Rare event simulation for steady-state probabilities via recurrency cycles

Cite as: Chaos **29**, 033131 (2019); doi: 10.1063/1.5080296
Submitted: 7 November 2018 · Accepted: 5 March 2019 ·
Published Online: 28 March 2019

Krzysztof Bisewski,[1,a)] Daan Crommelin,[1,2] and Michel Mandjes[2]

AFFILIATIONS

[1]Centrum Wiskunde and Informatica, Science Park 123, 1098 XG Amsterdam, The Netherlands
[2]Korteweg-de Vries Institute for Mathematics, University of Amsterdam, Science Park 105, 1098 XG Amsterdam, The Netherlands

Note: This article is part of the Focus Issue on "Rare Event Sampling Methods: Development, Analysis and Application."
[a)]**Electronic mail:** K.L.Bisewski@cwi.nl

ABSTRACT

We develop a new algorithm for the estimation of rare event probabilities associated with the steady-state of a Markov stochastic process with continuous state space $\mathbb{R}^d$ and discrete time steps (i.e., a discrete-time $\mathbb{R}^d$-valued Markov chain). The algorithm, which we coin Recurrent Multilevel Splitting (RMS), relies on the Markov chain's underlying recurrent structure, in combination with the Multilevel Splitting method. Extensive simulation experiments are performed, including experiments with a nonlinear stochastic model that has some characteristics of complex climate models. The numerical experiments show that RMS can boost the computational efficiency by several orders of magnitude compared to the Monte Carlo method.

We develop a new algorithm for the estimation of small probabilities associated with steady-state distribution of a Markov stochastic process. Such steady-state probabilities are relevant in application domains ranging from operations research to climate science. The algorithm is simple to use in practice and does not require detailed knowledge of the stochastic process so that it can be applied to a broad class of systems (including "black box" models that can be simulated numerically but that are too complex to be studied analytically). Extensive simulation experiments are performed, including experiments with a nonlinear stochastic model that has some characteristics of complex climate models. The numerical studies show that our method can give major computational efficiency gains (up to several orders of magnitude in our examples) compared to the Monte Carlo method.

I. INTRODUCTION

Many stochastic processes have a "stable regime," in the sense that with time their distribution converges to a so-called *steady-state*. The steady-state (or stationary, equilibrium, ergodic) probability distribution captures the long-term behavior of the process; the steady-state probability of an arbitrary event (or set) B is equal to the fraction of time the process spends in B in the long run (irrespective of the process' initial value). In many application domains, steady-state probabilities are of crucial interest, i.e., physics (e.g., particle systems), chemistry (e.g., reaction networks), and operations research (e.g., queueing systems). Within this context of steady-state distributions, an important subdomain concerns the analysis of *rare events*. Particularly when it concerns rare events with a potentially catastrophic impact, there is a clear need to accurately estimate their likelihood (earthquakes, extreme weather conditions, simultaneous failure of multiple components of a machine, etc.). As examples, we refer to Ragone *et al.* (2018) for rare-event simulation methods in the climate context and to Rubino and Tuffin (2009) for a textbook treatment covering applications in, e.g., engineering, chemistry, and biology.

Despite the evident importance of being able to estimate steady-state rare-event probabilities, relatively little attention has been paid to the development of efficient algorithms; rare-event simulation in a finite-time horizon context received considerably more attention (focusing, e.g., on the estimation of the probability to hit a set B_1 before hitting another set B_2). The main contribution of this paper concerns the development of a broadly applicable rare-event simulation method that is tailored to the estimation of small steady-state probabilities.

In our setup, we focus on discrete-time $\mathbb{R}^d$-valued Markov chains. This framework covers a wide class of intensively used

stochastic models. For instance, it includes the numerical solutions to stochastic differential equations (SDEs), see, e.g., Kloeden and Platen (1992). In addition, various (inherently discrete-time) standard models from, e.g., finance, biology, and econometrics, fall under this umbrella. The main advantage of our proposed algorithm is its broad applicability, the fact that it does not require detailed knowledge of the system under study, and that it is fairly straightforward to implement. In the sequel, we let $(X_n)_{n\in\mathbb{N}}$ be our d-dimensional Markov chain, which we assume to admit the stationary distribution μ. We are interested in the probability that in the steady-state, the process attains a value in set B, i.e.,

$$\gamma := \mu(B) = \lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} \mathbb{1}\{X_n \in B\}. \tag{1}$$

Throughout, event B is assumed to be *rare*, entailing that γ is very small, typically of the order 10^{-4} or less (depending on the application at hand).

Our interest lies in estimating rare-event probabilities in the context of *models*, so in principle, we can do more than applying statistical methods of extreme value analysis to model data; cf. Coles *et al.* (2001) for a textbook on Extreme Value Analysis. In our setup, the steady-state distribution is not explicitly known; one therefore has to resort to simulation. The naïve Monte Carlo estimator for γ is

$$\hat{\gamma}_{\mathrm{MC}} := \frac{1}{N} \sum_{n=1}^{N} \mathbb{1}\{X_n \in B\},$$

i.e., *the average number of visits to set B until time N*, which is known to be extremely inefficient when B is rare; see, e.g., Asmussen and Glynn (2007). Informally, one needs prohibitively many samples in order to obtain a reasonably accurate estimate of γ; the number of samples required to obtain an estimate of given precision is inversely proportional to γ. In many cases, especially while working with complex or high-dimensional systems, where the integration of the model is time consuming, such computation might not be feasible.

An additional complication is that sampling directly from the steady-state distribution can be challenging. In our new method, we settle this issue by dissecting the paths of the underlying Markov chain into *recurrency cycles*. For an arbitrary set A, we say that a recurrency occurs each time $(X_n)_{n\in\mathbb{N}}$ crosses A *inwards*, i.e., each time the event $\{X_{n-1} \notin A, X_n \in A\}$ occurs. Assuming the process is in stationarity, γ is equal to *the average amount of time spent in B between two visits to set A divided by the average length of a recurrency cycle.*

An example of a recurrency cycle is shown in Fig. 1. It starts at P_1 and ends at P_5; the time spent in set B is the time spent between states P_3 and P_4. Note that recurrency is defined with respect to A; it is not necessary that the system enters B during a recurrency cycle.

In our algorithm, we separately estimate the numerator (expected time spent in B during a single recurrency cycle) and the denominator (expected length of a single recurrency cycle). Here, two challenges arise. The first concerns the choice of set A. Any A could in principle be used, but in order to maximize the efficiency of the algorithm, it should be chosen so as to minimize the expected time spent between visits to set A. The second challenge is posed by the rarity of visiting B within a cycle. To tackle this issue, we propose the use of Multilevel Splitting (MLS), see Garvels (2000) and Rubino and Tuffin (2009), but we remark that instead of MLS other methods could be chosen. These alternatives include genealogical particle analysis (see, e.g., Del Moral and Garnier, 2005), RESTART (see, e.g., Villén-Altamirano and Villén-Altamirano, 2011), adaptive multilevel splitting (see, e.g., Cérou and Guyader, 2007), fixed-effort and fixed number of successes versions of multilevel splitting (see, e.g., Amrein and Künsch, 2011) and importance sampling (see, e.g., Heidelberger, 1995). We emphasize that we do not seek to compete with any of the aforementioned methods but rather introduce a new overarching framework, in which all these methods can be used to assess stationary performance metrics. We have chosen to work with MLS mostly for its conceptual simplicity and intuitive use.

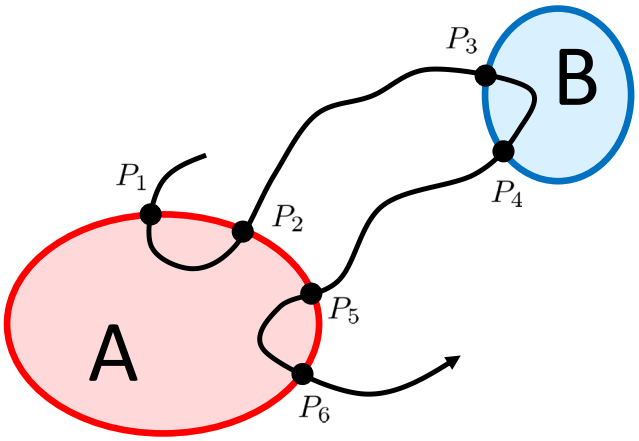

FIG. 1. An example of a recurrency cycle. The cycle begins at P_1, where the Markov chain enters set A from the outside and ends at P_5 where the chain enters A again (and the next recurrency cycle begins).

The algorithm we propose is inspired by expressions for steady-state probabilities resulting from the theory of *regenerative processes*. Regeneration instances dissect the path of the process into probabilistically identical, independent segments. For regenerative processes, we have that γ equals the average amount of time spent in B in a regeneration cycle divided by the average length of a regeneration cycle. For more background, we refer to Crane and Iglehart (1975) and Asmussen (2008) or (in a more informal language) Henderson and Glynn (1999). In our setup, with its uncountable state space and a steady-state distribution potentially lacking atoms, we cannot straightforwardly construct regeneration points. We therefore develop an approach that relies on the recurrency cycles introduced above, so as to set up a scheme that yields probabilistically identical (but not necessarily independent) cycles. We refer to Goyal *et al.* (1992) for an algorithm corresponding to the setting in which set A consists of finitely many elements (which inspired us to develop our algorithm). We also mention that a large subclass of general (continuous) state-space Markov chains, called *positive Harris*, is regenerative. However, constructing regeneration cycles in this context is typically technically difficult, and in addition, the implementation may be computationally inefficient due to excessively long cycle lengths; see Henderson and Glynn (2001).

The article is organized as follows. In Sec. II, we discuss preliminaries, such as the basic theory of general state-space Markov chains. We also give an alternative representation of parameter γ based on the recurrent structure of a Markov Chain in Theorem 1. Relying on this alternative representation, in Sec. III, we introduce a new algorithm for the estimation of γ, which we coin Recurrent Multilevel Splitting (RMS). In Sec. IV, we establish (in a simplified setting) the optimal parameters for the RMS algorithm and provide implementation-related guidelines. Theorem 3 in Appendix C establishes the asymptotic efficiency of the RMS algorithm. A technical derivation of the optimal parameters is given in Appendix B. In Sec. V, we test the method on a set of numerical examples, we discuss which factors affect the method's performance, and we provide heuristics. Finally, in Sec. VI, we discuss possible extensions of the algorithm and give a summary. Appendix A consists of a collection of required technical results.

II. PRELIMINARIES

Here, we introduce concepts used later in Sec. III such as (Harris) recurrence, the stationary measure, and *recurrency cycles*.

A. Continuous state-space Markov chains

In this subsection, we provide some background on the (well-established) theory of stability of discrete-time Markov chains with a general (continuous) state-space. The underlying theory can be found in textbooks on Markov chains; our notation is in line with the one used in Meyn and Tweedie (2012).

The theory of stability for general state-space time-discrete Markov chains differs from the one for its *finite* (or countable) state-space counterpart. Due to the continuous state space, multiple visits to the same state may happen with probability 0. This explains why the classic notion of *irreducibility* and *recurrence* of states has been generalized to *sets* (rather than states). In this setting, one typically works with the concept of the so-called *positive Harris recurrent* chains: sets of states are guaranteed to be visited infinitely often, with in addition, a *finite expected return time*. Effectively, all Markov chains with an invariant probability distribution are positive Harris (with an exception of pathological, custom-made examples); see Meyn and Tweedie (2012, Section 9) for a rigorous treatment of the topic.

Let $(X_n)_{n\in\mathbb{N}}$ be a Markov chain taking values in $\mathbb{R}^d$ with a transition kernel $P(x, \mathrm{d}y)$, meaning that the distribution of X_{n+1} conditional on $X_n = x$ is given by

$$\mathbb{P}(X_{n+1} \in A \mid X_n = x) = \int_A P(x, \mathrm{d}y) \tag{2}$$

for measurable sets $A \subseteq \mathbb{R}^d$. We denote $P(x, A) := \int_A P(x, \mathrm{d}y)$. Then, the stationary distribution μ satisfies the relation

$$\mu(A) = \int_{\mathbb{R}^d} \mu(\mathrm{d}x) P(x, A). \tag{3}$$

For an arbitrary probability measure ν, we define the conditional probability and expectation by $\mathbb{P}_\nu(\cdot) = \mathbb{P}(\cdot \mid X_0 \sim \nu)$ and $\mathbb{E}_\nu(\cdot) = \mathbb{E}(\cdot \mid X_0 \sim \nu)$, respectively. In particular, when ν corresponds to a point mass at x, we use the compact notations $\mathbb{P}_x(\cdot) = \mathbb{P}(\cdot \mid X_0 = x)$ and $\mathbb{E}_x(\cdot) = \mathbb{E}(\cdot \mid X_0 = x)$, respectively.

