On logarithmically optimal exact simulation of max-stable and related random fields on a compact set

Liu, Zhipeng; Blanchet, Jose H.; Dieker, A.b.; Mikosch, Thomas

Published in:
Bernoulli

DOI:
10.3150/18-BEJ1076

Publication date:
2019

Document version
Peer reviewed version

Bernoulli **25**(4A), 2019, 2949–2981
https://doi.org/10.3150/18-BEJ1076

On logarithmically optimal exact simulation of max-stable and related random fields on a compact set

ZHIPENG LIU[1,*], JOSE H. BLANCHET[2], A.B. DIEKER[1,**] and THOMAS MIKOSCH[3]

[1]*Industrial Engineering and Operations Research Department, School of Engineering and Applied Sciences, Columbia University, 500 West 120th Street, New York, NY 10027, U.S.A.*
*E-mail: *zl2337@columbia.edu; **ton.dieker@ieor.columbia.edu*
[2]*Department of Management Science and Engineering, School of Engineering, Stanford University, Huang Engineering Center, 475 Via Ortega Avenue, Stanford, CA 94305-6015, U.S.A.*
E-mail: jose.blanchet@stanford.edu
[3]*Department of Mathematics, University of Copenhagen, Universitetsparken 5, DK-2100 Copenhagen, Denmark. E-mail: mikosch@math.ku.dk*

We consider the random field

$$M(t) = \sup_{n\geq 1}\{-\log A_n + X_n(t)\}, \qquad t \in T,$$

for a set $T \subset \mathbb{R}^m$, where (X_n) is an i.i.d. sequence of centered Gaussian random fields on T and $0 < A_1 < A_2 < \cdots$ are the arrivals of a general renewal process on $(0, \infty)$, independent of (X_n). In particular, a large class of max-stable random fields with Gumbel marginals have such a representation. Assume that one needs $c(d) = c(\{t_1, \ldots, t_d\})$ function evaluations to sample X_n at d locations $t_1, \ldots, t_d \in T$. We provide an algorithm which samples $M(t_1), \ldots, M(t_d)$ with complexity $O(c(d)^{1+o(1)})$ as measured in the L_p norm sense for any $p \geq 1$. Moreover, if X_n has an a.s. converging series representation, then M can be a.s. approximated with error δ uniformly over T and with complexity $O(1/(\delta \log(1/\delta))^{1/\alpha})$, where α relates to the Hölder continuity exponent of the process X_n (so, if X_n is Brownian motion, $\alpha = 1/2$).

Keywords: Brown–Resnick process; exact simulation; Gaussian field; max-stable random fields; record-breaking

1. Introduction

Let X be a centered Gaussian random field on a set $T \subseteq \mathbb{R}^m$, $m \geq 1$ and consider a sequence (X_n) of independent and identically distributed copies of X. In addition, let (A_n) be a renewal sequence independent of (X_n). Under mild regularity conditions on the X, we will provide an efficient Monte-Carlo algorithm for sampling the field

$$M(t) = \sup_{n\geq 1}\{-\log A_n + X_n(t) + \mu(t)\}, \qquad t \in T, \tag{1.1}$$

where $\mu : T \longrightarrow \mathbb{R}$ is a bounded function.

1350-7265

We will design and analyze an algorithm for the exact simulation of

$$M(t_1), \ldots, M(t_d) \qquad \text{for any choice of distinct locations } t_1, \ldots, t_d \in T,$$

and we will show that, in some sense, this algorithm is asymptotically optimal as $d \to \infty$.

The algorithm proposed here shaves off a factor of order (nearly) d from the running time of any of the existing exact sampling procedures. In particular, we will show that, under mild boundedness assumptions on X, it is as hard to sample $(M(t_i))_{i=1,\ldots,d}$ as it is to sample $(X(t_i))_{i=1,\ldots,d}$. Therefore, at least from a simulation point of view, it is not more difficult to work with M than with X. More precisely, if it takes $O(c(d))$ units of computing time to sample X at d distinct locations $t_1, \ldots, t_d \in T$, then it takes $O(c(d)^{1+o(1)})$ units to sample M at the same locations; see Theorem 2.2 for a precise formulation.

We illustrate this result by considering fractional Brownian motion X on $T = [0, 1]$. Using the circulant-embedding method (see [4], Section XI.3), we have $c(d) = O(d \log d)$ provided we sample at the dyadic points $t_i = i/2^{-m}$ for $i = 1, \ldots, 2^m = d$ (which we call dyadic points at level d). In the case of Brownian motion, one even has $c(d) = O(d)$, corresponding to the simulation of d independent Gaussian random variables. Thus, in the case of fractional Brownian motion on [0, 1] we provide an algorithm for sampling M at the dyadic points at level d in [0,1] with complexity $o(d^{1+\epsilon})$ for any $\epsilon > 0$; see [4], Sec. XI.6.

Moreover, if X has a series representation a.s. converging uniformly on T (such as the Lévy–Ciesielski representation for Brownian motion, see [21], Section 3.1), we also propose an approximate simulation procedure for M with a user-defined (deterministic) bound on the error which holds with probability one uniformly throughout T. More precisely, for any $\delta > 0$, the procedure that we present outputs an approximation M_δ to M such that

$$\sup_{t \in T} \big| M(t) - M_\delta(t) \big| \le \delta \qquad \text{a.s.} \tag{1.2}$$

The results concerning (1.2) are reported in Theorem 7.4. The method of designing a family $(M_\delta)_{\delta > 0}$ such that (1.2) holds is known as *Tolerance Enforced Simulation* (TES) or *δ-strong simulation*; see [7] and [20] for details. Note that a TES algorithm enforces a strong (almost sure) guarantee without knowledge of any specific set of sampling locations. This is a feature which distinguishes TES from more traditional algorithms in the broad literature on simulation of random fields and processes.

As will be explained later, the evaluation of $M_\delta(t)$ for fixed t takes $O(1)$ units of computing time while the construction of the process M_δ will often take $O(1/(\delta \log(1/\delta))^2)$ units. The latter result holds under assumptions on the convergence of the series representation of X which, in particular, are satisfied for Brownian motion X. In the latter case, the proposed procedure achieves a complexity of order $O(d)$ for the exact sampling of M on the dyadic points at level d (because the series truncated at level d is exact on the dyadic points at level d). Therefore, the exact sampling procedure based on Theorem 7.4 applied to the dyadic points at level d is optimal because it takes $O(d)$ computational cost to sample X at d dyadic points. Moreover, the convergence rate of the TES algorithm is also optimal in the Brownian case. In order to obtain a uniform error of order $O(\delta)$, one requires to discretize Brownian motion using a grid of size $O(1/(\delta \log(1/\delta))^2)$; see [3].

Our results are mainly motivated by application to the simulation of max-stable random fields. Indeed, if (A_i) is the arrival sequence of a unit rate Poisson process on $(0, \infty)$, M is a *max-stable process* in the sense of de Haan [11]. This means, in particular, that the distribution of $M(t)$ for any fixed $t \in T$ has a Gumbel distribution which is one of the max-stable distributions. The latter class of distributions consists of the non-degenerate limit distributions for the suitably centered and scaled partial maxima of an iid sequence; see for example [15]. The non-Gumbel max-stable processes with Fréchet or Weibull marginals are obtained from the representation (1.1) by suitable monotone transformations. We also mention that de Haan [11] already proved that max-stable processes with Gumbel marginals have representation (1.1), where X may have a rather general dependence structure not restricted to Gaussian X. However, the case of Gaussian X has attracted major attention. The case of Brownian X was treated in [10]. In the paper [17], the case of a general Gaussian process X with stationary increments was treated, known as the *Brown–Resnick process*, including the case of a Gaussian process X defined on a multidimensional set T often referred to as *Smith model*. It is used in environmental applications for modeling storm profiles; see, for example, [23]. General characterizations, including spectral representations and further properties, have been obtained as well; for example, see [22] and [17]. However, the explicit joint distribution of the max-stable process is in general not tractable. Because max-stable processes are generated as weak limits of maxima of i.i.d. random fields, max-stable models are particularly suited for modeling extremal events in spatio-temporal contexts. These include a wide range of applications of environmental type, for example, extreme rainfall [12] and extreme temperature [25].

