$	$3.158\ 10^{-9}$	$7.106\ 10^{-12}$

TABLE VIII. Lorenz model. $n_{\rm rep} = 10^3$ and $M = 10^4$

This numerical experiment thus illustrates the potential of the adaptive multilevel splitting algorithms introduced in this article, for applications to complex, nonlinear, stochastic models.

E. Driven periodic diffusion

We finally consider an example taken from[24]. In this section, we consider the SDE on the unit circle, *i.e.* on the torus $\mathbb{T}$,

$$dX(t) = \big(-V'(X(t)) + \gamma\big)dt + \sqrt{2}dW(t),$$

where the potential energy function $V(x) = \cos(2\pi x)$ is periodic, and $\gamma \in \mathbb{R}$. If $\gamma \neq 0$, this is called a non-equilibrium process since the drift term $-V'(x) + \gamma$ is not the derivative of a function defined on the torus $\mathbb{T}$. In the remainder of this section, let $\gamma = 1$, and let the initial condition in the simulation be $X_0 = 0$. The discretization is performed using the Euler-Maruyama method, with time-step size $\Delta t = 10^{-2}$.

We are interested in the behavior of $\frac{X_T}{T}$, when $T \to \infty$, more precisely we apply the AMS algorithm to estimate

$$p(T, a) = \mathbb{P}\big(X(T) > aT\big) = \mathbb{P}\left(\frac{X(T)}{T} > a\right).$$

Following the same approach as for the temporal averages of the Ornstein-Uhlenbeck process, a large deviations rate function

$$I(a) = \lim_{T\to\infty} \; -\frac{1}{T}\log\big(p(T, a)\big),$$

is estimated, based on estimators of the probability $p(T, a)$ for several values of T.

In this numerical experiment, we compare two ways of applying the AMS algorithm, with the new score function ξ_{new} but with different processes: considering either the process $\big(X(t)\big)_{0\leq t\leq T}$ with the threshold aT, or the process $\big(Y(t) = \frac{X(t)}{t}\big)_{0<t\leq T}$, with the threshold a. Numerical values for different choices of a, T and n_{rep}, of the associated estimators $\hat{p}^X(T, a)$ and $\hat{p}^Y(T, a)$, and of the empirical variances $\hat{\sigma}^{2,X}(T, a)$ and $\hat{\sigma}^{2,Y}(T, a)$ are reported in Table IX below.

It is observed that $\hat{\sigma}^{2,Y}(T, a) < \hat{\sigma}^{2,X}(T, a)$, but a fair comparison requires to take into account the (average) computational cost. Thus the relative efficiency $\text{Eff}(Y|X)$ of using the process Y instead of X, is computed as the ratio

$$\text{Eff}(Y|X) = \frac{\hat{\sigma}^{2,X}\text{comp.time}(X)}{\hat{\sigma}^{2,Y}\text{comp.time}(Y)},$$

where $\frac{\text{comp.time}(X)}{\text{comp.time}(Y)}$ is the ratio of the total computational times for the experiments using X and Y respectively. The values of $\text{Eff}(Y|X)$ in this numerical experiment are reported in Table IX. We observe that $\text{Eff}(Y|X) > 1$ which means that the algorithm is more efficient using the process Y than the process X. To have a comparison with the committor score function, since the value of $p(T, a)$ is not known, an approximation of the optimal variance is computed using the estimator $\hat{p}^Y(T, a)$.

Estimators $\hat{I}(a)$ of the large deviations rate function $I(a)$ are estimated by a regression procedure (with respect to T) using the estimators $\hat{p}^Y(T,a)$, for several values of a. The numerical values are in excellent agreement with the numerical experiments in[24]. The AMS algorithm introduced in this article can thus be an efficient tool to estimate large deviations rate functions.