B. Recurrent structure of a Markov chain

As mentioned in Sec. I, a large class of general state-space Markov chains (more specifically, the class of positive Harris recurrent Markov chains) allows a regenerative structure; see, e.g., Henderson and Glynn (2001). However, for application purposes, it is often difficult to sample the regeneration times. Moreover, even when it is possible to sample these, the implementation is often inefficient due to the long cycle lengths—in fact, the regeneration may be a rare event itself.

There are many other ways to decompose a Markov chain into cycles. In this paper, we propose to work with cycles that start with an inward crossing of a set A (i.e., entering A from the outside). We denote the time of the $(k+1)$-th inward crossing by S_k, i.e., with $S_{-1} := 0$,

$$S_k := \inf\{n > S_{k-1} : X_{n-1} \notin A, X_n \in A\}. \tag{4}$$

Then, we define the paths within the cycles through

$$\mathcal{C}_k := \big(X_n : S_{k-1} \le n < S_k - 1\big). \tag{5}$$

With a k-th cycle, we associate the *cycle length*

$$L_k := S_k - S_{k-1}, \tag{6}$$

and the *cycle origin* (or starting point)

$$X_k^A := X_{S_{k-1}}. \tag{7}$$

We call A the *recurrency set* and $\mathcal{C}_1, \mathcal{C}_2, \ldots$ *recurrency cycles*. Under the assumption that the process $(X_n)_{n\in\mathbb{N}}$ starts in the steady-state ($X_0 \sim \mu$, that is), the pairs $(\mathcal{C}_1, L_1), (\mathcal{C}_2, L_2), \ldots$ are identically distributed. However, the cycles (5) are generally not independent, as two distinct cycle origins X_k^A, X_m^A separated by a short time period $S_{m-1} - S_{k-1}$ tend to be located within the same subregion of the recurrency set. Because of this dependence, the decomposition into recurrency cycles is neither *classic* nor *wide sense regenerative*, see Definitions 3.1 and 3.3 in Kalashnikov (1994). The way we define cycles is a special case of the *almost regenerative cycles* introduced by Gunther and Wolff (1980). The interested reader is referred to the introduction of Calvin *et al.* (2006), where a more exhaustive account of different regeneration-type methods is outlined.

A single recurrency cycle reflects the behavior of the process in steady-state. To make this claim more precise, define *the total time spent in set B within the k-th cycle*

$$R_k := \sum_{n=S_{k-1}}^{S_k - 1} \mathbb{1}\{X_n \in B\}. \tag{8}$$

Since (in the cycle-stationary regime) the cycles in (5) are identically distributed, so are $R_1, R_2, \ldots$. The following theorem states that the total fraction of time that the process (X_n) spends in set B is proportional to the expected time spent in B between two consecutive inward crossings into A. Define the *frequency of recurrence* $\alpha_A := \mathbb{P}_\mu(X_0 \notin A, X_1 \in A)$.

Theorem 1. *Let $(X_n)_{n\in\mathbb{N}}$ be a positive Harris recurrent Markov chain and let μ denote its unique stationary probability measure. Let A and B be measurable sets such that $\mu(A) \in (0,1)$. Let L_1 be as defined in (6), R_1 as defined in (8), and $T_B := \mathbb{E}_\mu R_1$. Then, $\mathbb{E}_\mu L_1 < \infty$,*

$$\mu(B) = \alpha_A \cdot T_B \tag{9}$$

and $\alpha_A = (\mathbb{E}_\mu L_1)^{-1}$.

Proof. See Appendix A. □

The factorization (9) of γ from Theorem 1 is the starting point from which we develop our steady-state rare-event simulation algorithm in Sec. III.

We note that an analogue of Theorem 1 holds for regenerative processes. The dissection of a Markov chain into *regeneration cycles* has one clear advantage over dissection into recurrency cycles, namely, the regeneration cycles are *independent*. Using this independence, one can easily infer the variance of an estimator based on regeneration cycles. Nonetheless, it is more attractive to use recurrency cycles than regeneration, as the latter is harder to implement and has a (much) longer expected cycle length. Moreover, in situations where it is possible to sample from the stationary distribution μ, one can simulate independent paths until the first recurrency cycle has ended such that the resulting cycles will be independent as well.

III. RECURRENT SPLITTING ALGORITHM

Our algorithm essentially relies on the result from Theorem 1, namely, the representation of γ as a product of two quantities. Thus, we divide our algorithm into two stages: first, there is the estimation of α_A (the frequency of recurrence, equal to the reciprocal of the expected cycle length) and second, the estimation of T_B (the expected time spent in set B within a recurrency cycle).

A. Estimation of α_A

While it is relatively straightforward to estimate α_A (for example, with a crude Monte Carlo method), the choice of the recurrency set A is non-trivial. In this section, we assume that A has already been chosen; the choice of A is discussed in Sec. IV B.

In typical situations, one can generate sample paths of X_n by simulation but it is not possible to *exactly* sample from the stationary distribution. Even though the law of X_n converges to μ weakly, as $n \to \infty$, at any fixed time n, the law of X_n is not exactly μ. Perhaps, the most straightforward method to estimate α_A in this setting is the method of batch-means. It relies on dissecting a path of the Markov chain of length N into $m \in \mathbb{N}$ batches of equal length, and calculating the sample frequency of entering the set A for each batch. More specifically, with $M := [N/m]$,

$$\hat{\alpha}_k := \frac{1}{M} \sum_{n=(k-1)M+1}^{kM} \mathbb{1}\{X_{n-1} \notin A, X_n \in A\}$$

and then the batch-means estimator is

$$\hat{\alpha}_A^{\mathrm{BM}} := \frac{1}{m} \sum_{k=1}^{m} \hat{\alpha}_k. \tag{10}$$

Let s_{BM}^2 be the sample variance of $\hat{\alpha}_1, \ldots, \hat{\alpha}_m$ and t_{m-1} a Student's t distribution with $m-1$ degrees of freedom. Then, due to the "near independence" between the batches, under appropriate regularity assumptions

$$\sqrt{m}(\hat{\alpha}_A^{\mathrm{BM}} - \alpha)/s_{\mathrm{BM}} \xrightarrow{\mathrm{d}} t_{m-1}, \tag{11}$$

as $N \to \infty$, with "$\xrightarrow{\mathrm{d}}$" denoting convergence in distribution. For more details and background, we refer to, e.g., Asmussen and Glynn (2007).

We remark that when an exact sampling procedure from μ is available, then it might be more efficient to use the following Monte Carlo estimator. Generate M independent pairs

$$(X_0^{(1)}, X_1^{(1)}), \ldots, (X_0^{(M)}, X_1^{(M)}),$$

with (for all $i = 1, \ldots, M$) $X_0^{(i)} \sim \mu$ and $X_1^{(i)}$ distributed according to the dynamics of the Markov chain (2) conditional on the value of $X_0^{(i)}$. The Monte Carlo estimator

$$\hat{\alpha}_A^{\mathrm{MC}} := \frac{1}{M} \sum_{i=1}^{M} \mathbb{1}\{X_0^{(i)} \notin A, X_1^{(i)} \in A\} \tag{12}$$

is unbiased, $\mathbb{V}\mathrm{ar}\, \hat{\alpha}_A^{\mathrm{MC}} = \alpha_A(1-\alpha_A)/M$, and, as $M \to \infty$,

$$\sqrt{M}(\hat{\alpha}_A^{\mathrm{MC}} - \alpha)/s_{\mathrm{MC}} \xrightarrow{\mathrm{d}} N(0,1), \tag{13}$$

with s_{MC}^2 being the sample variance.

Whether exact simulation from μ is available or not, both methods allow for the construction of confidence intervals based on the weak convergence results (11) and (13). It should be clear that set A should be chosen such that α_A is not prohibitively small so that methods (10) and (12) are computationally efficient. Otherwise, the estimation of α_A would be a rare event simulation problem itself (which we obviously want to avoid).

B. Estimation of T_B

The second stage of the algorithm concerns the estimation of T_B, as defined in Theorem 1. This step is the more challenging one, as the quantity T_B is expected to be very small. We resort to rare-event simulation methods. For clarity of exposition, throughout this section, we assume that the chain $(X_n)_{n\in\mathbb{N}}$ is stationary, $S_0 = 0$ and we drop the subscript in $\mathbb{P}_\mu$ and $\mathbb{E}_\mu$ (i.e., we write simply $\mathbb{P}$ and $\mathbb{E}$, respectively). We also assume that we can sample from the distribution of the cycle starting point X_1^A (note that $X_1^A, X_2^A, \ldots$ are all identically distributed). If we cannot, then we sample from X_1^A approximately; this is discussed in Sec. III C.

We first introduce some notation. Define

$$\tau_B := \inf\{n > 0 : X_n \in B\},$$
$$\tau_A^{\mathrm{in}} := S_1 = \inf\{n > 0 : X_{n-1} \notin A, X_n \in A\},$$
$$R_+ \overset{\mathrm{d}}{:=} (R_1 \mid R_1 > 0), \tag{14}$$
$$p_B := \mathbb{P}(\tau_B < \tau_A^{\mathrm{in}}), \tag{15}$$

with "$\stackrel{\mathrm{d}}{=}$" denoting equality in distribution. Note that $\tau_A^{\mathrm{in}} - 1$ marks the end of the first recurrency cycle. Since $\{R_1 > 0\} = \{\tau_B < \tau_A^{\mathrm{in}}\}$, p_B is the probability of reaching B within a cycle, and R_+ is a random variable distributed as the total time spent in the set B within a cycle conditioned on the cycle reaching set B. As was noted in Garvels (2000),

$$\mathbb{E}(R_1) = \mathbb{P}(R_1 > 0) \cdot \mathbb{E}(R_1 \mid R_1 > 0).$$

This entails that

$$T_B = \mathbb{P}(\tau_B < \tau_A^{\mathrm{in}}) \cdot \mathbb{E}(R_1 \mid \tau_B < \tau_A^{\mathrm{in}}) = p_B \cdot \mathbb{E}R_+. \tag{16}$$

The estimation of p_B is a classic rare-event simulation problem, for which various methods have been developed. Following Garvels (2000), we propose to use a Multilevel Splitting (MLS) algorithm to estimate T_B (but, as we mentioned before, other approaches could be followed as well). There are a number of variations of the MLS algorithm; we chose to rely on its simplest version (called "Fixed Splitting"). The following exposition aligns with Amrein and Künsch (2011).

As mentioned, the naïve Monte Carlo method is inefficient for the estimation of small probabilities, because of the computational effort wasted on simulating irrelevant paths. The core idea behind the MLS method is to *split* the path of the process when it approaches B. This way, we have more control over the simulation, by forcing the process into interesting regions. In order to implement the MLS algorithm, one must first choose an *importance function* $H : \mathbb{R}^d \to [0, 1]$ which assigns an *importance value* to every possible state. H should be chosen such that $H(x) = 1$ if and only if $x \in B$ and $H(x) = 0$ for $x \in A$. We postpone the discussion about the choice of the importance function to Sec. IV B.

We now formally introduce the MLS algorithm. First, divide the interval $[0, 1]$ into m subintervals with endpoints

$$0 = \ell_0 < \ell_1 < \cdots < \ell_m = 1 \tag{17}$$

and define the corresponding stopping times and events

$$\tau_k := \inf\{n \geq 0 : H(X_n) \geq \ell_k\}, \quad D_k := \{\tau_k < \tau_A^{\mathrm{in}}\} \tag{18}$$

for $k \in \{0, \ldots, m\}$. Note that τ_k is the first time an importance value greater or equal to ℓ_k has been reached; in particular, $\tau_m = \tau_B$ and $\tau_0 = 0$ so that $X_{\tau_0} \stackrel{\mathrm{d}}{=} X_1^A$. Finally, let

$$p_k := \mathbb{P}(D_k \mid D_{k-1}), \quad k \in \{1, \ldots, m\},$$

and $p_0 = 1$, to which we refer as *conditional probabilities*. From definition (18), we have $\mathbb{P}(D_m) = p_B$ and since $D_0 \subseteq D_1 \subseteq \cdots \subseteq D_m$, we conclude

$$p_B = \prod_{k=0}^{m} p_k.$$

Finally, define[!h] *splitting factors* $n_0, n_1, \ldots, n_m \in \mathbb{N}$, representing the number of independent continuations of the process that are sampled when reaching the respective importance levels. Here, n_0 plays a special role, as it is a number of independent MLS estimators; the final estimator will be a mean of n_0 independent MLS estimators. By virtue of this independence, we are able to estimate the variance of the final estimator. For simplicity, in the following, it is assumed that $n_0 = 1$.