Recently, several exact sampling procedures for M have been proposed and studied in the literature. In [13], an elegant and easy-to-implement procedure was proposed for the case in which X has stationary increments. Such a procedure has a computational complexity at least of order $O(c(d)d)$; see Proposition 4 in [14]. So, for example, if X is fractional Brownian motion, the procedure takes at least $O(d^2 \log d)$ units of computing time to produce d dyadic points of M in $[0, 1]$.

Another exact simulation method for M was recently proposed in [14]. It also has complexity $O(c(d)d)$ (see Proposition 4 in [14]), thus the procedure in [14] takes $O(d^2)$ for fractional Brownian motion (neglecting the contribution of logarithmic factors). This method is based on the idea of simulating the extremal functions. It is completely different from the approach taken here. Additional work concentrates on max-stable processes which satisfy special characteristics. For example, [22] proposed an exact simulation algorithm for the moving maxima model under suitable uniformity conditions.

Another recent development is [19], where the authors discuss an exact sampling algorithm for max-stable fields using the so-called normalized spectral representation. If the normalized spectral functions can be sampled with cost $c_{\mathrm{NS}}(d)$, then the algorithm in [19] samples the max-stable field exactly with complexity $O(c_{\mathrm{NS}}(d))$. However, for Gaussian-based max-stable fields, it is an open problem to devise exact sampling algorithms for the normalized spectral function, and it is unclear how $c_{\mathrm{NS}}(d)$ compares with the complexity $c(d)$ of sampling X.

An important difference between our method and those in [13] and [14] is the following: Both [13] and [14] take advantage of representations or structures which allow to truncate the infinite max-convolution in (1.1) while preserving the simple Gaussian structure of the number of terms in the truncation. Because the simple structure of these terms is preserved, the number of terms

in the truncation increases at least linearly in d. In contrast, we are able to truncate the number of terms in the infinite max-convolution uniformly in d. While the terms in the truncation have a slightly more complex structure (they are no longer i.i.d. Gaussian), they are still quite tractable from a simulation standpoint.

This paper is organized as follows: In Section 2, we present our main result and in Section 3 we discuss our general strategy, based on *milestone events* or *record-breakers.* The record-breaking strategy is illustrated in Section 4 in the setting of random walks, which is needed in our context due to the presence of (A_n) in M. Then we apply the record-breaking strategy to the setting of maxima of Gaussian random vectors with focus on Section 5: This section describes the main algorithmic developments of the paper. A complexity analysis is performed in Section 6. We introduce and analyze a TES algorithm in Section 7. Finally, in Section 8, we conclude our paper with a series of empirical comparison results.

2. Main result

This section provides a formal statement of the main result and its underlying assumptions. We assume that $(A_n)_{n\geq 0}$ is a renewal sequence, as mentioned in the Introduction. In particular, $A_0 = 0$, and $A_n = \tau_1 + \cdots + \tau_n$, $n \geq 1$, where (τ_i) is an i.i.d. sequence of positive random variables, independent of (X_n).

We introduce the following technical assumptions applicable to (A_n):

(A1) For any $\gamma < \mathbb{E}\tau_1$, there exists some $\theta_\gamma > 0$ such that $\mathbb{E}[\exp(\theta_\gamma(\gamma - \tau_1))] = 1$.

(A2) It is possible to sample step sizes under the nominal probability measure as well as under the exponentially tilted distribution

$$\mathbb{E}\big[\exp\big(\theta_\gamma(\gamma - \tau_1)\big)\mathbf{1}(\tau_1 \in dt)\big].$$

We also introduce the following assumptions on the Gaussian field $(X(t))_{t\in T}$.

(B1) $\mathbb{E}[X(t)] = 0$.

(B2) $\mathbb{E}[\exp(p \sup_{t\in T} X(t))] < \infty$ for any $p \geq 1$.

Remark 2.1. By Borell's inequality [1], Thm. 2.1.1, if T is bounded, a sufficient condition for (B2) is

$$\mathrm{Var}\big(X(s) - X(t)\big) \leq c|s - t|^\beta$$

for any $s, t \in T$ and some $c > 0$, $\beta > 0$. Define $\sigma^2(t) = \mathrm{Var}(X(t))$. Then, under (B1) and (B2),

$$\sup_{t\in T} \sigma^2(t) = \sup_{t\in T} \mathbb{E}\big[X(t)^2\big] \leq \mathbb{E}\Big[\sup_{t\in T} X(t)^2\Big] < \infty.$$

We also assume that sampling $(X(t_i))_{i=1,\ldots,d}$ costs $c(\{t_1, \ldots, t_d\}) \geq d$ units of operations. In this paper, a single operation can be any single arithmetic operation, generating a uniform random variable, calculating a Gaussian cumulative probability function, comparing any two numbers, or

retrieving a Gaussian quantile value. For simplicity in the notation, we shall simply write $c(d) = c(\{t_1, \ldots, t_d\})$. The locations $t_1, \ldots, t_d$ will be assumed given throughout our development.

The following is our performance guarantee for our final algorithm, **Algorithm M**, presented in Section 6. A crucial part of the theorem is that the points $t_1, \ldots, t_d$ for any $d \geq 1$ lie in a fixed set T.

Theorem 2.2. *Assume the conditions* (A1), (A2), *and* (B1), (B2). *Then* ***Algorithm M*** *outputs* $M(t_1), \ldots, M(t_d)$ *without any bias, and the total number* R *of operations in the execution of this algorithm satisfies* $\mathbb{E}[R^p] = O(c(d)^{p+o(1)})$ *for any* $p \geq 1$.

3. Building blocks for our algorithm

This section serves as a roadmap for the algorithmic elements behind our approach. We start with a few definitions:

$$\overline{X}_n = \max_{i=1,\ldots,d} X_n(t_i), \qquad \underline{X}_n = \min_{i=1,\ldots,d} X_n(t_i).$$

We shall use $\overline{X}$ and $\underline{X}$ to denote generic copies of $\overline{X}_n$ and $\underline{X}_n$, respectively.

Our algorithm relies on three random times which are finite a.s. They depend on parameters $a \in (0, 1]$, $C \in \mathbb{R}$, $0 < \gamma < \mathbb{E}[A_1]$ to be chosen later.

1. $N_X = N_X(a, C)$: for all $n > N_X$,

$$\overline{X}_n \leq a \log n + C.$$

   A straightforward Borel–Cantelli argument shows that N_X is finite.

2. $N_A = N_A(\gamma)$: for all $n > N_A$,

$$A_n \geq \gamma n. \tag{3.1}$$

3. $N_a = N_a(\gamma, a, C)$: for all $n > N_a$,

$$n\gamma \geq A_1 n^a \exp(C - \underline{X}_1). \tag{3.2}$$

Applying the defining properties of these random times, we find that for $n > N := \max(N_A, N_X, N_a)$ and any $t \in \{t_1, \ldots, t_d\}$,

$$\begin{aligned} -\log A_n + X_n(t) &\leq -\log A_n + \overline{X}_n \\ &\leq -\log A_n + a \log n + C \\ &\leq -\log(n\gamma) + a \log n + C \\ &\leq -\log A_1 + \underline{X}_1 \\ &\leq -\log A_1 + X_1(t). \end{aligned}$$

We conclude that, for $t \in \{t_1, \ldots, t_d\}$,

$$\sup_{n\geq 1}\{-\log A_n + X_n(t) + \mu(t)\} = \max_{1\leq n\leq N}\{-\log A_n + X_n(t) + \mu(t)\}, \tag{3.3}$$

and thus we can sample $M(t_1), \ldots, M(t_d)$ with computational complexity $Nc(d)$ plus the overhead to identify N_A and N_X.

From an algorithmic point of view, the key is the simulation of the random variables N_X, N_A, and N_a. If we know how to simulate these quantities, relation (3.3) indicates that we must be able to simulate the sequences (A_n) and (X_n) up to and jointly with N which heavily depends on both sequences.