a	T	$n_{\rm rep}$	$\hat{p}^X(T,a)$	$\hat{p}^Y(T,a)$	$\hat{\sigma}^{2,X}(T,a)$	$\hat{\sigma}^{2,Y}(T,a)$	$\frac{-(\hat{p}^Y(T,a))^2\log(\hat{p}^Y(T,a))}{n_{\rm rep}}$	Eff$(Y\|X)$	$\hat{I}(a)$
0.8	100	10^2	$8.483\ 10^{-2}$	$8.489\ 10^{-2}$	$7.136\ 10^{-4}$	$2.487\ 10^{-4}$	$1.777\ 10^{-4}$	1.0	
–	200	–	$2.647\ 10^{-2}$	$2.776\ 10^{-2}$	$1.832\ 10^{-4}$	$3.519\ 10^{-5}$	$2.762\ 10^{-5}$	1.7	0.0112
1	50	10^3	$1.529\ 10^{-2}$	$1.505\ 10^{-2}$	$7.046\ 10^{-6}$	$1.513\ 10^{-6}$	$9.504\ 10^{-7}$	3.0	
–	100	–	$1.026\ 10^{-3}$	$1.085\ 10^{-3}$	$2.586\ 10^{-7}$	$3.879\ 10^{-8}$	$8.036\ 10^{-9}$	2.8	0.0526
1.25	50	10^3	$1.374\ 10^{-4}$	$1.311\ 10^{-4}$	$1.227\ 10^{-8}$	$1.355\ 10^{-9}$	$1.537\ 10^{-10}$	4.9	
–	100	–	$8.941\ 10^{-7}$	$1.017\ 10^{-7}$	$1.048\ 10^{-13}$	$1.585\ 10^{-15}$	$1.665\ 10^{-16}$	35	0.189

TABLE IX. Estimates of $\mathbb{P}(X_T > aT)$ and of $I(a)$ for the Periodic driven diffusion. The sample size is $M = 100$.

ACKNOWLEDGMENTS

The authors would like to thank G. Ferré and H. Touchette for discussions concerning the numerical experiments, and F. Cérou, A. Guyader and M. Rousset for stimulating discussions at early stages of this work. The work of T. Lelièvre is supported by the European Research Council under the European Union's Seventh Framework Programme (FP/2007-2013) / ERC Grant Agreement number 614492.

REFERENCES

[1]A. Agarwal, S. De Marco, E. Gobet, and G. Liu. Study of new rare event simulation schemes and their application to extreme scenario generation. *Math. Comput. Simulation*, 143:89–98, 2018.

[2]D. Aristoff, T. Lelièvre, C. G. Mayne, and I. Teo. Adaptive multilevel splitting in molecular dynamics simulations. In *CEMRACS 2013—modelling and simulation of complex systems:*

stochastic and deterministic approaches, volume 48 of *ESAIM Proc. Surveys*, pages 215–225. EDP Sci., Les Ulis, 2015.

[3]S. Asmussen and P. Glynn. *Stochastic simulation: algorithms and analysis*, volume 57 of *Stochastic Modelling and Applied Probability*. Springer, New York, 2007.

[4]S. K. Au and J. L. Beck. Estimation of small failure probabilities in high dimensions by subset simulation. *Journal of Probabilistic Engineering Mechanics*, 16:263–277, 2001.

[5]J. L. Beck and K. M. Zuev. Rare-event simulation. *Handbook of uncertainty quantification*, pages 1075–1100, 2017.

[6]Z. I. Botev and D. P. Kroese. Efficient Monte Carlo simulation via the generalized splitting method. *Stat. Comput.*, 22(1):1–16, 2012.

[7]Z. I. Botev, P. L'Ecuyer, G. Rubino, R. Simard, and B. Tuffin. Static network reliability estimation via generalized splitting. *INFORMS J. Comput.*, 25(1):56–71, 2013.

[8]F. Bouchet, J. Rolland, and E. Simonnet. A rare event algorithm links transitions in turbulent flows with activated nucleations. *ArXiv preprint arXiv:1810.11057*, 2018.

[9]C.-E. Bréhier. Large deviations principle for the adaptive multilevel splitting algorithm in an idealized setting. *ALEA Lat. Am. J. Probab. Math. Stat.*, 12(2):717–742, 2015.

[10]C.-E. Bréhier, M. Gazeau, L. Goudenège, T. Lelièvre, and M. Rousset. Unbiasedness of some generalized adaptive multilevel splitting algorithms. *arXiv preprint arXiv:1505.02674*, 2015.