Algorithm 1. (Multilevel Splitting).

1. *Set* $k := 0$, $r_0 := 1$, *sample* $X_0^1 \sim X_1^A$.
2. *In the k-th stage, we have a sample of* r_k *entrance states* $(X_k^1, \ldots, X_k^{r_k})$, *where we denote*

$$X_k^i := X_{\tau_k}^i.$$

For each state X_k^i, *generate* n_k *independent path continuations until* $\min\{\tau_{k+1}, \tau_A^{\mathrm{in}}\}$. *The number of paths for which the event* D_{k+1} *occurred is denoted by* r_{k+1}. *Store all* r_{k+1} *states* X_{k+1}^i, *for which the event* D_{k+1} *occurred, in memory.*

3. *If* $r_{k+1} = 0$, *then stop the algorithm and put* $\hat{p}_B := 0$, $\hat{T}_B := 0$.
4. *If* $k < m - 1$, *then increase* k *by one and go back to step 2; otherwise put*

$$\hat{p}_B := \frac{r_m}{\prod_{k=0}^{m-1} n_k}. \tag{19}$$

5. *If* $r_m = 0$, *then return* $\hat{T}_B = 0$; *otherwise, for each state* X_m^i *generate* n_m *independent path continuations until* τ_A^{in}. *For each of these* $r_m n_m$ *continuations record the time spent in set B*

$$\hat{R}_+^{(j)} := \sum_{k=\tau_m}^{\tau_A^{\mathrm{in}}-1} \mathbb{1}\{X_k \in B\}.$$

Calculate the total time spent in B by

$$r_{m+1} := \sum_{j=1}^{r_m n_m} \hat{R}_+^{(j)}. \tag{20}$$

6. *The final estimator is*

$$\hat{T}_B := \frac{r_{m+1}}{\prod_{k=0}^{m} n_k}. \tag{21}$$

Theorem 2. *The estimators* $\hat{p}_B$ *and* $\hat{T}_B$, *as defined in (19) and (21), are unbiased estimators for* p_B *and* T_B, *respectively.*

The following proof is based on notes of the *Summer School in Monte Carlo Methods for Rare Events* that took place at Brown University, Providence RI, USA in June 2016 (authored by J. Blanchet, P. Dupuis, and H. Hult). It is noted that various alternative derivations can be constructed; see, e.g., Asmussen and Glynn (2007).

Proof of Theorem 2. Let $\bar{X}_{i,j}$ be labeling all descendants of the original particle, with i indexing time and j indexing the descendant. All descendants $\bar{X}_{\cdot,j}$ are identically distributed (but not independent). Now suppose that each particle has an evolving weight $w_{i,j}$. Concretely, this means that when a particle crosses a threshold ℓ_k, it is split into n_k particles and its weight is divided equally among its descendants (i.e., each of them obtaining a share $1/n_k$ of $w_{i,j}$). Each particle that reaches set B has been split m times, and its weight is thus $1/\prod_{k=1}^{m} n_k$. For particles that did not reach set B, we artificially split these particles (keeping them in A) for the remaining thresholds so that the total number of particles is $\prod_{k=1}^{m} n_k$, each of equal weight. Then, using the fact that the descendants are identically distributed,

we obtain

$$\mathbb{E}\hat{T}_B = \mathbb{E}\Bigg(\sum_{j=1}^{\prod_{k=1}^m n_k} \frac{1}{\prod_{k=1}^m n_k} \sum_i \mathbb{1}\{\bar{X}_{i,j} \in B\}\Bigg)$$
$$= \mathbb{E}\sum_i \mathbb{1}\{\bar{X}_{i,1} \in B\} = T_B.$$

Analogously, $\mathbb{E}\hat{p}_B = p_B$, which ends the proof. □

We remark that, with $r_1, \ldots, r_m$ as defined in Algorithm 1, the same arguments as the ones featuring in the proof of Theorem 2 imply the unbiasedness of the estimators for $\mathbb{P}(D_k)$

$$\mathbb{E}\left(\frac{r_k}{\prod_{i=0}^{k-1} n_i}\right) = \mathbb{P}(D_k) = p_1 \cdots p_k. \tag{22}$$

C. Estimation of γ

As already mentioned at the beginning of Sec. III, the final estimator for γ is the product $\hat{\gamma} := \hat{\alpha}_A \cdot \hat{T}_B$. In the description of the MLS algorithm, in Step 1, we tacitly assumed that we can sample the recurrency cycle origin X_1^A. As this is typically not the case, we sample X_1^A approximately, in the following way. During the estimation of α_A with the batch-means method (10), we store each inwards crossing to set A and we bootstrap these states in Step 1 of Algorithm 1.

We thus end up with the following algorithm for estimating the rare-event probability γ, as defined in (1).

Algorithm 2. (Recurrent Multilevel Splitting).

1. *Choose a recurrency set A satisfying the assumptions of Theorem 1 and an importance function $H : \mathbb{R}^d \to [0, 1]$.*
2. *Estimate α_A using the batch-means method (10), and return $\hat{\alpha}_A$. Store the locations of the cycle origins in the set $\mathcal{S}_{\text{rec}} := \{X_1^A, X_2^A, \ldots\}$.*
3. *Estimate T_B using the Multilevel Splitting algorithm (Algorithm 1); in Step 1, sample the origin X_0^1 uniformly from S_{rec}. The output is $\hat{T}_B$.*
4. *The final estimator is*

$$\hat{\gamma} := \hat{\alpha}_A \cdot \hat{T}_B. \tag{23}$$

It is assumed that set S_{rec} is "representative enough" to make sure that resampling from S_{rec} can be interpreted as taking i.i.d. samples of X_1^A in the stationary regime. Under this assumption, the estimators $\hat{\alpha}_A$, $\hat{T}_B$ are independent and the variance of $\hat{\gamma}$ can be inferred using the sample variance of $\hat{\alpha}_A$ and $\hat{T}_B$. However, in our numerical experiments in Sec. V, we do not assume this independence to get an estimate of the variance. Instead, we run Algorithm 2 multiple times, resulting in multiple estimates $\hat{\gamma}$ from which we obtain a reliable estimate for the variance of $\hat{\gamma}$. For implementation details, see Sec. V A.

IV. CHOICE OF PARAMETERS

In a rare-event setting, both the expectation and the variance of an estimator are very small, so that the variance itself is not a meaningful measure of accuracy. Instead, it makes sense to look at its value relative to the expectation, i.e., the *relative error* (RE)

$$\text{RE}^2(\hat{\gamma}) := \mathbb{E}(\hat{\gamma} - \gamma)^2/\gamma^2.$$

An estimator with a lower relative error is not necessarily preferred; a more meaningful criterion involves the corresponding total computational time (or: *workload*), which we denote $W(\hat{\gamma})$; see the beginning of Sec. V A for more details. In Sec. IV A, we consider a setting, in which we can derive optimal parameters of the MLS estimator by minimizing the workload under a constraint on the relative error [i.e., $\text{RE}^2(\hat{\gamma}) \le \rho$ for a given accuracy $\rho > 0$].

A. Simplified setting

Due to possible dependencies between the number of successes $r_1, \ldots, r_m$, there is no tractable general expression for the variance of MLS estimator. A typical assumption made in the literature is to assume some sort of independence between them and to study the variance afterwards. With τ_k, D_k defined as in (18) and R_+ as defined in (14), we assume

(I) for all $k \in \{1, \ldots, m\}$,

$$\mathbb{P}(D_k \mid D_{k-1}, X_{\tau_{k-1}}) \equiv \mathbb{P}(D_k \mid D_{k-1}) = p_k,$$

(II) for all X_{τ_m},

$$\left(R_1 \mid R_1 > 0, X_{\tau_m}\right) \overset{\text{d}}{=} (R_1 \mid R_1 > 0) =: R_+.$$

Assumption (I) has been proposed in Amrein and Künsch (2011). It states that the probability of reaching the k-th importance level, given the $(k-1)$-st level has been reached, is constant over all possible entrance states. Assumption (II) states that the time spent in the rare set B within a cycle, conditioned on the set B has been reached, does not depend on the position of the entrance state to B. In principle, we have the possibility to choose the set A and the importance function $H(\cdot)$ such that Assumption (I) is satisfied; see the discussion in Sec. IV B. Whether Assumption (II) holds or not is effectively problem specific, in the sense that we do not have control over it due to the fact that the set B is given. We argue that for a large class of problems, there exists a most likely point of entry X_{τ_B} to B, which implies (II) approximately. We emphasize that Assumptions (I–II) are not required for the RMS algorithm to work, but if they are fulfilled, optimality results can be derived. Under (I-II), we find the squared relative error of $\hat{T}_B$

$$\text{RE}^2(\hat{T}_B) = \sum_{k=1}^m \frac{(1-p_k)/p_k}{\prod_{j=0}^{k-1} n_j p_j} + \frac{\text{RE}^2(R_+)}{\prod_{j=0}^m n_j p_j}. \tag{24}$$

We derive (24) in Appendix A. Following the approach of Amrein and Künsch (2011), in Appendix B, we derive the optimal parameters $m, p_1, \ldots, p_m, n_0, \ldots, n_m$ for the MLS algorithm; here, optimality refers to the property that the expected computational time is minimized under the constraint for the relative error $\text{RE}^2(\hat{T}_B) \le \rho$ for a given accuracy $\rho > 0$. It is worth noting that the optimal number of thresholds m is roughly equal to $|\log p_B|$ with conditional probabilities p_k all equal to approximately 0.2. What is more, the optimal solution satisfies $n_k p_{k+1} = 1$ for $k \in \{1, \ldots, m-1\}$, so we can choose $n_k = 5$. This so-called *balanced growth* (see Garvels, 2000) ensures that, on average, n_0 paths are sampled in each stage of the

algorithm (with an exception of the last stage, which corresponds to the estimation of R_+). The optimal workload reads

$$W(\hat{T}_B) = \frac{1}{q}\left[\frac{c\,|\log p_B|}{\sqrt{2c-1}} + \mathrm{RE}(R_+)\right]^2, \tag{25}$$

with a constant c defined as below display (B4). As already mentioned, a rigorous derivation of this result can be found in Appendix B, and the exact values of the optimal parameters $m, p_1, \ldots, p_m, n_0, \ldots, n_m$ in Eq. (B4). In all our numerical experiments in Sec. V, we spend an initial portion of computational time on a rough estimation of p_B and $\mathrm{RE}(R_+)$ in order to find a sufficiently accurate approximation of the optimal parameters. See Sec. V A for a more detailed account of the implementation details.

The optimal workload in (25) is proportional to $(\log p_B)^2$, which offers a huge gain in efficiency, compared with the Monte Carlo method (C4) (whose workload is inversely proportional to p_B). We derive efficiency results in Appendix C; in particular, Theorem 3 proves that RMS is logarithmically efficient under specific assumptions.

B. Choice of recurrency set and importance function

In Sec. IV A, we have seen that under Assumptions (I–II), the MLS method is particularly efficient. As already mentioned, the level up to which Assumption (I) is fulfilled depends on both the choice of the recurrency set and the importance function; we thus aim to choose A and $H(\cdot)$ in such a way that (I) is approximately satisfied. At the same time, we would like to choose A so as to maximize α_A, so that the batch-means estimator $\hat{\alpha}_A$ [as defined in (10)] is computationally efficient as well. These two requirements are often conflicting and one must in the end strike a proper balance between them.

For each k, Assumption (I) concerns the choices of both A and $H(\cdot)$. However, it implies a property that relates to the choice of A only, namely, the probability of reaching set B within a recurrency cycle is independent of the initial point

$$\mathbb{P}(\tau_B < \tau_A^{\mathrm{in}} \mid X_1^A) \equiv p_B.$$

Thus, Assumption (I) implies that

$$X_1^A \overset{\mathrm{d}}{=} \left(X_1^A \mid R_1 > 0\right) =: X_+^A \tag{26}$$

informally, there is independence between the origin of the cycle on one hand, and the random variable $\mathbb{1}\{R_1 > 0\}$ (indicating whether set B has been reached within a cycle) on the other hand. Intuitively, the smaller the set A is, the more closely (26) is satisfied but also, the smaller α_A is. In particular, (26) trivially holds when A consists of one point only, but then $\alpha_A = 0$. In Sec. V B 3, we give an example of a setting in which (26) is violated, but one can imagine that in many situations (26) "roughly holds." Thus, for practical purposes, it is desirable that *set A maximizes α_A while it also approximately satisfies (26)*. In full generality, it is not an easy task to fulfill both aims.