Remark 3.1. Assumptions (A1) and (A2) can be removed without loss of generality. To see this, we first observe that for any $r > 0$, $\tau_i(r) = \min(\tau_i, r) \leq \tau_i$ and, therefore,

$$A_n(r) = \tau_1(r) + \cdots + \tau_n(r) \leq A_n.$$

Moreover, we can select $r > 0$ so that $\gamma < \mathbb{E}[\tau_i(r)] < \mathbb{E}[\tau_i]$. Hence we can use $(A_n(r))_{n\geq 1}$ to find N_A satisfying

$$A_n > A_n(r) > \gamma n.$$

Because $0 \leq \tau_n(r) \leq r$, the moment generating function of $\tau_n(r)$ exists on the whole real line. By convexity, one can always choose θ_γ which satisfies $\mathbb{E}[\exp(\theta_\gamma(\gamma - \tau_1(r)))] = 1$, as long as $\mathrm{Var}(\tau_i(r)) > 0$ (i.e. if $\tau_i > 0$ is non-deterministic, by choosing $r > 0$ large enough). If τ_i is deterministic, the strategy can be implemented directly, that is, we can simply select N_A deterministic. Once we find N_A, we can recover $(A_n)_{n\leq N_A}$ from $(A_n(r))_{n\leq N_A}$ by replacing $\tau_n(r)$ with an independent sample of τ_n given $\tau_n \geq r$, for any $n \leq N_A$ such that $\tau_n(r) = r$, and keeping $\tau_n(r)$ if it is less than r.

Given our previous discussion, we might concentrate on how to sample from an exponentially tilted distribution of a random variable with compact support, which may require evaluating the moment generating function in closed form. Sampling from an exponentially tilted distribution is straightforward for random variables with finite support. So, the strategy can be implemented for $\lfloor \tau_i(r)\Delta \rfloor/\Delta < \tau_i(r)$, where $\lfloor\cdot\rfloor$ is the round-down operator, picking $\Delta > 0$ sufficiently small so that $\mathbb{E}[\lfloor \tau_i(r)\Delta \rfloor/\Delta] > \gamma$. Once $\lfloor \tau_i(r)\Delta \rfloor$ is sampled we can easily simulate $\tau_i(r)$ using acceptance/rejection. The details of this idea are explained in [8].

4. Sampling a random walk up to a last passage time

In this section, we discuss the simulation of the random time N_A jointly with the sequence $(A_n)_{n\geq 0}$. We lead this discussion in the context of a general random walk $(S_n)_{n\geq 0}$ starting from the origin with negative drift. It is eventually negative almost surely. We review an algorithm from [9] for finding a random time N_S such that $S_n < 0$ for all $n > N_S$. Our aim is to develop a sampling algorithm for $(S_1, \ldots, S_{N_S+\ell})$ for any fixed $\ell \geq 0$. Our discussion here provides a simpler version of the algorithm in [9] and allows us to provide a self-contained development of the whole procedure for sampling $M(t_1), \ldots, M(t_d)$.

The algorithm is based on alternately sampling upcrossings and downcrossings of the level 0. We write $\xi_0^+ = 0$ and, for $i \geq 1$, we recursively define

$$\xi_i^- = \begin{cases} \inf\{n \geq \xi_{i-1}^+ : S_n < 0\} & \text{if } \xi_{i-1}^+ < \infty, \\ \infty & \text{otherwise} \end{cases}$$

together with

$$\xi_i^+ = \begin{cases} \inf\{n \geq \xi_i^- : S_n \geq 0\} & \text{if } \xi_i^- < \infty, \\ \infty & \text{otherwise.} \end{cases}$$

As usual, in these definitions the infimum of an empty set should be interpreted as ∞. Writing

$$N_S = \sup\{\xi_n^- : \xi_n^- < \infty\},$$

and keeping in mind that (S_n) starts as zero and has negative drift, we have by construction $0 \leq N_S < \infty$ almost surely, and for $n > N_S$, $S_n \leq 0$. The random variable $N_S - 1$ is an upward last passage time:

$$N_S - 1 = \sup\{n \geq 0 : S_n \geq 0\}.$$

We write $\mathbb{P}_x$ for the distribution of the random walk starting from $x \in \mathbb{R}$, so that $\mathbb{P} = \mathbb{P}_0$. We assume the existence of *Cramér's root*, $\theta > 0$, satisfying $\mathbb{E}[\exp(\theta S_1)] = 1$. Also assume that we can sample a random walk starting from x under $\mathbb{P}_x^\theta$, which is defined with respect to $\mathbb{P}_x$ through an exponential change of measure: on the σ-field generated by $S_1, \ldots, S_n$ we have

$$\frac{d\mathbb{P}_x}{d\mathbb{P}_x^\theta} = \exp\big(-\theta(S_n - x)\big).$$

Under $\mathbb{P}_x^\theta$, the random walk (S_n) has positive drift.

The rest of this section is organized as follows:

- In Section 4.1, we discuss sampling of downcrossing and upcrossing segments of the random walk.
- In Section 4.2, we explain how to sample beyond N_S.
- In Section 4.3, we presents our full algorithm for sampling $(S_1, \ldots, S_{N_S+\ell})$.

4.1. Downcrossings and upcrossings

To introduce the algorithm, we first need the following definitions:

$$\tau^- = \inf\{n \geq 0 : S_n < 0\}, \qquad \tau^+ = \inf\{n \geq 0 : S_n \geq 0\}.$$

For $x \geq 0$, it is immediate that we can sample a downcrossing segment $S_1, \ldots, S_{\tau^-}$ under $\mathbb{P}_x$ due to the negative drift, and we record this for later use in a pseudocode function. *Throughout*

this paper, 'sample' in pseudocode stands for 'sample independently of anything that has been sampled already.'

Function SAMPLEDOWNCROSSING(x)**: Samples** $(S_1, \ldots, S_{\tau^-})$ **under** $\mathbb{P}_x$ **for** $x \geq 0$

Step 1: Return sample $(S_1, \ldots, S_{\tau^-})$ under $\mathbb{P}_x$.
Step 2: EndFunction

Sampling an upcrossing segment is much more challenging because it is possible that $\tau^+ = \infty$, so an algorithm needs to be able to detect this event within a finite amount of computing resources. For this reason, we understand sampling an upcrossing segment under $\mathbb{P}_x$ for $x < 0$ to mean that an algorithm outputs $(S_1, \ldots, S_{\tau^+})$ if $\tau^+ < \infty$, and otherwise it outputs 'degenerate.'

Our algorithm is based on importance sampling and exponential tilting, techniques that are widely used for rare event simulation [4], page 164. Under Assumption (A1), it is well-known that $\mathbb{E}_x^\theta[\tau^+] < \infty$; for instance, see [2], page 231, Cor. 4.4. In particular, the expected time to simulate $(S_1, \ldots, S_{\tau^+})$ is finite under $\mathbb{P}_x^\theta$ for any $x < 0$.

The following proposition is the key to our algorithm.

Proposition 4.1. *Let $x < 0$. Suppose there exists some $\theta > 0$ with $\mathbb{E}[\exp(\theta S_1)] = 1$. With U being a standard uniform random variable independent of (S_n) under $\mathbb{P}_x^\theta$, we have the following*:

1. *The law of $\mathbf{1}(\tau^+ < \infty)$ under $\mathbb{P}_x$ equals the law of $\mathbf{1}(U \leq \exp(-\theta(S_{\tau^+} - x)))$ under $\mathbb{P}_x^\theta$.*
2. *The law of τ^+ given $\tau^+ < \infty$ under $\mathbb{P}_x$ equals the law of τ^+ given $U \leq \exp(-\theta(S_{\tau^+} - x))$ under $\mathbb{P}_x^\theta$.*
3. *For any $k \geq 1$, the law of $(S_1, \ldots, S_k)$ given $\tau^+ = k$ under $\mathbb{P}_x$ equals the law of $(S_1, \ldots, S_k)$ given $U \leq \exp(-\theta(S_{\tau^+} - x))$ and $\tau^+ = k$ under $\mathbb{P}_x^\theta$.*

Proof. For any integer $k \geq 1$ and Borel sets $B_1, B_2, \ldots, B_k$, we have

$$\begin{aligned}
&\mathbb{P}_x\big(S_1 \in B_1, \ldots, S_k \in B_k, \tau^+ = k\big) \\
&\quad = \mathbb{E}_x^\theta\big[\exp\big(-\theta(S_k - x)\big)\mathbf{1}\big(S_1 \in B_1, \ldots, S_k \in B_k, \tau^+ = k\big)\big] \\
&\quad = \mathbb{E}_x^\theta\big[\mathbf{1}\big(U \leq \exp\big(-\theta(S_{\tau^+} - x)\big)\big)\mathbf{1}\big(S_1 \in B_1, \ldots, S_k \in B_k, \tau^+ = k\big)\big].
\end{aligned}$$

All claims are elementary consequences of this identity, upon noting that $\tau^+ < \infty$ under $\mathbb{P}_x^\theta$. □

This proposition immediately yields the following algorithm.