[11]C.-E. Bréhier, M. Gazeau, L. Goudenège, T. Lelièvre, and M. Rousset. Unbiasedness of some generalized adaptive multilevel splitting algorithms. *Ann. Appl. Probab.*, 26(6):3559–3601, 2016.

[12]C.-E. Bréhier, M. Gazeau, L. Goudenège, and M. Rousset. Analysis and simulation of rare events for SPDEs. In *CEMRACS 2013—modelling and simulation of complex systems: stochastic and deterministic approaches*, volume 48 of *ESAIM Proc. Surveys*, pages 364–384. EDP Sci., Les Ulis, 2015.

[13]C.-E. Bréhier, L. Goudenège, and L. Tudela. Central limit theorem for adaptive multilevel splitting estimators in an idealized setting. In *Monte Carlo and quasi-Monte Carlo methods*, volume 163 of *Springer Proc. Math. Stat.*, pages 245–260. Springer, [Cham], 2016.

[14]C.-E. Bréhier, T. Lelièvre, and M. Rousset. Analysis of adaptive multilevel splitting algorithms in an idealized case. *ESAIM Probab. Stat.*, 19:361–394, 2015.

[15]J. Bucklew. *Introduction to rare event simulation*. Springer Series in Statistics. Springer-

Verlag, New York, 2004.

[16]F. Cérou, P. Del Moral, T. Furon, and A. Guyader. Sequential Monte Carlo for rare event estimation. *Stat. Comput.*, 22(3):795–808, 2012.

[17]F. Cérou, B. Delyon, A. Guyader, and M. Rousset. On the asymptotic normality of adaptive multilevel splitting. *arXiv preprint arXiv:1804.08494*, 2018.

[18]F. Cérou and A. Guyader. Adaptive multilevel splitting for rare event analysis. *Stoch. Anal. Appl.*, 25(2):417–443, 2007.

[19]F. Cérou, A. Guyader, T. Lelièvre, and D. Pommier. A multiple replica approach to simulate reactive trajectories. *Journal of Chemical Physics*, 134(5), 2011.

[20]T. Dean and P. Dupuis. Splitting for rare event simulation: a large deviation approach to design and analysis. *Stochastic Process. Appl.*, 119(2):562–587, 2009.

[21]P. Del Moral and J. Garnier. Genealogical particle analysis of rare events. *Ann. Appl. Probab.*, 15(4):2496–2534, 2005.

[22]P. Dupuis, K. Spiliopoulos, and H. Wang. Importance sampling for multiscale diffusions. *SIAM J. Multiscale Model. and Simul.*, 10:1–27, 2012.

[23]P. Dupuis, K. Spiliopoulos, and X. Zhou. Escaping from an attractor: Importance sampling and rest points i. *Ann. Appl. Probab.*, 25(5):2909–2958, 2015.

[24]G. Ferré and H. Touchette. Adaptive sampling of large deviations. *arXiv preprint arXiv:1803.11117*, 2018.

[25]M. Garvels, D. Kroese, and J. van Ommeren. On the importance function in splitting simulation. *European Transactions on Telecom- munications*, 13(4):363–371, 2002.

[26]P. Glasserman, P. Heidelberger, P. Shahabuddin, and T. Zajic. Multilevel splitting for estimating rare event probabilities. *Oper. Res.*, 47(4):585–600, 1999.

[27]E. Gobet and G. Liu. Rare event simulation using reversible shaking transformations. *SIAM J. Sci. Comput.*, 37(5):A2295–A2316, 2015.

[28]A. Guyader, N. Hengartner, and E. Matzner-Lø ber. Simulation and estimation of extreme quantiles and extreme probabilities. *Appl. Math. Optim.*, 64(2):171–196, 2011.

[29]H. Kahn and T. E. Harris. Estimation of particle transmission by random sampling. *National Bureau of Standards*, 12:27–30, 1951.

[30]T. Lestang, F. Ragone, C.-E. Bréhier, C. Herbert, and F. Bouchet. Computing return times or return periods with rare event algorithms. *J. Stat. Mech. Theory Exp.*, (4):043213, 33, 2018.