A poorly chosen importance function will lead the split particles into uninteresting regions, or it will force the paths to hit the rare set in an unlikely fashion. This potentially leads to low efficiency of the MLS algorithm. Given that we have already chosen a set A satisfying (26), there exists an importance function guaranteeing (I) to be satisfied

$$H(x) := \mathbb{P}_x(\tau_B \le \tau_A^{\mathrm{in}}).$$

Of course this insight is of theoretical value only: if we knew the quantity on the right-hand side, then we would not even have to use the MLS algorithm. However, also

$$H_g(x) := g[\mathbb{P}_x(\tau_B \le \tau_A^{\mathrm{in}})],$$

with $g : [0,1] \to \mathbb{R}$ any increasing function, satisfies (I). This already gives a helpful guideline for the choice of H. Namely, *the states from which it is more likely to visit B before returning to A should have larger importance*. When an approximation or asymptotic behavior of $\mathbb{P}_x(\tau_B \le \tau_A^{\mathrm{in}})$ is available it might be useful to use it as an importance function. In Dean and Dupuis (2009), a large-deviations based approach to the choice of importance function is discussed.

Sometimes, a so-called *distance-based* importance function can be a good choice. This function is basically

$$H(x) := \mathrm{dist}(x, B) = \inf\{\|x - a\| : a \in B\},$$

normalized in such a way that $H(x) = 1$ iff $x \in B$ and $H(x) = 0$ for $x \in A$. This importance function can be a good choice for systems whose paths conditioned on $\{\tau_B < \tau_A^{\mathrm{in}}\}$ are effectively gradually driven towards B. In contrast, distance-based importance function will be a poor choice for systems for which it is most likely to reach rare set B by first getting away from it. In Sec. V, we include examples of problems for which a distance-based importance function is a good choice, but also one in which it does not work well.

In some cases, we may have already chosen a particular *shape* of the set A (e.g., an ellipsoid, half-space, or multidimensional cube) which can be parametrized by a single parameter $\ell \in \mathbb{R}$. Even better, if we have already chosen an importance function, then a level set

$$A(\ell) = \{x \in \mathbb{R}^d : H(x) \le \ell\}$$

could be a good choice. In any case, we should choose ℓ to maximize $\alpha_{A(\ell)}$. We propose to use a crude estimator to find ℓ^*: we find a maximizer of $\alpha_{A(\ell)}$ by putting

$$\hat{\ell}^* := \arg\max\left\{\sum_{n=0}^{N} \mathbb{1}\{X_n \notin A(\ell), X_{n+1} \in A(\ell)\}\right\}. \tag{27}$$

Quantile validation. While it is not clear in general how to choose A such that it satisfies (26), one can statistically test whether (26) holds after the choice of A has been made. We now propose one particular method to do so that can be used in combination with the RMS algorithm. In Step 2 of Algorithm 2, we calculate and store the *maximum importance attained within cycles*, i.e.,

$$H_k^{\max} := \max\{H(x) : x \in \mathcal{C}_k\},$$

with $\mathcal{C}_k$ as defined in (5). Assuming a good importance function has been chosen, the cycle origins corresponding to the highest importance should also be approximately distributed as X_+^A. This gives us means of comparing the distributions of X_1^A and X_+^A. Let N_{rec} be the total number of pairs $(X_k^A, H_k^{\max})$ obtained in Step 2 of Algorithm 2. Let

$$\sigma : \{1, \ldots, N_{\mathrm{rec}}\} \to \{1, \ldots, N_{\mathrm{rec}}\}$$

be a permutation ordering $(H_k^{\max})_{1\geq k\geq N_{\text{rec}}}$ into a non-decreasing sequence, i.e.,

$$H_{\sigma(1)}^{\max} \leq H_{\sigma(2)}^{\max} \leq \cdots \leq H_{\sigma(N_{\text{rec}})}^{\max}.$$

Now, choose $q \in (0,1)$ and let

$$S_{\text{rec}}^q := \{X_{\sigma(\lfloor (1-q)N_{\text{rec}} \rfloor)}^A, \ldots, X_{\sigma(N_{\text{rec}})}^A\}. \tag{28}$$

That is, S_{rec}^q is a subset of S_{rec} which contains the cycle origins corresponding to the fraction q of values with highest importance. In particular, $S_{\text{rec}}^1 = S_{\text{rec}}$. Then, S_{rec} and S_{rec}^q (for small q) can be thought of as sets of samples from the random variables X_1^A and X_+^A, respectively. Various tests can now be performed to compare, e.g., the means or variances; alternatively, QQ-plots can be made or histograms can be compared.

V. NUMERICAL EXPERIMENTS

The aim of this section is to test the RMS method on a series of specific examples. The examples range from simple cases, where the ground truth is known, to more complicated dynamical systems, where the ground truth is unknown and we can only compare to estimates obtained with Monte Carlo (MC) methods. In Sec. V B 3, we also carefully look into an example where the RMS method (with a naïve choice of the importance function) does not perform that well; we discuss why this was to be expected. It will be seen throughout that RMS is superior to MC in terms of the computational time needed to achieve a desired level of accuracy; in extreme cases, like in Sec. V C, the RMS method can be three orders of magnitude faster than MC (and the efficiency gain is expected to be even greater as γ decreases).

A. Implementation details

As already mentioned in Sec. IV, the relative error of an estimator is not always a meaningful measure of its performance, as it does not take the workload into account. We therefore compare RMS with MC using the ratio of *work normalized squared relative errors*; see, e.g., Kroese *et al.* (2013). In particular, we define

$$\text{Eff}(\hat{\gamma}) = \frac{W(\hat{\gamma}^{\text{MC}})}{W(\hat{\gamma})} \cdot \frac{\text{RE}^2(\hat{\gamma}^{\text{MC}})}{\text{RE}^2(\hat{\gamma})}. \tag{29}$$

This value can be interpreted as the ratio of the computational cost of MC to the cost of RMS when both methods reach the same accuracy (same relative error). Clearly, the larger the $\text{Eff}(\hat{\gamma})$ is, the more efficient the RMS method is in comparison with Monte Carlo.

In each of our experiments, the underlying Markov chain $(X_n)_{n\in\mathbb{N}}$ represents the numerical solution to a d-dimensional Stochastic Differential Equation (SDE) using an explicit Euler scheme, with time step $h > 0$; see, e.g., Kloeden and Platen (1992). We remark that the time discretization potentially has a significant effect on a the underlying value of γ, especially, in the rare-event setting; see the recent systematic study of Bisewski *et al.* (2018). However, in the context of this article, we only focus on discrete recursions that arise from numerical time integration schemes. For these recursions, we compare RMS with the corresponding Monte Carlo results; we do not aim at studying the behavior as $h \downarrow 0$.

Notice that our method relies on properties of discrete-time processes, in particular, in the definition of the recurrency cycles. More specifically, in the corresponding continuous-time model recurrency cycles are ill-defined, as a set may be entered and left infinitely often in a time interval of finite length. This feature could potentially lead to computational issues when working with a small time step h. However, one can easily circumvent the problem and still integrate the process with arbitrarily small h_0 but store values every $h > h_0$. Note that the discretization error depends only on h_0 (and not h), since h_0 determines the stationary distribution. In fact, this is what we do in Sec. V C, where the process is integrated with $h_0 = 10^{-4}$, but it is stored only every $h = 10^{-2}$.

In each experiment, the rare event B is a half-space parametrized by $u \in \mathbb{R}$

$$B(u) = \{(x_1, \ldots, x_d) \in \mathbb{R}^d : x_1 \geq u\}. \tag{30}$$

In other words, the probability under consideration corresponds to the first dimension attaining high values in stationarity

$$\gamma(u) := \mathbb{P}_\mu[X_0 \in B(u)] \tag{31}$$

for large u. Furthermore, in each experiment, we choose the recurrency set A to be a half-space parametrized by ℓ (where the value of ℓ is chosen depending on the particular experiment)

$$A(\ell) = \{(x_1, \ldots, x_d) \in \mathbb{R}^d : x_1 \leq \ell\}. \tag{32}$$

We use a distance-based importance function, i.e.,

$$H(x_1, \ldots, x_d) = \begin{cases} 0, & x_1 \leq 0, \\ x_1/u, & x_1 \in (0,u), \\ 1, & x_1 \geq u. \end{cases} \tag{33}$$

We now provide more details on our implementation of Algorithm 2. In Step 2, we estimate α_A using the method of batch means as in (10); the number of iterations of the Markov chain N is chosen such that S_{rec} consists of roughly 10^4 inwards crossings of A. In Step 3, we want to choose parameters $m, n_0, \ldots, n_m, \ell_1, \ldots, \ell_m$ for the Multilevel Splitting in such a way that the workload is minimized and the resulting estimator satisfies

$$\text{RE}(\hat{T}_B) = 5 \cdot 10^{-3}. \tag{34}$$

We run a pilot MLS with many intermediate thresholds ($m = 20$). The pilot gives us rough estimates of p_B, T_B, and $\text{RE}(R_+)$. We put the number of thresholds m and splitting factors $n_0, \ldots, n_m$ as in (B4); we emphasize that the optimal n_0 is also determined by the desired squared relative error ρ. We find the intermediate thresholds $\ell_1, \ldots, \ell_m$ following the log-linear interpolation approach from Wadman *et al.* (2014). Assuming (I–II) are satisfied, the MLS method with these parameters should give the desired relative error, as in (34). We note that in the pilot, we use the variant of MLS called "Fixed Number of Successes" developed by Amrein and Künsch (2011).

The final estimator $\hat{\gamma}$ is the mean of $N = 100$ independent replicas $\hat{\gamma}^{(1)}, \ldots, \hat{\gamma}^{(N)}$ of the RMS estimator (23) with parameters as discussed above; i.e.,

$$\hat{\gamma} := \frac{1}{N}\sum_{i=1}^{N} \hat{\gamma}^{(i)}. \tag{35}$$

This additional "Monte Carlo wrapper" around the RMS method enables us to approximate the relative error $\text{RE}(\hat{\gamma})$ with

$$\text{RE}^2(\hat{\gamma}) \approx \frac{1}{N-1}\sum_{i=1}^{N}\left(\frac{\hat{\gamma}^{(i)}}{\hat{\gamma}} - 1\right)^2 \tag{36}$$

and we can approximate $\mathrm{RE}(\hat{\alpha}_A)$ and $\mathrm{RE}(\hat{T}_B)$ in a similar way. For each experiment, we present a table with results corresponding to multiple values of the threshold u. Each table displays the final estimator $\hat{\gamma}$ as well as its estimate for $\mathrm{RE}(\hat{\gamma})$, as in (36), and $\mathrm{Eff}(\hat{\gamma})$, as in (29) based on the run of an MC estimator $\hat{\gamma}^{\mathrm{MC}}$.

Various checks can be done in order to assess the reliability of the estimator $\hat{\gamma}$. In each table, we additionally give the estimate for $\mathrm{RE}(\hat{T}_B)$; if it matches the desired relative error, i.e., $\mathrm{RE}(\hat{T}_B) \approx 5 \cdot 10^{-3}$, then this is an indication that Assumptions (I–II) are satisfied. When $\mathrm{RE}(\hat{T}_B)$ is larger than desired, it might be a result of poorly chosen intermediate thresholds $\ell_1, \ldots, \ell_m$; we propose to verify, after the algorithm has been executed, whether the estimates for all the intermediate probabilities $p_1, \ldots, p_m$ roughly equal the optimal $p_{\mathrm{opt}} \approx 0.20$. If this is the case and we still get a particularly large $\mathrm{RE}(\hat{T}_B)$, this is an indication that either the recurrency set or the importance function have not been properly chosen. In the case of violation of the former, in Sec. IV B, we proposed a test for the appropriateness of the choice of the set A. Additional verification can be performed to assess whether resampling from the set S_{rec} obtained in Step 2 of the RMS algorithm is a good approximation of taking i.i.d. samples of X_1^A. This implies that $\hat{\alpha}_A$ and $\hat{T}_B$ are independent, but if they are independent then necessarily

$$\mathrm{RE}^2(\hat{\gamma}) = \mathrm{RE}^2(\hat{\alpha}_A) + \mathrm{RE}^2(\hat{T}_B). \tag{37}$$

Thus, if (37) is not approximately satisfied, it is an indication that S_{rec} does not represent the distribution of X_1^A well. We emphasize that the relative error of $\hat{\gamma}$ presented in tables is calculated as in (36).