Function SAMPLEUPCROSSING(x)**: Samples** $(S_1, \ldots, S_{\tau^+})$ **under** $\mathbb{P}_x$ **for** $x < 0$

Step 1: $S \leftarrow$ sample $(S_1, \ldots, S_{\tau^+})$ under $\mathbb{P}_x^\theta$
Step 2: $U \leftarrow$ sample a standard uniform random variable
Step 3: If $U \leq \exp(-\theta(S_{\tau^+} - x))$
Step 4: Return S
Step 5: Else

Step 6: Return 'degenerate'
Step 7: EndIf
Step 8: EndFunction

4.2. Beyond N_S

We next describe how to sample $(S_1, \ldots, S_\ell)$ from $\mathbb{P}_x$ conditionally on $\tau^+ = \infty$ for $x < 0$. Because $\tau^+ = \infty$ is equivalent to $\sup_{k \le \ell} S_k < 0$ and $\sup_{k > \ell} S_k < 0$ for any $\ell \ge 1$, after sampling $S_1, \ldots, S_\ell$, by the Markov property we can use SAMPLEUPCROSSING(S_ℓ) to verify whether or not $\sup_{k>\ell} S_k < 0$. This observation immediately yields an acceptance/rejection algorithm that achieves our goal.

Function SAMPLEWITHOUTRECORDS(x, ℓ)**: Samples** $(S_1, \ldots, S_\ell)$ **from** $\mathbb{P}_x$ **given** $\tau^+ = \infty$ **for** $\ell \ge 1$**,** $x < 0$

Step 1: Repeat
Step 2: $S \leftarrow$ sample $(S_1, \ldots, S_\ell)$ under $\mathbb{P}_x$
Step 3: Until $\sup_{1 \le k \le \ell} S_k < 0$ and SAMPLEUPCROSSING(S_ℓ) is 'degenerate'
Step 4: Return S
Step 5: EndFunction

4.3. Sampling a random walk until a last passage time

We summarize our findings in this section in our full algorithm for sampling $(S_0, \ldots, S_{N_S+\ell})$ under $\mathbb{P}$ given some $\ell \ge 0$. The validity of the algorithm is a direct consequence of the strong Markov property.

Algorithm S: Samples $(S_0, \ldots, S_{N_S+\ell})$ **under** $\mathbb{P}$ **for** $\ell \ge 0$**.** # We use S_{end} to denote the last element of S.

Step 1: $S \leftarrow [0]$
Step 2: Repeat
Step 3: DowncrossingSegment $\leftarrow$ SAMPLEDOWNCROSSING(S_{end})
Step 4: $S \leftarrow [S,$ DowncrossingSegment$]$
Step 5: UpcrossingSegment $\leftarrow$SAMPLEUPCROSSING(S_{end})
Step 6: If UpcrossingSegment is not 'degenerate'
Step 7: $S \leftarrow [S,$ UpcrossingSegment$]$
Step 8: EndIf
Step 9: Until UpcrossingSegment is 'degenerate'
Step 10: If $\ell > 0$
Step 11: $S \leftarrow [S,$ SAMPLEWITHOUTRECORDS$(S_{\text{end}}, \ell)]$
Step 12: EndIf

5. Record-breaker technique for the maximum of a Gaussian field

After the excursion to random walks in Section 4 we return to the main theme of this paper. In particular, we stick to the notation and assumptions of Sections 1–3. Define $\eta_0 = n_0$ for some fixed n_0 to be defined later. Let $(X_n)_{n\geq 1}$ be i.i.d. copies of X and define, for $i \geq 1$, a sequence of *record-breaking times* (η_i) through

$$\eta_i = \begin{cases} \inf\{n > \eta_{i-1} : \overline{X}_n > a \log n + C\} & \text{if } \eta_{i-1} < \infty, \\ \infty & \text{otherwise.} \end{cases}$$

It is the aim of this section to develop a sampling algorithm for $(X_1, \ldots, X_{N_X+\ell})$ for any fixed $\ell \geq 0$, where

$$N_X = \max\{\eta_i : \eta_i < \infty\}.$$

Here and in what follows, we write X_i for a sample path at the given points $t_1, \ldots, t_d \in T$. Section 5.1 first discusses an algorithm to sample (X_n) up to a single record. For this algorithm to work, n_0 needs to be large enough so that $\mathbb{P}(\overline{X} > a \log n + C)$ is controlled for every $n > n_0$; the choice of n_0 is also discussed in Section 5.1. Section 5.2 describes how to sample (X_n) beyond the last record-breaking time. Section 5.3 presents our algorithm for sampling $(X_1, \ldots, X_{N_X+\ell})$.

5.1. Breaking a single record

We define for $n \geq 0$,

$$T_n = \inf\big\{k \geq 1 : \overline{X}_k > a \log(n+k) + C\big\}.$$

We describe an algorithm that outputs 'degenerate' if $T_n = \infty$ and $(X_1, \ldots, X_{T_n})$ if $T_n < \infty$. Ultimately, the strategy is based on acceptance/rejection. We will eventually sample T_n given $T_n < \infty$ using a suitable random variable K as a proxy with probability mass function g_{n_0}, which we discuss later in this subsection. In order to apply this acceptance/rejection strategy, we need to introduce auxiliary sampling distributions.

Our algorithm makes use of a measure $\mathbb{P}^{(n)}$ that is designed to appropriately approximate the conditional distribution of X given $\overline{X} > a \log n + C$, which is defined through

$$\frac{d\mathbb{P}^{(n)}}{d\mathbb{P}}(x) = \frac{\sum_{i=1}^d \mathbf{1}(x(t_i) > a \log n + C)}{\sum_{i=1}^d \mathbb{P}(X(t_i) > a \log n + C)}.$$

For any index $j \in \{1, \ldots, d\}$ and $t \in \{t_1, \ldots, t_d\}$, define $w^j(t) = \mathrm{Cov}(X(t), X(t_j))/\mathrm{Var}(X(t_j))$. Since X is centered Gaussian $X(t) - w^j X(t_j)$ and $X(t_j)$ are uncorrelated, hence independent.

Now one readily verifies that the following algorithm outputs samples from $\mathbb{P}^{(n)}$. Here and in what follows, Φ is the standard normal distribution function.

Function CONDITIONEDSAMPLEX(a, C, n)**: Samples** X **from** $\mathbb{P}^{(n)}$

Step 1: $\nu \leftarrow$ sample with probability mass function

$$\mathbb{P}(\nu = j) = \frac{\mathbb{P}(X(t_j) > a \log n + C)}{\sum_{i=1}^{d} \mathbb{P}(X(t_i) > a \log n + C)}$$

Step 2: $U \leftarrow$ sample a standard uniform random variable
Step 3: $X(t_\nu) \leftarrow \sigma(t_\nu)\Phi^{-1}(U + (1-U)\Phi(\frac{a \log n + C}{\sigma(t_\nu)}))$ # Conditions on $X(t_\nu) > a \log n + C$
Step 4: $Y \leftarrow$ sample of X under $\mathbb{P}$
Step 5: Return $Y - w^\nu Y(t_\nu) + w^\nu X(t_\nu)$
Step 6: EndFunction

We are now ready to see how CONDITIONEDSAMPLEX is used to sample until the first record.