[31]L. J. Lopes and T. Lelièvre. Analysis of the adaptive multilevel splitting method with the alanine di-peptide's isomerization. *arXiv preprint arXiv:1707.00950*, 2017.

[32]E. N. Lorenz. Deterministic nonperiodic flow. *Journal of the atmospheric sciences*, 20(2):130–141, 1963.

[33]E. Louvin, E. Dumonteil, T. Lelièvre, M. Rousset, and C. Diop. Adaptive multilevel splitting for Monte Carlo particle transport. *EPJ Nuclear Sci. Technol.*, 3(29), 2017.

[34]H. Louvin. *Development of an adaptive variance reduction technique for Monte Carlo particle transport*. PhD thesis, Université Paris-Saclay, 2017.

[35]H. Louvin, E. Dumonteil, and T. Lelievre. Three-dimensional neutron streaming calculations using adaptive multilevel splitting. In *International Conference on Mathematics and Computational Methods Applied to Nuclear Science and Engineering (M&C 2017), Jeju, Korea*, 2017.

[36]R. Poncet. *Méthodes numériques pour la simulation d'équations aux dérivées partielles stochastiques non-linéaires en condensation de Bose-Einstein*. PhD thesis, Université Paris-Saclay, 2017.

[37]F. Ragone, J. Wouters, and F. Bouchet. Computation of extreme heat waves in climate models using a large deviation algorithm. *Proc. Natl. Acad. Sci. USA*, 115(1):24–29, 2018.

[38]J. Rolland. Extremely rare collapse and build-up of turbulence in stochastic models of transitional wall flows. *Physical Review E*, 97(2):023109, 2018.

[39]J. Rolland, F. Bouchet, and E. Simonnet. Computing transition rates for the 1-D stochastic Ginzburg-Landau-Allen-Cahn equation for finite-amplitude noise with a rare event algorithm. *J. Stat. Phys.*, 162(2):277–311, 2016.

[40]J. Rolland and E. Simonnet. Statistical behaviour of adaptive multilevel splitting algorithms in simple models. *Journal of Computational Physics*, 283:541 – 558, 2015.

[41]G. Rubino and B. Tuffin. Introduction to rare event simulation. In *Rare event simulation using Monte Carlo methods*, pages 1–13. Wiley, Chichester, 2009.

[42]C. Schütte, S. Winkelmann, and C. Hartmann. Optimal control of molecular dynamics using markov state models. *Mathematical Programming*, 134(1):259–282, 2012.

[43]J. Skilling. Nested sampling for general Bayesian computation. *Bayesian Anal.*, 1(4):833–859 (electronic), 2006.

[44]J. Skilling. Nested sampling for Bayesian computations. In *Bayesian statistics 8*, Oxford Sci. Publ., pages 491–524. Oxford Univ. Press, Oxford, 2007.

[45]I. Teo, C. Mayne, K. Schulten, and T. Lelièvre. Adaptive multilevel splitting method for molecular dynamics calculation of benzamidine-trypsin dissociation time. *Journal of chemical theory and computation*, 12(6):2983–2989, 2016.

[46]E. Vanden-Eijnden and J. Weare. Rare event simulation of small noise diffusions. *Comm. Pure Appl. Math.*, 65(12):1770–1803, 2012.

[47]M. Villén-Altamirano and J. Villén-Altamirano. RESTART: A method for accelerating rare events simulations. In *Proceeding of the thirteenth International Teletraffic Congress*, volume Copenhagen, Denmark, June 19-26 of *Queueing, performance and control in ATM: ITC-13 workshops*, pages 71–76. North-Holland, Amsterdam-New York, 1991.

[48]M. Villén-Altamirano and J. Villén-Altamirano. RESTART: a straightforward method for fast simulation of rare events. In *Proceedings of the 1994 Winter Simulation Conference*, volume Orlando 1994, December 1994, pages 282–289. 1994.

[49]J. Wouters and F. Bouchet. Rare event computation in deterministic chaotic systems using genealogical particle analysis. *J. Phys. A*, 49(37):374002, 24, 2016.