B. Ornstein-Uhlenbeck process

Let $(X_t)_{t\geq 0}$ be a d-dimensional Ornstein-Uhlenbeck process (d-dim OU), i.e., a process taking values in $\mathbb{R}^d$ solving the SDE

$$\mathrm{d}X_t = -QX_t\mathrm{d}t + \mathrm{d}W_t, \tag{38}$$

with $Q \in \mathbb{R}^{d\times d}$ and $(W_t)_{t\geq 0}$ denoting a standard d-dimensional Wiener process. Applying the explicit Euler numerical scheme to (38), with time step $h > 0$ yields

$$X_{n+1} = (I - Qh)X_n + Z_n, \tag{39}$$

with I the d-dimensional identity matrix I, and $Z_1, Z_2, \ldots$ i.i.d. d-dimensional standard normal random variables. It is known (Schurz, 1999) that the stationary distribution μ of (39) exists if there exists a positive-definite matrix $M = (M_{ij})_{i,j\in\mathbb{N}}$ solving

$$M = (I - Qh)M(I - Qh)^\top + hI; \tag{40}$$

then the stationary distribution μ is d-dimensional centered normal with covariance matrix M. The rare event of our interest is the exceedance of a high threshold in the first dimension under the stationary distribution [of the discrete-time Markov chain in (39)], as in (31). Equation (40) is a well-known Sylvester equation and its solution M can be found numerically so that $\gamma(u)$ can be evaluated as

$$\gamma(u) = \Phi(-u/\sqrt{M_{11}}), \tag{41}$$

with $\Phi(\cdot)$ the standard normal cdf. Knowing the ground truth $\gamma(u)$ gives us means to determine how accurate the RMS estimator $\hat{\gamma}$ is.

In Secs. V B 1–3, we study the OU process with different sets of parameters but with the same choice of the recurrency set and importance function, as in (32) and (33). First, we study the simplest case of a one-dimensional OU process. This is an "ideal" example in the sense that Assumptions (I–II) are (approximately) satisfied. Second, we study a multidimensional OU process; while the simplifying assumptions do not seem to be satisfied, they are "close enough" for the RMS method to give satisfactory results. The third case describes a two-dimensional OU process with the matrix Q chosen such that Assumptions (I–II) are not satisfied for our choice of the recurrency set and the importance function.

1. 1-dim OU

In this experiment, we put $d = 1$, $Q = 1$, $h = 0.01$. The recurrency set $A(\ell)$ and importance function $H(\cdot)$ are as in (32) with $\ell = 0$ and (33), respectively.

If we would study the stationary distribution of the original SDE driven by (38) [rather than the time-discrete numerical solution in (39)], then the paths of the process would be continuous and thus $X_1^A = 0$ a.s. Moreover, because of their continuity, these paths must cross all intermediate states $x \in (0, u)$ before reaching B. Therefore, $x \mapsto \mathbb{P}_x(\tau_B < \tau_A^{\mathrm{in}})$ is an increasing function, implying that the distance-based importance function satisfies (I) in the continuous-time case. By similar arguments, $X_{\tau_B} = u$ a.s., and hence (II) is satisfied as well in that case.

The Markov chain driven by (39) is a discrete-time approximation of (38), so the assumptions will not be satisfied *exactly*. In particular, we note that for any time step $h > 0$, the support of X_{τ_B} is the entire halfline $[u, \infty)$ because in principle the process can exceed the threshold u by any positive value upon the first entry. This shows that Assumption (II) is not satisfied. An analogous argument can be used to show that Assumption (I) is not satisfied either. Nonetheless, for a small time step $h > 0$, extreme overshooting upon the first entry (i.e., X_{τ_B} being significantly larger than u, or X_{τ_k} significantly larger than $\ell_k u$) is very unlikely. We conclude that the assumptions are satisfied approximately.

TABLE I. RMS algorithm for an 1-dim OU process. Parameters: $Q = 1$, $A = \{x_1 \leq 0\}$, $B = \{x_1 \geq u\}$; u has been chosen using (41) to match the values of γ in the first row. We have $\hat{\alpha}_A = 0.0225$ and $\mathrm{RE}(\hat{\alpha}_A) = 1.66 \times 10^{-3}$.

$\gamma(u)$	1×10^{-03}	1×10^{-04}	1×10^{-05}	1×10^{-06}	1×10^{-07}
$\hat{\gamma}$	9.94×10^{-04}	9.93×10^{-05}	9.96×10^{-06}	9.96×10^{-07}	9.96×10^{-08}
$\mathrm{RE}(\hat{\gamma})$	3.95×10^{-03}	5.45×10^{-03}	6.53×10^{-03}	6.31×10^{-03}	5.49×10^{-03}
$\mathrm{Eff}(\hat{\gamma})$	4.1	8.9	45.2	378.9	1836.2
$\mathrm{RE}(\hat{T}_B)$	3.90×10^{-03}	4.99×10^{-03}	6.42×10^{-03}	6.30×10^{-03}	5.32×10^{-03}

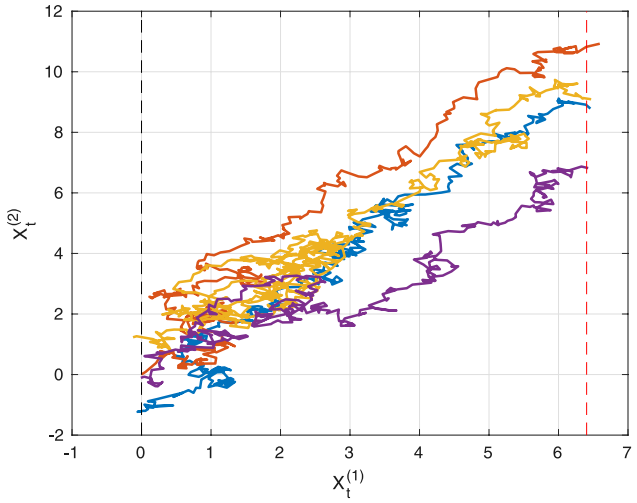

FIG. 2. 10-dim OU process. Four random realizations of recurrency cycles conditioned on reaching the rare set. The cycles have been plotted until the first hitting time of B. Parameters: $A = \{x_1 \le 0\}$, $B = \{x_1 \ge u\}$ with $u \approx 6.4$ such that and $\gamma(u) = 10^{-6}$.

Since the value of $\gamma(u)$ can be evaluated using (41), we chose the thresholds u to match the desired value of $\gamma(u)$, as in Table I. The results show that $\mathrm{RE}(\hat{T}_B) \approx 5 \cdot 10^{-3}$, as desired in (34); this is a good indication that Assumptions (I–II) are satisfied. Also, the relative error calculated under the independence assumption via (37) matches the estimated $\mathrm{RE}(\hat{\gamma})$.

Conclusions. In this setting, the RMS algorithm is very efficient, as compared with MC. The numerical results agree very well with the theoretical outcomes, confirming our observation that Assumptions (I–II) are approximately satisfied.

2. 10-dim OU, Q with real eigenvalues

In this experiment, we put $d = 10$, $h = 0.01$. The matrix $Q = (Q_{ij})_{i,j\in\{1,\dots,d\}}$ is randomly generated such that all its eigenvalues are real. The recurrency set $A(\ell)$ and importance function $H(\cdot)$ are as in (32) with $\ell = 0$ and (33), respectively.

In Fig. 2, we plot four randomly chosen recurrency cycles, projected onto the first and second dimension, which have reached the rare event B. These conditional paths seem to follow a linear pattern; similar behavior is seen in other projections (not shown). This indicates that attaining high values in the first dimension is coupled with attaining high values in the second dimension (and similar statements can be made about other dimensions). Therefore, the distance-based importance function is not expected to satisfy (I), as it does not take this behavior into account; an ideal importance function should give larger importance to states which attain *simultaneously* high values in the first and second dimension. While the distance-based importance function is not the most appropriate choice, it is still expected to give satisfactory results, as it drives the paths gradually towards the rare event.

The results of the RMS algorithm are presented in Table II. It can be seen that the values of $\mathrm{RE}(\hat{T}_B)$ do not exactly match the desired value $5 \cdot 10^{-3}$ in (34), which in view of the earlier discussion is not surprising, as we did not expect Assumptions (I–II) to hold. However, the estimates $\hat{\gamma}$ are still very accurate, and the efficiency is still excellent (relative to the MC method).

Conclusions. This experiment shows that the RMS algorithm can be effectively implemented in a multidimensional setting, even when Assumptions (I–II) are violated. This underscores the robustness of the distance-based importance function.

3. 2-dim OU, Q with complex eigenvalues

In this experiment, we put $d = 2, h = 0.01$. We choose Q to have non-real eigenvalues: for a positive θ,

$$Q(\theta) = \begin{bmatrix} 1 & \theta \\ -\theta & 1 \end{bmatrix}. \tag{42}$$

The drift generates a rotating (or spiraling) motion of the paths, with the speed of rotation increasing as θ increases. We compare the efficiency of the RMS method for increasing values of θ. The recurrency set $A(\ell)$ and importance function $H(\cdot)$ are as in (32) with $\ell = 0$ and (33), respectively.

The results are presented in Table III. We see that for most values of θ, RMS outperforms the Monte Carlo, but the larger θ is, the lower the efficiency ratio $\mathrm{Eff}(\hat{\gamma})$ becomes. At the same time, as θ grows, the value of $\mathrm{RE}(\hat{T}_B)$ deviates more and more from the desired target $5 \cdot 10^{-3}$, as in (34). This indicates a violation of Assumptions (I–II). We note that the estimates $\hat{\gamma}$ are quite accurate nonetheless, with a minor relative error of a few percent visible for larger values of θ.

In Fig. 3, we plot five random recurrency cycles conditioned on reaching the rare set B. We see that the paths do not gradually drift towards B, but rather first move far away from B, due to the drift-induced rotation. This hints that the distance-based importance function might be a poor choice. Figure 4 shows that even property (26) seems to be violated. In this figure, we compare the histograms of S_{rec} and S^q_{rec} in order to compare the distributions of X_1^A and X_+^A

TABLE II. RMS algorithm for a 10-dim OU process. Parameters: Q is a matrix with only real eigenvalues, $A = \{x_1 \le 0\}$, $B = \{x_1 \ge u\}$; u has been chosen using (41) to match the values of γ in the first row. We have $\hat{\alpha}_A = 0.0124$, $\mathrm{RE}(\hat{\alpha}_A) = 2.46 \times 10^{-3}$.

$\gamma(u)$	1×10^{-03}	1×10^{-04}	1×10^{-05}	1×10^{-06}	1×10^{-07}
$\hat{\gamma}$	1.00×10^{-03}	9.95×10^{-05}	1.02×10^{-05}	9.92×10^{-07}	1.00×10^{-07}
$\mathrm{RE}(\hat{\gamma})$	7.84×10^{-03}	1.03×10^{-02}	1.35×10^{-02}	1.12×10^{-02}	1.49×10^{-02}
$\mathrm{Eff}(\hat{\gamma})$	0.8	2.4	9.3	34.9	180.5
$\mathrm{RE}(\hat{T}_B)$	7.87×10^{-03}	1.02×10^{-02}	1.35×10^{-02}	1.12×10^{-02}	1.49×10^{-02}

TABLE III. RMS algorithm applied to the 2-dim OU process. Parameters: $Q(\theta)$ as in (42), $A = \{x_1 \le 0\}$, $B = \{x_1 \ge u\}$; u has been chosen depending on θ such that in every case $\gamma(u) = 10^{-6}$.

θ	0.5	1	1.5	2	3
$\hat{\gamma}$	9.91×10^{-07}	1.00×10^{-06}	1.00×10^{-06}	9.73×10^{-07}	9.60×10^{-07}
$\mathrm{RE}(\hat{\gamma})$	8.20×10^{-03}	1.05×10^{-02}	2.34×10^{-02}	2.66×10^{-02}	4.01×10^{-02}
$\mathrm{Eff}(\hat{\gamma})$	31.9	27.9	7.1	5.8	1.0
$\mathrm{RE}(\hat{T}_B)$	7.63×10^{-03}	1.05×10^{-02}	2.37×10^{-02}	2.67×10^{-02}	4.01×10^{-02}

(see the discussion Sec. IV B). The figure shows that X_+^A has more probability mass in the sets $\{x_2 \le -1\}$ or $\{x_2 \ge 1\}$ than X_1^A.