Function SAMPLESINGLERECORD(a, C, n)**: Samples** $(X_1, \ldots, X_{T_n})$ **for** $a \in (0, 1]$**,** $C \in \mathbb{R}$**,** $n \geq n_0$

Step 1: $K \leftarrow$ sample from pmf g_{n_0}
Step 2: $(X_1, \ldots, X_{K-1}) \leftarrow$ i.i.d. sample under $\mathbb{P}$
Step 3: $X_K \leftarrow$ CONDITIONEDSAMPLEX$(a, C, n + K)$
Step 4: $U \leftarrow$ sample a standard uniform random variable
Step 5: If $\overline{X}_k \leq a \log(n + k) + C$ for $k = 1, \ldots, K - 1$ and $U g_{n_0}(K) \leq d\mathbb{P}/d\mathbb{P}^{(n+K)}(X_K)$
Step 6: Return $(X_1, \ldots, X_K)$
Step 7: Else
Step 8: Return 'degenerate'
Step 9: EndIf
Step 10: EndFunction

The following proposition shows that SAMPLESINGLERECORD achieves the desired goal.

Proposition 5.1. *Assume the condition*

$$\sum_{i=1}^{d} \mathbb{P}\big(X(t_i) > a \log(n_0 + k) + C\big) \leq g_{n_0}(k) \qquad \text{for } k \geq 1. \tag{5.1}$$

For $n \geq n_0$, if $(\widetilde{X}_1, \ldots, \widetilde{X}_{\widetilde{T}})$ has the distribution of the output of SAMPLESINGLERECORD *conditioned on not being 'degenerate,' then we have*

1. *the algorithm* SAMPLESINGLERECORD *returns 'degenerate' with probability $\mathbb{P}(T_n = \infty)$,*
2. *the length $\widetilde{T}$ has the same distribution as T_n given $T_n < \infty$, and*
3. *the distribution of $(\widetilde{X}_1, \ldots, \widetilde{X}_{\widetilde{T}})$ given $\widetilde{T} = \ell$ is the same as the distribution of $(X_1, \ldots, X_\ell)$ given $T_n = \ell$.*

Proof. Write $A_m = \{x \in \mathbb{R}^d : \max_i x_i > a\log(n+m)+C\}$ for $m \geq 1$. For $B_1 \subset A_1^c, \ldots, B_{k-1} \subset A_{k-1}^c$ and $B_k \subset A_k$, we have

$$\begin{aligned}
&\mathbb{P}(\widetilde{X}_1 \in B_1, \ldots, \widetilde{X}_{k-1} \in B_{k-1}, \widetilde{X}_k \in B_k, \widetilde{T} = k) \\
&\quad = \mathbb{P}(K=k)\mathbb{P}(X \in B_1)\cdots\mathbb{P}(X \in B_{k-1}) \\
&\qquad \times \mathbb{P}^{(n+k)}\left(U g_{n_0}(k) \leq \frac{d\mathbb{P}}{d\mathbb{P}^{(n+k)}}(X), X \in B_k\right) \\
&\quad = \mathbb{P}(K=k)\mathbb{P}(X \in B_1)\cdots\mathbb{P}(X \in B_{k-1}) \\
&\qquad \times \mathbb{E}^{(n+k)}\left(\frac{1}{g_{n_0}(k)}\frac{d\mathbb{P}}{d\mathbb{P}^{(n+k)}}(X) I(X \in B_k)\right) \\
&\quad = g_{n_0}(k)\mathbb{P}(X \in B_1)\cdots\mathbb{P}(X \in B_{k-1})\frac{\mathbb{P}(X \in B_k)}{g_{n_0}(k)} \\
&\quad = \mathbb{P}(X_1 \in B_1, \ldots, X_k \in B_k, T_n = k),
\end{aligned}$$

and all claims follow from this identity. The second equality follows from the assumption, which implies that $d\mathbb{P}/d\mathbb{P}^{(n+k)}(x)/g_{n_0}(k)$ is bounded by 1 for all $k \geq 1$ and $x \in \mathbb{R}^d$. □

Choosing n_0 and the density g_{n_0}

We start with g_{n_0}, guided by (5.1) and the requirement that we need to sample from g_{n_0}. The random variable K is a proxy for the first-record epoch T_{n_0}, the distribution of which we can approximate with a union-bound. This leads to the idea to use, for $k \geq 1$,

$$g_{n_0}(k) = \frac{\int_{k-1}^{k} \phi((a\log(n_0+s)+C)/\overline{\sigma})\,ds}{\int_0^\infty \phi((a\log(n_0+s)+C)/\overline{\sigma})\,ds}, \tag{5.2}$$

where $\phi(\cdot)$ is the density function of the standard normal distribution, $\overline{\sigma}^2 = \max_{t\in T}\mathrm{Var}(X(t))$. The following lemma resolves the sampling question.

Lemma 5.2. *Let U be a uniform random variable on $(0,1)$. The quantity*

$$\left\lceil \exp\left\{\frac{\overline{\sigma}^2}{a^2} - \frac{C}{a} + \frac{\overline{\sigma}}{a}\overline{\Phi}^{-1}\left(U\overline{\Phi}\left(\frac{a\log n_0 + C}{\overline{\sigma}} - \frac{\overline{\sigma}}{a}\right)\right)\right\} - n_0 \right\rceil$$

has probability mass function g_{n_0}, where $\lceil\cdot\rceil$ is the round-up operator, $\overline{\Phi} = 1-\Phi$, and $\overline{\Phi}^{-1}$ is the inverse of $\overline{\Phi}$.

Proof. Write $f_{n_0}(U)$ for the expression inside the exponential operator. For $k \geq 1$, we have

$$\begin{aligned}
\mathbb{P}\big(\lceil \exp\big(f_{n_0}(U)\big) - n_0 \rceil \geq k\big) &= \mathbb{P}\big(f_{n_0}(U) > \log(n_0+k-1)\big) \\
&= \frac{\overline{\Phi}((a\log(n_0+k-1)+C)/\overline{\sigma} - \overline{\sigma}/a)}{\overline{\Phi}((a\log n_0 + C)/\overline{\sigma} - \overline{\sigma}/a)},
\end{aligned}$$

so it remains to show that this equals

$$\sum_{m\geq k} g_{n_0}(m) = \frac{\int_{n_0+k-1}^{\infty} \phi((a\log x + C)/\overline{\sigma})\,dx}{\int_{n_0}^{\infty} \phi((a\log x + C)/\overline{\sigma})\,dx}.$$

To see this, we note that, for $y > 0$,

$$\begin{aligned}\int_y^{\infty} \phi\big((a\log(x) + C)/\overline{\sigma}\big)\,dx &= \frac{1}{\sqrt{2\pi}} \int_{\log y}^{\infty} \exp\left(-\frac{(at + C)^2}{2\overline{\sigma}^2} + t\right) dt \\ &= \frac{e^{-C/a}}{\sqrt{2\pi}\phi(\overline{\sigma}/a)/(\overline{\sigma}/a)} \times \overline{\Phi}\big((a\log y + C)/\overline{\sigma} - \overline{\sigma}/a\big) \quad (5.3)\end{aligned}$$

and we thus obtain the claim. □

It is convenient to give the integral encountered on the left-hand side of (5.3) a name: for $y > 0$, we set

$$r(y) = \int_y^{\infty} \phi\big((a\log(x) + C)/\overline{\sigma}\big)\,dx.$$

The following proposition shows that, for large enough n_0, the choice of g_{n_0} as in (5.2) ensures that (5.1) is satisfied. The proposition also shows how $\mathbb{P}(T_n < \infty)$ for $n \geq n_0$ can be controlled explicitly.

Proposition 5.3. *If n_0 satisfies $a\log n_0 + C \geq \overline{\sigma}$ and $dr(n_0) \leq \delta$ for a given $\delta \in (0, 1)$, then* (5.1) *is satisfied and* SAMPLESINGLERECORD(a, C, n) *returns 'degenerate' at least with probability* $1 - \delta$.

Proof. Since $\overline{\Phi}(x) \leq \phi(x)$ for $x \geq 1$, $dr(n_0) \leq \delta$, and in view of (5.3) we have

$$\begin{aligned}\sum_{i=1}^{d} \mathbb{P}\big(X(t_i) > a\log(n_0 + k) + C\big) &\leq d\overline{\Phi}\big((a\log(n_0 + k) + C)/\overline{\sigma}\big) \\ &\leq d\phi\big((a\log(n_0 + k) + C)/\overline{\sigma}\big) \\ &\leq d\int_{k-1}^{k} \phi\big((a\log(n_0 + s) + C)/\overline{\sigma}\big)\,ds \\ &= d\int_0^{\infty} \phi\big((a\log(n_0 + s) + C)/\overline{\sigma}\big)\,ds g_{n_0}(k) \\ &= dr(n_0)g_{n_0}(k) < \delta g_{n_0}(k). \quad (5.4)\end{aligned}$$

This proves the first claim.

Applying Proposition 5.1 and (5.4) for every k, the probability that SAMPLESINGLERECORD does not return 'degenerate' is bounded as follows:

$$\begin{aligned}\sum_{k=1}^{\infty}\mathbb{P}(T_n=k) &\le \sum_{k=1}^{\infty}\sum_{i=1}^{d}\mathbb{P}\big(X(t_i)>a\log(n+k)+C\big)\\ &\le \sum_{k=1}^{\infty}\sum_{i=1}^{d}\mathbb{P}\big(X(t_i)>a\log(n_0+k)+C\big)\\ &< \delta\sum_{k=1}^{\infty}g_{n_0}(k)=\delta,\end{aligned}$$

which proves the second claim. □

5.2. Beyond N_X

We next describe how to sample $(X_1,\dots,X_n)$ conditionally on $T_n=\infty$. As in Section 4.2, we use an acceptance/rejection algorithm, but we have to modify the procedure slightly because we work with a sequence of i.i.d. random fields instead of a random walk.

Function SAMPLEWITHOUTRECORDX(n,ℓ)**: Samples** $(X_1,\dots,X_\ell)$ **conditionally on** $T_n=\infty$ **for** $\ell\ge 1$

Step 1: Repeat
Step 2: $X\leftarrow$ sample $(X_1,\dots,X_\ell)$ under $\mathbb{P}$
Step 3: Until $\sup_{1\le k\le\ell}[X_k-a\log(n+k)]<C$
Step 4: Return X
Step 5: EndFunction

5.3. The full algorithm

We summarize our findings in this section in our full algorithm for sampling $(X_1,\dots,X_{N_X+\ell})$ under $\mathbb{P}$ given some $\ell\ge 0$.

The idea is to successively apply SAMPLESINGLERECORD to generate the η_i from the beginning of this section. Starting from $\eta_0=n_0$ satisfying the requirements in Proposition 5.3, we generate T_n where n is replaced by each of the subsequent η_i. As a result, we have $\mathbb{P}(\eta_i=\infty|\eta_{i-1}<\infty)\ge 1-\delta$ by Proposition 5.3. Thus, the number of records is bounded in probability by a geometric random variable with parameter $1-\delta$.

Algorithm X: Samples $(X_1,\dots,X_{N_X+\ell})$ **given** $a\in(0,1]$**,** $\delta\in(0,1)$**,** $C\in\mathbb{R}$**,** $\sigma>0$**,** $\ell\ge 0$**.** # n_0 must satisfy the requirements in Proposition 5.3.

Step 1: $X\leftarrow[\,]$, $\eta\leftarrow n_0$

Step 2: $X \leftarrow$ sample $(X_1, \ldots, X_\eta)$ under $\mathbb{P}$
Step 3: Repeat
Step 4: segment $\leftarrow$ SAMPLESINGLERECORD(a, C, η)
Step 5: If segment is not 'degenerate'
Step 6: $X \leftarrow [X, \text{segment}]$
Step 7: $\eta \leftarrow \text{length}(X)$
Step 8: EndIf
Step 9: Until segment is 'degenerate'
Step 10: If $\ell > 0$
Step 11: $X \leftarrow [X, \text{SAMPLEWITHOUTRECORDX}(\eta, \ell)]$
Step 12: EndIf

6. Final algorithm and proof of Theorem 2.2

In this section, we give our final algorithm. We also provide the remaining arguments showing why the algorithm outputs exact samples and prove a bound on the computational complexity. Together these proofs establish Theorem 2.2.

We start with a description of our final algorithm for sampling M, which exploits that for $S_n = \gamma n - A_n$ and $N_A = N_S$, we have $S_n < 0$ and therefore $A_n \geq \gamma n$ for $n > N_A$.

Algorithm M: Samples $(M(t_1), \ldots, M(t_d))$ **given** $\delta \in (0, 1)$**,** $a \in (0, 1]$**,** $\gamma < \mathbb{E}A_1$**,** $C \in \mathbb{R}$**,** σ**.**

Step 1: Sample $A_1, \ldots, A_{N_A}$ using Steps 1–9 from **Algorithm S** with $S_n = \gamma n - A_n$.
Step 2: Sample $X_1, \ldots, X_{N_X}$ using Steps 1–9 from **Algorithm X**.
Step 3: Calculate N_a with (3.2) and set $N = \max(N_A, N_X, N_a)$.
Step 4: If $N > N_A$
Step 5: Sample $A_{N_A+1}, \ldots, A_N$ as in Steps 10–12 from **Algorithm S** with $S_n = \gamma n - A_n$.
Step 6: EndIf
Step 7: If $N > N_X$
Step 8: Sample $X_{N_X+1}, \ldots, X_N$ as in Steps 10–12 from **Algorithm X**.
Step 9: EndIf
Step 10: Return $M(t_i) = \max_{1 \leq n \leq N}\{-\log A_n + X_n(t_i) + \mu(t_i)\}$ for $i = 1, \ldots, d$.

The pathwise construction in Section 3 implied that the output of **Algorithm M** is an exact sample of $\{M(t_1), \ldots, M(t_d)\}$. Thus it remains to study the running time of **Algorithm M**.

6.1. Computational complexity

We next study the truncation point N in (3.3). Because the number of records is bounded in probability by a geometric random variable, it is clear that $N < \infty$ almost surely. As we now explain,

the number of rejections in Steps 5 and 8 in **Algorithm M** are also bounded in probability by geometric random variables. In particular, in Step 5 the sample path is accepted with probability no less than $\mathbb{P}_0(\tau_+ = \infty)$, while in Step 8 the acceptance probability is bounded below by $\mathbb{P}(T_0 = \infty)$. Both bounds are strictly positive, so the total path length generated is on the same order as N.

Our aim is to study the dependence of our algorithm on the dimension d. The only places where d enters the algorithm are in the definition of n_0 and the measure $\mathbb{P}^{(n)}$. Sampling from the latter happens at most a geometric number of times with parameter $1-\delta$, so the computational complexity is dominated by the choice of n_0.

For any $\zeta > 0$, if d is large enough and if we ignore rounding, the following choice of $n_0 = n_0(d)$

$$\log\bigl(n_0(d)\bigr) = \frac{\overline{\sigma}^2}{a^2} - \frac{C}{a} + \frac{\overline{\sigma}}{a}\sqrt{(2+\zeta)\log\biggl(\frac{de^{-C/a}}{\delta\sqrt{2\pi}\phi(\overline{\sigma}/a)/(\overline{\sigma}/a)}\biggr)}$$

satisfies the assumption $dr(n_0) \le \delta$ of Proposition 5.3.

The following result will be needed for the proof of the second part of Theorem 2.2. Recall that K is a positive integer-valued random variable with probability mass function g_{n_0}.