Conclusions. When Q has non-real eigenvalues, the naïve choice of the recurrency set and the distance-based importance function [i.e., (32) and (33)] seems inadequate and leads to a relative error higher than expected. This underscores the fact that one has to be careful with the choice of A and $H(\cdot)$ and verify whether Assumptions (I–II) are satisfied; this can be done e.g., by the means described in Sec. IV B. Despite violation of Assumptions (I–II), RMS still gives rather accurate estimates of γ, and outperforms Monte Carlo for small θ.

C. Franzke (2012) stochastic climate model

As our final example, we consider the low-order stochastic climate model presented by Franzke (2012). This is a 4-dimensional SDE with certain key features that are also present in more complex climate models, including nonlinear (quadratic) drift terms that are energy-conserving. We refer to Franzke (2012) for a more detailed discussion of the physical interpretation of this model.

The model is given by the following set of SDEs. It uses a standard, two-dimensional Wiener process $(W_t^{(1)}, W_t^{(2)})$. We write $x_i := X_t^{(i)}$, $y_i := Y_t^{(i)}$ and $W_i := W_t^{(i)}$ to simplify notation. We consider the system

$$\begin{aligned}
\mathrm{d}x_1 &= \mu\big(- x_2(L_{12} + a_1 x_1 + a_2 x_2) + d_1 x_1 + F_1 \\
&\quad + F_1 + L_{13} y_1 + B^1_{123} x_2 y_1 + (B^2_{131} + B^2_{113}) x_1 y_1 \big)\mathrm{d}t, \\
\mathrm{d}x_2 &= \mu\big(+ x_1(L_{21} + a_1 x_1 + a_2 x_2) + d_2 x_2 + F_2 \\
&\quad + L_{24} y_2 + B^1_{213} x_1 y_1 + (B^3_{242} + B^3_{224}) x_2 y_2 \big)\mathrm{d}t, \\
\mathrm{d}y_1 &= \mu\big(- L_{13} x_1 + B^1_{312} x_1 x_2 + B^2_{311} x_1^2 + F_3 - \tfrac{\gamma_1}{\varepsilon} y_1 \big)\mathrm{d}t \\
&\quad + \tfrac{\sigma_1}{\sqrt{\varepsilon}} \mathrm{d}W_1, \\
\mathrm{d}y_2 &= \mu\big(- L_{24} x_2 + B^3_{422} x_2 x_2 + F_4 - \tfrac{\gamma_2}{\varepsilon} y_2 \big)\mathrm{d}t + \tfrac{\sigma_2}{\sqrt{\varepsilon}} \mathrm{d}W_2.
\end{aligned}$$

When the parameter ε is set to a small value, a separation of timescales is created between the variables x_1, x_2 (slow) and y_1, y_2 (fast). The main interest is in the behavior of the slow variables x_1, x_2.

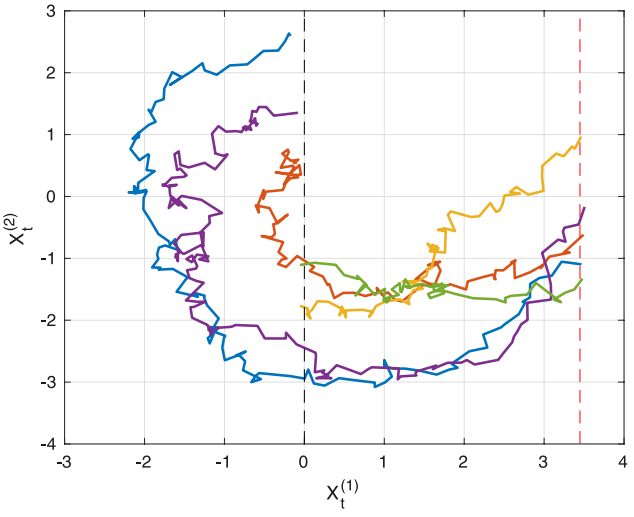

FIG. 3. 2-dim OU process. Five random realizations of recurrency cycles conditioned on reaching the rare set. The cycles have been plotted until the first hitting time of B. Parameters: $A = \{x_1 \le 0\}$, $\theta = 3$, $B = \{x_1 \ge u\}$ with $u \approx 3.4$ such that $\gamma(u) = 10^{-6}$.

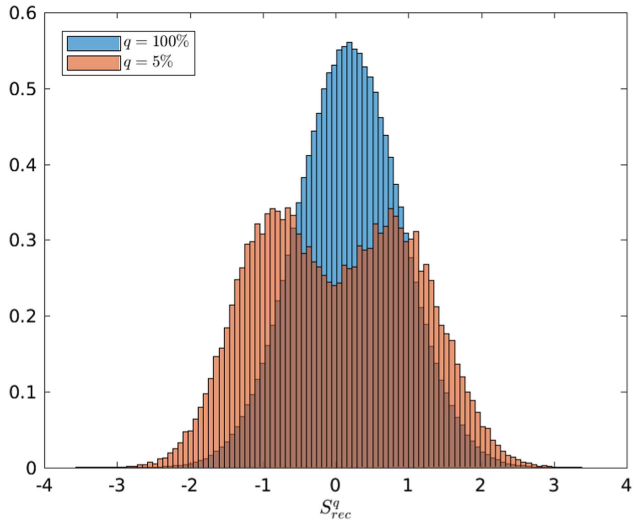

FIG. 4. 2-dim OU process, $\theta = 3$. Marginal histograms of S^q_{rec} projected onto the second dimension. The histograms have been normalized to a probability density function.

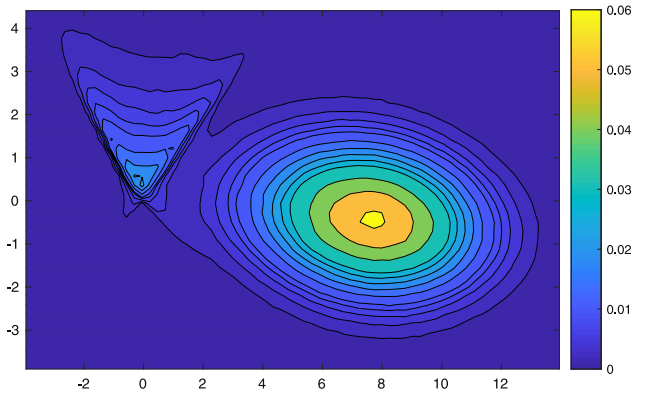

FIG. 5. Contour plot of the marginal stationary density of slow variates (x_1, x_2) of the model of Franzke (2012).

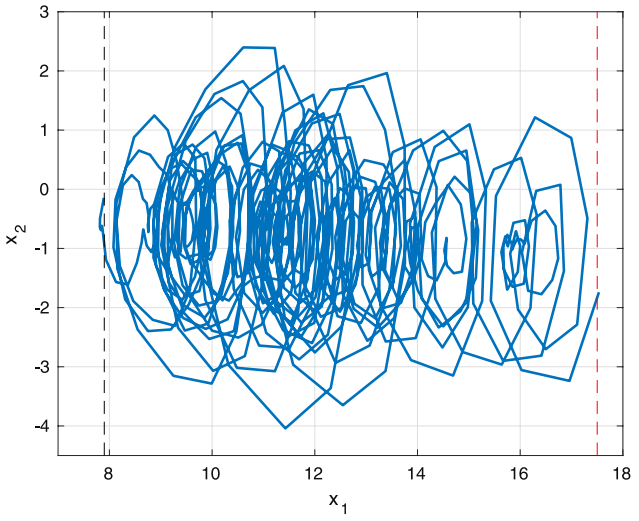

FIG. 6. The model of Franzke (2012). A random realization of a recurrency cycle conditioned on reaching the rare set. The cycle has been plotted until the first hitting time of B. Parameters: $A = \{x_1 \le 7.9\}$, $B = \{x_1 \ge 17.5\}$, $\gamma(17.5) \approx 1.14 \cdot 10^{-6}$.

The parameters we use match those used in Franzke (2012). This means that we set $\mu = 1$, the B-coefficients are given by $B^1_{123} = 4$, $B^1_{213} = 4$, $B^1_{312} = -8$, $B^2_{131} = 0.25$, $B^2_{113} = 0.25$, $B^2_{311} = -0.5$, $B^3_{242} = -0.3$, $B^3_{224} = -0.4$, $B^3_{422} = 0.7$, the L-coefficients by $L_{13} = -L_{24} = -0.2$, and the other parameters by $\omega = 1$, $a_1 = 1$, $a_2 = -1$, $d_1 = -0.2$, $d_2 = -0.1$, $\gamma_1 = \gamma_2 = 1$, $\sigma_1 = 3$, $\sigma_2 = 1$. In addition, we put $L_{12} = -L_{21} = 1, \varepsilon = 0.2$. The forcing vector (F_1, F_2, F_3, F_4) is given by $(-0.25, 0, 0, 0)$.

Since this process is non-standard, in order to build intuition, we first generated a contour plot of the estimated stationary density of (x_1, x_2); see Fig. 5. The process turns out to randomly switch between two modes: one mode with $x_1 \le x_2$ and a second mode with $x_1 \ge x_2$. The estimated density function in Fig. 5 shows that the process is more likely to be in the second mode.

We use the explicit Euler scheme with $h_0 = 10^{-4}$ but we store the values of the process every $h = 0.01$. The small integration time step h_0 is needed for numerical stability. Similar to the previous examples, the rare event we study is the exceedance of a high threshold by x_1 under the stationary distribution, cf. (31). We choose the recurrency set $A(\ell)$ as in (32) with $\ell = 7.9$, as suggested by the algorithm (27). The importance function $H(\cdot)$ is as in (33).

The results of the RMS method are outstanding, see Table IV. For $u = 18.5$, when $\gamma(u) \approx 10^{-7}$, we find $\mathrm{Eff}(\hat{\gamma}) \approx 1522$. In other words, the RMS algorithm is more than 1500 times faster than MC. The values of $\mathrm{RE}(\hat{T}_B)$ match the desired $5 \cdot 10^{-3}$ [see (34)] very closely even for very high thresholds, indicating that Assumptions (I–II) are satisfied. A random realization of a cycle reaching the rare event, shown in Fig. 6, is yet another indication that the distance-based importance function is a good choice, as the path seems to gradually drift towards the rare event.

Conclusions. This example shows a successful application of the RMS algorithm to a multidimensional nonlinear stochastic-dynamical model with characteristics of complex climate models. We find that RMS is up to three orders of magnitude faster than MC in this example, and the efficiency gain is expected to be even larger for higher thresholds u.

VI. SUMMARY

In this article, we have proposed a new algorithm for the estimation of small steady-state probabilities $\gamma = \mu(B)$, as in (1), of Markov processes with continuous state space. Our approach, which we have called the Recurrent Multilevel Splitting (RMS) algorithm, is based on the alternative representation (9) of γ (as given in Theorem 1). This representation is obtained by dissecting the path of the Markov process into recurrency cycles, each cycle beginning with an inwards

TABLE IV. RMS algorithm applied to the model of Franzke (2012). Parameters: $A = \{x_1 \le 7.9\}$, $B = \{x_1 > u\}$. We have $\hat{\alpha}_A = 0.0124$, $\mathrm{RE}(\hat{\alpha}_A) = 2.83 \times 10^{-3}$.

u	14	15	16	17.5	18.5
$\hat{\gamma}$	1.08×10^{-03}	1.99×10^{-04}	3.00×10^{-05}	1.14×10^{-06}	9.78×10^{-08}
$\mathrm{RE}(\hat{\gamma})$	6.1×10^{-03}	7.2×10^{-03}	7.4×10^{-03}	7.4×10^{-03}	5.8×10^{-03}
$\hat{\gamma}^{\mathrm{MC}}$	1.08×10^{-03}	2.00×10^{-04}	2.98×10^{-05}	1.12×10^{-06}	8.85×10^{-08}
$\mathrm{RE}(\hat{\gamma}^{\mathrm{MC}})$	1.4×10^{-03}	2.9×10^{-03}	6.5×10^{-03}	2.7×10^{-02}	8.5×10^{-02}
$\mathrm{Eff}(\hat{\gamma})$	1.9	8.6	32.1	269.9	1521.8
$\mathrm{RE}(\hat{T}_B)$	5.1×10^{-03}	6.4×10^{-03}	7.2×10^{-03}	6.6×10^{-03}	5.4×10^{-03}

crossing of a set A. It allows to transform the problem of estimating γ essentially into the problem of estimating T_B, the expected time spent in set B in a recurrency cycle.