Lemma 6.1. *For $p \ge 1$, we have $\log(\mathbb{E}[K^p]) = O(\log n_0)$ as $d \to \infty$.*

Proof. Assume n_0 sufficiently large. Then

$$\begin{aligned}
\mathbb{E}\bigl[K^p\bigr] &= \sum_{k=1}^{\infty} k^p g_{n_0}(k) \\
&\le \frac{\int_0^\infty (s+n_0)^p \phi((a\log(n_0+s)+C)/\overline{\sigma})\,ds}{\int_0^\infty \phi((a\log(n_0+s)+C)/\overline{\sigma})\,ds} \\
&= e^{\frac{p^2\overline{\sigma}}{2a^2} - \frac{Cp}{a}} \frac{\overline{\Phi}((a\log(n_0)+C-p\overline{\sigma}^2/a)/\overline{\sigma} - \overline{\sigma}/a)}{\overline{\Phi}((a\log(n_0)+C)/\overline{\sigma} - \overline{\sigma}/a)} \\
&\le e^{\frac{p^2\overline{\sigma}}{2a^2} - \frac{Cp}{a}} \frac{\frac{1}{(a\log(n_0)+C-p\overline{\sigma}^2/a)/\overline{\sigma}-\overline{\sigma}/a}\phi((a\log(n_0)+C-p\overline{\sigma}^2/a)/\overline{\sigma} - \overline{\sigma}/a)}{\frac{(a\log(n_0)+C)/\overline{\sigma}-\overline{\sigma}/a}{((a\log(n_0)+C)/\overline{\sigma}-\overline{\sigma}/a)^2+1}\phi((a\log(n_0)+C)/\overline{\sigma} - \overline{\sigma}/a)} \\
&\le 2e^{\frac{p^2\overline{\sigma}}{2a^2} - \frac{Cp}{a}} \exp\biggl(-\frac{p^2\overline{\sigma}^2}{2a^2} + \frac{p\overline{\sigma}}{a}\bigl(a\log(n_0)+C\bigr)/\overline{\sigma} - \overline{\sigma}/a\biggr) \\
&= 2\exp\biggl(p\log(n_0) - \frac{p\overline{\sigma}^2}{a^2}\biggr).
\end{aligned}$$

Therefore, $\log\mathbb{E}[K^p] \le p\log(n_0) + \log 2 - p\overline{\sigma}^2/a^2$. □

Next, we show that $\log \mathbb{E}[N_X^p] = O(\sqrt{\log d})$. We have the decomposition

$$N_X = n_0 + \sum_{i=1}^{G} K_i,$$

where K_i are i.i.d. copies of K, G is the last time that the segment is not 'degenerate' and the definition of n_0 implies $\log(n_0(d)) = O(\sqrt{\log d})$.

Proposition 5.3 shows that G is bounded by a geometric random variable G' with parameter δ almost surely, while G' is independent of the sequence (K_i). Therefore, we have by Jensen's inequality

$$\begin{aligned}\mathbb{E}\big[N_X^p\big] &\le \mathbb{E}\left[\left(n_0 + \sum_{i=1}^{G'} K_i\right)^p\right] \\ &\le \mathbb{E}\left[\left(n_0^p + \sum_{i=1}^{G'} K_i^p\right)\big(1+G'\big)^{p-1}\right] \\ &= n_0^p \mathbb{E}\big[\big(1+G'\big)^{p-1}\big] + \mathbb{E}\big[K_1^p\big]\mathbb{E}\big[G'\big(1+G'\big)^{p-1}\big].\end{aligned}$$

Therefore, we have shown that $\log \mathbb{E}[N_X^p] = O(\sqrt{\log d})$, which implies that $\mathbb{E}[N_X^p] = O(d^{o(1)})$.

Clearly, N_A or N_a do not depend on d. We only need to show $\mathbb{E}[N_A^p] < \infty$, and $\mathbb{E}[N_a^p] < \infty$.

Recall that in Section 4 we sample the downcrossing segment of the random walk with the nominal distribution, then the upcrossing segment with the exponential tilted distribution. We denote the ith downcrossing segment having length τ_i^-, and the ith upcrossing segment having length τ_i^+. Therefore,

$$N_A = \sum_{i=1}^{L} \big(\tau_i^- + \tau_i^+\big),$$

where L is the first time that the upcrossing segment is 'degenerate'. Recall that τ^+ denotes the first upcrossing time of level 0. Because for any $x \le 0$,

$$\mathbb{P}_x\big(\tau^+ = \infty\big) \ge \mathbb{P}_0\big(\tau^+ = \infty\big) > 0,$$

L is a.s. bounded by a geometric random variable L' with parameter $q < 1$.

According to the discussion in Remark 3.1, we may assume without loss of generality that A_n has step sizes bounded by $r > 0$. Therefore, $S_{\tau_i^+} \le \gamma$ and $S_{\tau_i^-} \ge \gamma - r$. Thus, with Theorem 8.1 in [16], for any $p \ge 1$ and $\epsilon > 0$, there exists some constant $V > 0$, such that

$$\mathbb{E}\big[\big(\tau_i^-\big)^{p(1+\epsilon)}\big] < V \quad \text{and} \quad \mathbb{E}\big[\big(\tau_i^+\big)^{p(1+\epsilon)}\big] < V.$$

Again using Jensen's inequality, this time together with Hölder's inequality, we obtain

$$\begin{aligned}\mathbb{E}\big[N_A^p\big] &\le \mathbb{E}\left[\left(\sum_{i=1}^{L'}\big(\tau_i^- + \tau_i^+\big)\right)^p\right] \\ &\le \mathbb{E}\left[\frac{\sum_{i=1}^{L'}((\tau_i^-)^p + (\tau_i^+)^p)}{2L'}\big(2L'\big)^p\right] \\ &\le \sum_{i=1}^{\infty}\mathbb{E}\big[\big((\tau_i^-)^p + \big(\tau_i^+\big)^p\big)I\big(L' \ge i\big)\big(2L'\big)^{p-1}\big] \\ &\le 2V^{\frac{1}{1+\epsilon}}\big(\mathbb{E}\big[\big(2L'\big)^{(p-1)\frac{1+\epsilon}{\epsilon}}L'\big]\big)^{\frac{\epsilon}{1+\epsilon}} < \infty.\end{aligned}$$

The value of N_a is only required to satisfy (see (3.2))

$$N_a \ge \left(\frac{A_1\exp(C - \underline{X}_1)}{\gamma}\right)^{\frac{1}{1-a}}, \tag{6.1}$$

while $a \in (0, 1)$. Therefore, for $a \in (0, 1)$, we have

$$\begin{aligned}\mathbb{E}\big[N_a^p\big] &\le \mathbb{E}\left[\left(\left(\frac{A_1\exp(C - \underline{X}_1)}{\gamma}\right)^{\frac{1}{1-a}} + 1\right)^p\right] \\ &\le 2^{p-1}\left(\left(\frac{\mathbb{E}[A_1^{\frac{p}{1-a}}]\exp(\frac{pC}{1-a})}{\gamma^{\frac{p}{1-a}}}\right)\mathbb{E}\left[\exp\left(-\frac{p\underline{X}_1}{1-a}\right)\right] + 1\right) \\ &< \infty.\end{aligned}$$

This naturally holds by Assumption (B2). When $a = 1$, with proper choice of C, (3.2) always holds.

6.2. Choosing a, C, and γ

Although the values of $a \in (0, 1]$, $\gamma \in (0, \mathbb{E}[A_1])$ and $C \in \mathbb{R}$ do not affect the order of the computational complexity of our algorithm, we are still interested in discussing some guiding principles which can be used to choose those parameters for a reasonably good implementation.