In order to efficiently estimate T_B, we use Multilevel Splitting (MLS), but we emphasize that other rare event simulation methods could have been used instead (such as Genealogical Particle Analysis or Importance Sampling). We have derived optimal parameters for the MLS in Appendix B, and we have shown (Theorem 3) that under simplifying assumptions, a suitable choice of the recurrency set A in combination with the optimal choice of the parameters leads to logarithmic efficiency of the RMS algorithm.

In Sec. V, four numerical studies were presented, where we used the RMS algorithm to estimate steady state probabilities of high threshold exceedances for various SDEs discretized in time. The experiments demonstrate that RMS gives accurate results. Furthermore, they unanimously show the efficiency gain of RMS compared to Monte Carlo (MC); in the most notable case of the (Franzke, 2012) model (Sec. V C), RMS outperforms MC by up to three orders of magnitude.

One of the numerical experiments (Sec. V B 3) was designed to give suboptimal results, with an SDE displaying rotating motion so that the most straightforward choices of the recurrency set and importance function (as used in the experiments) were expected to be not very suitable. Although the estimates obtained with RMS were still quite accurate, the efficiency gain of RMS compared to MC was decreasing as the rotation speed was increasing. This example showed how the choice of the recurrency set and the importance function can impact the performance of the algorithm.

In light of this example, an interesting topic for future research is the choice of the recurrency set A. As already mentioned in Sec. IV B, a good choice of A should be a suitable compromise between visiting A relatively often and (26) being (approximately) met. We have proposed a method of optimizing $A(\ell)$ parametrized by ℓ in (27), and pointed out a method of testing whether A satisfies (26) through a quantile validation (28). Further development of these ideas to construct an optimal A is a challenging open research topic.

ACKNOWLEDGMENTS

We thank the organizers of the *Summer School in Monte Carlo for Rare Events* (June 2016 at Brown University) for making lecture notes available. This work is part of the research programme "Mathematics of Planet Earth" which was funded by the Netherlands Organisation for Scientific Research (NWO) (Grant No. 657.014.003). Michel Mandjes' research is partly funded by the NWO Gravitation Programme NETWORKS (Grant No. 024.002.003).

APPENDIX A: TECHNICAL RESULTS

Proof of Theorem 1. Define a new Markov chain $Z_n := (X_{n-1}, X_n)$; it is also positive Harris with a stationary measure $\tilde{\mu}$ satisfying for measurable sets C_0, C_1,

$$\tilde{\mu}(C_0, C_1) = \mathbb{P}(X_0 \in C_0, X_1 \in C_1 \mid X_0 \sim \mu).$$

We see that the stopping times S_n coincide with the times the process Z_n visits a set $\mathcal{A} := (A^c, A)$, with $A^c := \mathbb{R}^d \setminus A$. Since $\mu(A) \in (0,1)$, we have

$$\alpha_A = \mathbb{P}_\mu(X_0 \in A, X_1 \in A^c) > 0.$$

According to Meyn and Tweedie (2012, Theorem 10.4.9), we have, with $\tau_{\mathcal{A}} := \inf\{n > 0 : Z_n \in \mathcal{A}\}$,

$$\tilde{\mu}(\mathbb{R}^d, B) = \int_{\mathcal{A}} \tilde{\mu}(\mathrm{d}x, \mathrm{d}y)\, \mathbb{E}_x \sum_{n=0}^{\tau_{\mathcal{A}}-1} \mathbb{1}\{Z_n \in (\mathbb{R}^d, B)\}.$$

Due to $\tilde{\mu}(\mathbb{R}^d, B) = \mu(B)$, $\{Z_n \in (\mathbb{R}^d, B)\} = \{X_n \in B\}$, and $\tilde{\mu}(\mathcal{A}) = \alpha_A$, it follows that

$$\mu(B) = \alpha_A \cdot \mathbb{E}\Big(\sum_{n=0}^{\tau_{\mathcal{A}}-1} \mathbb{1}\{X_n \in B\} \mid Z_0 \sim \tilde{\mu}, Z_0 \in \mathcal{A} \Big).$$

Finally, we recognize that the conditioning above is equivalent to X_0 being distributed as an initial point of a recurrency cycle X_1^A in stationarity so that we conclude (9). Similarly, one can show that $\alpha_A = (\mathbb{E}_\mu L_1)^{-1}$ by considering the expected time spent in $(\mathbb{R}^d, \mathbb{R}^d)$ within a recurrency cycle. □

Derivation of (24). Note that (I) implies the number of times the k-th threshold r_k is hit and is distributed as a sum of $n_{k-1}\, r_{m-1}$ *independent* Bernoulli trials, each with probability of success p_k

$$(r_k \mid r_{k-1}) \overset{\mathrm{d}}{=} \mathrm{Bin}(n_{k-1} r_{k-1}, p_k). \tag{A1}$$

Here, $\mathrm{Bin}(n,p)$ denotes a Binomial distribution with n trials with success probability p, with the convention that $\mathrm{Bin}(0,p) \equiv 0$. Similarly, (II) implies that the total time spent in the rare set is distributed as a sum of $n_m r_m$ *independent* copies from the distribution R_+

$$(r_{m+1} \mid r_m) \overset{\mathrm{d}}{=} \sum_{k=1}^{n_m r_m} R_+^{(k)}, \tag{A2}$$

where $R_+^{(1)}, R_+^{(2)}, \ldots$ are i.i.d. copies of R_+ (with the empty sum being defined as 0). Using (A1) and the law of total variance we obtain, for $k \in \{1, \ldots m\}$,

$$\begin{aligned}
\mathbb{V}\mathrm{ar}(r_k) &= \mathbb{E}[\mathbb{V}\mathrm{ar}(r_k | r_{k-1})] + \mathbb{V}\mathrm{ar}[\mathbb{E}(r_k | r_{k-1})] \\
&= \mathbb{E}[n_{k-1} r_{k-1} p_k (1-p_k)] + \mathbb{V}\mathrm{ar}(n_{k-1} r_{k-1} p_k) \\
&= n_{k-1} p_k (1-p_k) \mathbb{E}(r_{k-1}) + n_{k-1}^2 p_k^2 \mathbb{V}\mathrm{ar}(r_{k-1}).
\end{aligned}$$

Similarly, using (A2), we obtain

$$\mathbb{V}\mathrm{ar}(r_{m+1}) = n_m \mathbb{E}(r_m) \mathbb{V}\mathrm{ar}(R_+) + n_m^2 (\mathbb{E}R_+)^2 \mathbb{V}\mathrm{ar}(r_m).$$

Combining these results with (22) yields (24). □

APPENDIX B: DERIVATION OF OPTIMAL PARAMETERS

Following Amrein and Künsch (2011), we assume that the computational effort w_k in the k-th stage of Algorithm 1 (to sample a path starting from X_{τ_k} until $\min\{\tau_{k+1}, \tau_A^{\mathrm{in}}\}$) does not depend on the entry state X_{τ_k}. Simplifying this further, we assume that w_k does not depend on k, so without loss of generality,

$$w_k \equiv 1, \quad k \in \{0, \ldots, m\}. \tag{B1}$$

A more general cost w_k can be considered for particular problems, see, e.g., Lagnoux (2006).

Let $N_k := n_k r_k$, for $k \in \{0, \dots, m\}$ be the number of paths simulated in the k-th stage of the algorithm, with $r_0 := 1$. Then, the average total workload equals

$$W := \sum_{k=0}^{m} \mathbb{E}N_k \tag{B2}$$

and since $\mathbb{E}r_k = p_1 \cdots p_k$, cf. (22), we conclude

$$\mathbb{E}N_k = \prod_{j=0}^{k} n_j p_j.$$

Finally, we formulate the minimization problem

$$\begin{aligned} &\text{minimize: } W := \textstyle\sum_{k=0}^{m} \prod_{j=0}^{k} n_j p_j \\ &\text{with respect to: } m, p_1, \dots, p_m, n_0, \dots, n_m \\ &\text{subject to: } \begin{cases} \mathrm{RE}^2(\hat{T}_B) \le \rho, \\ \prod_{k=1}^{m} p_k = p_B, \\ m \in \mathbb{N}, \\ p_k \in (0,1),\ k \in \{1, \dots, m\}, \\ n_k \in \mathbb{N},\ k \in \{0, \dots, m\}. \end{cases} \end{aligned}$$

In our simplified setting, i.e., under Assumptions (I–II), we have derived a formula for the corresponding squared relative error in (24). We are able to solve the optimization problem above under the additional relaxation that n_k and m are real and positive. To this end, it is helpful to denote

$$\begin{aligned} c_k &:= \textstyle\prod_{j=0}^{k-1} n_j p_j, \quad k \in \{1, \dots, m+1\}, \\ a_k &:= (1 - p_k)/p_k, \quad k \in \{1, \dots m\}, \\ a_{m+1} &:= \mathrm{RE}^2(R_+). \end{aligned}$$

Then, we can write

$$W = \sum_{k=1}^{m+1} c_k \quad \text{and} \quad \mathrm{RE}^2(\hat{T}_B) = \sum_{k=1}^{m+1} \frac{a_k}{c_k}.$$

We want to minimize the workload W under the constraint that

$$\mathrm{RE}^2(\hat{T}_B) \le \rho.$$

We do this in steps. First, we fix m and the conditional probabilities $p_1, \dots, p_m$ so that $a_1, \dots, a_m$ are fixed (recall that a_{m+1} is not a parameter of the algorithm). We relax the problem and let the splitting factors n_k be allowed to attain any real, positive value. This means that we wish to solve (over $c_1, \dots, c_{m+1} > 0$)

$$\begin{aligned} &\text{minimize: } W(c_1, \dots, c_{m+1}) := \textstyle\sum_{k=1}^{m+1} c_k \\ &\text{subject to: } \begin{cases} g(c_1, \dots, c_{m+1}) := \sum_{k=1}^{m+1} \frac{a_k}{c_k} \le \rho, \\ c_k > 0,\ k \in \{1, \dots, m+1\}. \end{cases} \end{aligned}$$

The corresponding Karush–Kuhn–Tucker conditions are

$$\begin{cases} \nabla W + \mu \nabla g = 0, \\ \mu(g - \rho) = 0, \\ \mu \in [0, \infty), \end{cases}$$

with the gradient "∇" taken with respect to vector $(c_1, \dots, c_{m+1})$. These are solved by

$$c_k := \frac{1}{\rho} \sqrt{a_k} \sum_{j=1}^{m+1} \sqrt{a_j},$$

with the optimal workload

$$W = \frac{1}{\rho} \left(\sum_{j=1}^{m+1} \sqrt{a_j} \right)^2.$$

In the next step, we keep m fixed and minimize over $a_1, \dots, a_m$. Notice that $1 + a_k = 1/p_k$ so that our minimization problem takes the form

$$\begin{aligned} &\text{minimize: } W(a_1, \dots, a_m) := \tfrac{1}{\rho} \left(\textstyle\sum_{k=1}^{m+1} \sqrt{a_k} \right)^2 \\ &\text{subject to: } \begin{cases} h(a_1, \dots, a_m) := \prod_{k=1}^{m} (1 + a_k) = p_B^{-1}, \\ a_k > 0, \quad k \in \{1, \dots, m\}. \end{cases} \end{aligned}$$

Not surprisingly, this system is solved by

$$a_1 = \cdots = a_m = p_B^{-1/m} - 1$$

so that the optimal intermediate probabilities coincide

$$p_k = p_B^{1/m}, \quad k \in \{1, \dots, m\},$$

with the optimal workload being

$$W(m) = \frac{1}{\rho} \left(m \sqrt{p_B^{-1/m} - 1} + \sqrt{a_{m+1}} \right)^2.$$

The final step is finding the optimal number of thresholds m. We see that the minimizer of $W(m)$ is also a minimizer of

$$m \sqrt{\exp(-\log(p_B)/m) - 1}. \tag{B3}$$

Again, we relax this problem, allowing m to be any real, positive number. Finally, the optimal parameters are

$$\begin{aligned} m &= c\,|\log p_B|, \\ p_k &= p_{\mathrm{opt}} := \frac{2c - 1}{2c} \approx 0.2032, \quad k \in \{1, \dots, m\}, \\ n_0 &= \frac{1}{\rho \sqrt{2c - 1}} \cdot \left(\frac{c\,|\log p_B|}{\sqrt{2c - 1}} + \mathrm{RE}(R_+) \right), \\ n_k &= 1/p_{k+1} = 1/p_{\mathrm{opt}}, \quad k \in \{1, \dots, m-1\}, \\ n_m &= \mathrm{RE}(R_+) \cdot \frac{2c}{\sqrt{2c - 1}}, \end{aligned} \tag{B4}$$

with $c \approx 0.6275$ solving $\exp(1/c) = 2c/(2c - 1)$ and the optimal workload reads as in (25). Since m, n_k must be integers, we propose to simply round the optimal parameters to the closest integer. A similar result (but without the last splitting stage, in which we estimate the time spent in the set B) has been presented in (Lagnoux 2006, Example 3.2).

APPENDIX C: LOGARITHMIC EFFICIENCY OF THE RMS ALGORITHM

In this section, we study the efficiency of the RMS method in the asymptotic regime that the rare event probability (1) tends to 0 (i.e., $\gamma \to 0$). First, we note that if we fix the recurrency set A, then α_A does not change as $\gamma \to 0$; hence we only have that $T_B \to 0$. This indicates that asymptotic efficiency properties of RMS will be closely related to those of MLS. In order to study the performance of the estimator, we first introduce the concepts of *strong* and *logarithmic efficiency*.

Let $\hat{\Psi}_\ell$ be a family of unbiased estimators for $\Psi_\ell > 0$, parametrized by ℓ such that $\Psi_\ell \to 0$, as $\ell \to \infty$. Let $W(\hat{\Psi}_\ell)$ denote the computation time corresponding to $\hat{\Psi}_\ell$. The estimator Ψ_ℓ is called *strongly efficient* if

$$\limsup_{\ell\to\infty} \frac{W(\hat{\Psi}_\ell)\cdot \mathbb{V}\mathrm{ar}(\hat{\Psi}_\ell)}{\Psi_\ell^2} < \infty \tag{C1}$$

and *logarithmically efficient* if

$$\lim_{\ell\to\infty} \frac{W(\hat{\Psi}_\ell)\cdot \mathbb{V}\mathrm{ar}(\hat{\Psi}_\ell)}{\Psi_\ell^{2-\varepsilon}} = 0, \quad \text{for all } \varepsilon > 0. \tag{C2}$$

Strong efficiency implies that the workload needed to estimate the quantity of interest Φ_ℓ with a desired accuracy $\mathrm{RE}^2(\Psi_\ell) \le \rho$ is uniformly bounded as $\ell \to \infty$. Logarithmic efficiency implies that workload needed to achieve the accuracy $\mathrm{RE}^2(\Psi_\ell) = \rho$ is increasing slower than $\Psi_\ell^{-\varepsilon}$ for any $\varepsilon > 0$, as $\ell \to \infty$. Evidently, strong efficiency implies logarithmic efficiency.

Before we prove the logarithmic efficiency of RMS in Theorem 3, we show an inefficiency result for the Monte Carlo estimator for T_B. Let $\hat{T}_B^{\mathrm{MC}}$ be a sample mean of N independent copies of R_1. We then have

$$\mathrm{RE}^2(\hat{T}_B^{\mathrm{MC}}) = \frac{1 - p_B + \mathrm{RE}^2(R_+)}{p_B N}. \tag{C3}$$

Now to achieve a desired level of accuracy $\mathrm{RE}^2(\hat{T}_B^{\mathrm{MC}}) \le \rho$, assuming (B1), the total required workload is

$$W(\hat{T}_B^{\mathrm{MC}}) := \frac{1}{q}\cdot\frac{1 - p_B + \mathrm{RE}^2(R_+)}{p_B}. \tag{C4}$$

As already noted in Sec. IV A, $W(\hat{T}_B^{\mathrm{MC}})$ is inversely proportional to p_B and so it follows that the Monte Carlo estimator is not logarithmically efficient.

We have seen, cf. (25), that the workload of the MLS estimator with the optimal parameters $W(\hat{T}_B)$ is proportional to $[\log(p_B)]^2$. It turns out that under mild additional assumption, the MLS algorithm is logarithmically efficient and thus so is RMS. We make this rigorous in the following theorem.

Theorem C.3 (Logarithmic Efficiency of RMS). *Fix the recurrency set A and let the set B_ℓ be parametrized by ℓ, such that $\gamma_\ell := \mu(B_\ell) \to 0$ as $\ell \to \infty$. Assume*

- *that the estimators $\hat{\alpha}_A$ and $\hat{T}_{B_\ell}$ are independent;*
- *that Assumptions (I–II) are valid for each ℓ;*
- *that the workload satisfies (B1);*
- *and that, for $\delta > 0$ sufficiently small,*

$$\limsup_{\ell\to\infty} \frac{\mathbb{V}ar(R_+)}{(\mathbb{E}R_+)^2} < \infty, \quad \lim_{\ell\to\infty} T_{B_\ell}\cdot p_{B_\ell}^{-\delta} = 0. \tag{C5}$$

Then, the RMS estimator $\hat{\gamma}_\ell$ for γ_ℓ, with the optimal choice of the parameters (B4), is logarithmically efficient.

We point out that the first part of the assumption (C5) is equivalent to strong efficiency of the crude Monte Carlo estimator for R_+, under the workload assumption (B1). This is not too restrictive, as often the main difficulty when estimating T_B lies in the fact that p_B is extremely small (and does not relate to the large variance of R_+.) Since $\gamma_\ell \to 0$ and A is fixed then necessarily $T_{B_\ell} \to 0$. In the second part of (C5), we require that there exists a $\delta > 0$ such that $\mathbb{E}R_+ p_{B_\ell}^{1-\delta} \to 0$. Loosely speaking, it means that p_{B_ℓ} converges to 0 at least polynomially faster than $\mathbb{E}R_+$ grows to infinity; this is trivially satisfied when $\mathbb{E}R_+$ is bounded.

Proof of Theorem 3. Since the recurrency set A is fixed, the quantities $\hat{\alpha}_A$, $\mathrm{RE}(\hat{\alpha}_A)$, and $W(\hat{\alpha}_A)$ do not depend on ℓ. In addition, $\alpha_A \cdot T_{B_\ell} = \mu(B_\ell) \to 0$ is equivalent to $T_{B_\ell} \to 0$. Moreover, since $T_{B_\ell} = p_{B_\ell}\cdot \mathbb{E}R_+$, cf. (16), and $\mathbb{E}R_+ \ge 1$, we necessarily have $p_{B_\ell} \to 0$, as ℓ grows. Observe that

$$\begin{aligned}\frac{W(\hat{\gamma}_\ell)\mathbb{V}\mathrm{ar}(\hat{\gamma}_\ell)}{\gamma_\ell^{2-\varepsilon}} &= \frac{W(\hat{\alpha}_A) + W(\hat{T}_{B_\ell})}{\gamma_\ell^{-\varepsilon}}\cdot\frac{\mathbb{V}\mathrm{ar}(\hat{\alpha}_A\cdot\hat{T}_{B_\ell})}{\alpha_A^2\cdot T_{B_\ell}^2}\\ &= \gamma_\ell^{\varepsilon}\big[W(\hat{\alpha}_A) + W(\hat{T}_{B_\ell})\big]\cdot\big[\mathrm{RE}(\hat{\alpha}_A) + \mathrm{RE}(\hat{T}_{B_\ell})\big].\end{aligned} \tag{C6}$$

We put $\mathrm{RE}(\hat{T}_{B_\ell}) = q$. Then, the workload $W(\hat{T}_{B_\ell})$ is given as in (25), and we see that

$$\begin{aligned}\gamma_\ell^{\varepsilon} W(\hat{T}_{B_\ell}) &= \alpha_A^{\varepsilon} T_{B_\ell}^{\varepsilon}\cdot\frac{1}{q}\left(\frac{c\,|\log p_{B_\ell}|}{\sqrt{2c-1}} + \mathrm{RE}(R_+)\right)^2\\ &\sim \frac{c^2\alpha_A^{\varepsilon}(T_{B_\ell}p_{B_\ell}^{-\delta})^{\varepsilon}}{q(2c-1)}\cdot p_{B_\ell}^{\delta\varepsilon}(\log p_{B_\ell})^2,\end{aligned}$$

where $\delta > 0$ is as in (C5). Now since $p_{B_\ell} \to 0$, we also have

$$p_{B_\ell}^{\delta\varepsilon}(\log p_{B_\ell})^2 \to 0$$

and $\gamma_\ell^{\varepsilon} W(\hat{T}_{B_\ell}) \to 0$, which applied to (C6) finishes the proof. □

REFERENCES

Amrein, M. and Künsch, H. R., "A variant of importance splitting for rare event estimation: Fixed number of successes," ACM Trans. Model. Comput. Simul. (TOMACS) **21**(2), 13 (2011).

Asmussen, S., *Applied Probability and Queues* (Springer Science & Business Media, 2008), Vol. 51.

Asmussen, S. and Glynn, P. W., *Stochastic Simulation: Algorithms and Analysis* (Springer Science & Business Media, 2007), Vol. 57.

Bisewski, K., Crommelin, D., and Mandjes, M., "Simulation-based assessment of the stationary tail distribution of a stochastic differential equation," in *Proceedings of the 2018 Winter Simulation Conference* (IEEE, 2018), pp. 1742–1753.

Calvin, J. M., Glynn, P. W., and Nakayama, M. K., "The semi-regenerative method of simulation output analysis," ACM Trans. Model. Comput. Simul. (TOMACS) **16**(3), 280–315 (2006).

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

Coles, S., Bawa, J., Trenner, L., and Dorazio, P., *An Introduction to Statistical Modeling of Extreme Values* (Springer, 2001), Vol. 208.
Crane, M. A. and Iglehart, D. L., "Simulating stable stochastic systems: III. Regenerative processes and discrete-event simulations," Oper. Res. **23**(1), 33–45 (1975).
Dean, T. and Dupuis, P., "Splitting for rare event simulation: A large deviation approach to design and analysis," Stoch. Process. Their Appl. **119**(2), 562–587 (2009).
Del Moral, P. and Garnier, J., "Genealogical particle analysis of rare events," Ann. Appl. Probab. **15**(4), 2496–2534 (2005).
Franzke, C., "Predictability of extreme events in a nonlinear stochastic-dynamical model," Phys. Rev. E **85**(3), 031134 (2012).
Garvels, M. J. J., "The splitting method in rare event estimation," Ph.D. thesis (University of Twente, Twente, The Netherlands, 2000), see http://doc.utwente.nl/29637/1/t0000013.pdf.
Goyal, A., Shahabuddin, P., Heidelberger, P., Nicola, V. F., and Glynn, P. W., "A unified framework for simulating Markovian models of highly dependable systems," IEEE Trans. Comput. **41**(1), 36–51 (1992).
Gunther, F. and Wolff, R., "The almost regenerative method for stochastic system simulations," Oper. Res. **28**(2), 375–386 (1980).
Heidelberger, P., "Fast simulation of rare events in queueing and reliability models," ACM Trans. Model. Comput. Simul. (TOMACS) **5**(1), 43–85 (1995).
Henderson, S. G. and Glynn, P. W., "Can the regenerative method be applied to discrete-event simulation?" in *Proceedings of the 31st Winter Simulation Conference* (IEEE, 1999), pp. 367–373.
Henderson, S. G. and Glynn, P. W., "Regenerative steady-state simulation of discrete-event systems," ACM Trans. Model. Comput. Simul. (TOMACS) **11**(4), 313–345 (2001).
Kalashnikov, V. V., *Topics on Regenerative Processes* (CRC Press, 1994).
Kloeden, P. E. and Platen, E., *Numerical Solution of Stochastic Differential Equations* (Springer, 1992).
Kroese, D. P., Taimre, T., and Botev, Z. I., *Handbook of Monte Carlo Methods* (John Wiley & Sons, 2013), Vol. 706.
Lagnoux, A., "Rare event simulation," Probab. Eng. Inform. Sci. **20**(1), 45–66 (2006).
Meyn, S. P. and Tweedie, R. L., *Markov Chains and Stochastic Stability* (Springer, 2012).
Ragone, F., Wouters, J., and Bouchet, F., "Computation of extreme heat waves in climate models using a large deviation algorithm," Proc. Natl. Acad. Sci. U.S.A. **115**(1), 24–29 (2018).
Rubino, G. and Tuffin, B., *Rare Event Simulation Using Monte Carlo Methods* (John Wiley & Sons, 2009).
Schurz, H., "The invariance of asymptotic laws of stochastic systems under discretization," Z. Angew. Math. Mech. **79**(6), 375–382 (1999).
Villén-Altamirano, M. and Villén-Altamirano, J., "The rare event simulation method restart: efficiency analysis and guidelines for its application," in *Network Performance Engineering* (Springer, 2011), pp. 509–547.
Wadman, W., Crommelin, D., and Frank, J., "A separated splitting technique for disconnected rare event sets," in *Proceedings of the 46th Winter Simulation Conference* (IEEE, 2014), pp. 522–532.