First, note that among N_X, N_A, and N_a, only N_X would increase to ∞ as the number d of sampled locations increases to ∞. (Although N_a also increases in d, it remains bounded since $\underline{X}_1$ decreases to the minimum over T.) Assuming that C has been fixed, we can see that N_X decreases pathwise while a increases, therefore we should try to choose a close to 1. On the other hand, while $a \in (0, 1)$, we have (6.1). If $A_1\exp(C - \underline{X}_1) > \gamma$, then $N_a \nearrow \infty$ while $a \nearrow 1$. This analysis highlights a trade-off between the values of N_X and N_a with respect to the choice of a. Because $\mathbb{E}[N_X]$ is not explicitly tractable, we can have a reasonable balancing of

the computational effort by equating n_0 with $\mathbb{E}[N_a]$. In particular, we look for the largest value of $a \in (0, 1)$ satisfying the following equation

$$\exp\left(\frac{\overline{\sigma}}{a}\overline{\Phi}^{-1}\left(\delta\sqrt{2\pi}\frac{\phi(\overline{\sigma}/a)}{d\overline{\sigma}/a}\right)+\frac{\overline{\sigma}^2}{a^2}-\frac{C}{a}\right)=\mathbb{E}\left[\left(\frac{A_1\exp(C-\underline{X}_1)}{\gamma}\right)^{\frac{1}{1-a}}\right]. \tag{6.2}$$

Note that the left-hand side converges to infinity as $a \searrow 0$ while the right-hand side is bounded, but the right-hand side converges to infinity as $a \nearrow 1$ while the left-hand side is bounded, so a solution exists. Such a solution can be obtained by running a pilot run of $\underline{X}_1$, then search for the desired a numerically.

Another approach consists of selecting $a = 1$ and adjusting C so that (3.2) holds true for all $n \geq 1$. Therefore, we choose $C = \underline{X}_1 + \log(A_1/\gamma)$. The value of C is random, but the algorithms can be modified accordingly, by changing the definition of n_0, which depends on C. However, the expected computational cost has the same order as in the case when C is deterministic.

Similarly, N_A increases pathwise while γ increases, while N_a decreases if γ increases. One could get the empirical average value of N_A via simulation, and choose γ accordingly such that N_A and N_a are balanced.

7. Tolerance enforced simulation

In this section, we illustrate a general procedure which can be applied so that, for any given $\delta > 0$ one can construct a fully simulatable process M_δ, with the property that

$$\mathbb{P}\Big(\sup_{t\in T}\big|M(t)-M_\delta(t)\big|\leq\delta\Big)=1.$$

For ease of notation we focus on the case $T = [0, 1]$. The technique can be easily adapted to higher-dimensional sets T, as long as one has an infinite series representation for X which satisfies certain regularity conditions.

A TES estimator can be used to easily obtain error bounds for sample-path functionals of the underlying field. For example, in the context of parametric catastrophe bonds, it is not uncommon to use the average extreme precipitation over a certain geographical region as the trigger; see [18]. This motivates estimating $\mathbb{E}[u(\int_T M(s)\,ds)]$ for some function be consistent: u that is specified by the contract characteristics of the catastrophe bond. If u is Lipschitz continuous with Lipschitz constant 1, then one immediately obtains

$$\left|\mathbb{E}\left[u\left(\int_T M(s)\,ds\right)\right]-\mathbb{E}\left[u\left(\int_T M_\epsilon(s)\,ds\right)\right]\right|\leq|T|\epsilon.$$

The form of the TES estimator discussed in this section has the feature that $\int_T M_\epsilon(s)\,ds$ can be evaluated in closed form. Thus, a TES estimator facilitates the error analysis that could otherwise be significantly more involved.

The technique presented in this section is not limited to Gaussian processes, and we do not make this assumption here. As a result, we do not use Assumptions (B1) and (B2) in this section, but we replace them with (C1)–(C4) below. However, Assumptions (A1) and (A2) on the renewal sequence (A_n) are in force throughout this section.

7.1. An infinite series representation

We assume that $(X_n(t))_{t\in T}$ can be expressed as an almost surely convergent series of basis functions with random weights. We illustrate the procedure with a particularly convenient family of basis functions.

First, let us write any $m \geq 1$ as $m = 2^j + k$ for $j \geq 0$ and $0 \leq k \leq 2^j - 1$, and note that there is only one way to write m in this form. We assume that there exists a sequence of basis functions $(\Lambda_m(\cdot))_{m\geq 0}$, with support on $[0, 1]$ (i.e., $\Lambda_m(t) = 0$ for $t \notin [0, 1]$). Moreover, we assume that $|\Lambda_0(t)|, |\Lambda_1(t)| \leq 1$ for all $t \in [0, 1]$, and that for every $m \geq 1$,

$$\Lambda_m(t) = \Lambda_1\big(2^j\big(t - k/2^j\big)\big).$$

In other words, for $m \geq 2$, each $\Lambda_m(\cdot)$ is a wavelet with the shape of $\Lambda_1(\cdot)$, while shrunk horizontally by factor of 2^j, and shifted to start at $k/2^j$.

We introduce normalizing constants, $\lambda_0 > 0$ and $\lambda_m = \lambda' 2^{-j\alpha}$ for $m \geq 1$, where $\alpha \in (0, 1)$ and $\lambda' > 0$. Finally, we assume that

$$X_n(t) = \sum_{m=0}^{\infty} Z_{m,n}\Lambda_m(t)\lambda_m,$$

where the random variables $(Z_{m,n})_{m\geq 0,n\geq 1}$ are iid. We shall use Z to denote a generic copy of the $Z_{m,n}$'s and we shall impose suitable assumptions on the tail decay of Z. The parameter α relates to the Hölder continuity exponent of the process X_n. For example, if X_n is Brownian motion, $\alpha = 1/2$. This interpretation of α will not be used in our development, but it helps to provide intuition which can be used to inform the construction of a model based on the basis functions that we consider. For more information on the connection to the Hölder properties implied by α, the reader should consult [7] and the references therein.

Throughout, we use the following total order among the pairs $\{(m, n) : m \geq 0, n \geq 1\}$. We say $(m, n) < (m', n')$ if $m + n < m' + n'$ and in case $m + n = m' + n'$, we say that (m, n) is smaller than (m', n') in lexicographic order. In particular, we have

$$(0, 1) < (0, 2) < (1, 1) < (0, 3) < (1, 2) < (2, 1) < \cdots.$$

We let $\theta(m, n)$ be the position of (m, n) in the total order. We also define $\eta(\cdot) : \mathbb{N} \to \mathbb{N}\cup\{0\} \times \mathbb{N}$ to be the inverse function of $\theta(\cdot)$, and given $\theta \in \mathbb{N}$, we write

$$\eta(\theta) = \big(\eta_m(\theta), \eta_n(\theta)\big).$$

7.2. Building blocks for our algorithm

We now proceed to describe the construction of M_δ, which is adapted from a record-breaking technique introduced in [6]. An important building block of M_δ is the truncated series

$$X_n(t; K) = \sum_{m\leq K} \lambda_m Z_{m,n}\Lambda_m(t).$$

It is not required that the distribution of $X_n(\cdot; K)$, with K large enough, agrees with the distribution of X on dyadic points, although this is the case in our primary example of Brownian motion. We abuse notation by re-using notation such as N_X and N_A throughout our discussion of TES, but the random variables are not the same as in the rest of the paper.

Our algorithm relies on three random times.We choose suitable positive functions a, ξ_0, ξ_1 and a positive constant γ; see Proposition 7.2 below for details.

1. N_X: for $k \geq N_X$ and $n \geq 1$,

$$\sup_{t\in T}\big|X_n(t) - X_n(t;k)\big| \leq \xi_1(k) + \xi_0(k)a(n) \tag{7.1}$$

and, for $n \geq N_X$,

$$\sup_{t\in T}\big|X_n(t)\big| \leq \big(a(0)\lambda_0 + \xi_1(1)\big) + \big(\lambda_0 + \xi_0(1)\big)a(n). \tag{7.2}$$

2. $N_A = N_A(\gamma)$: for $n \geq N_A$,

$$A_n \geq \gamma n,$$

and we sample N_A jointly with $(A_1, \ldots, A_{N_A})$ using **Algorithm S** in Section 4.

3. N_ξ: for $n \geq N_\xi$,

$$\begin{aligned}&\big(a(0)\lambda_0 + \xi_1(1)\big) + \big(\lambda_0 + \xi_0(1)\big)a(n) - \log(n\gamma)\\ &\quad\leq \inf_{t\in[0,1]} X_1(t, N_X) - \log(A_1) - \xi_1(N_X) - \xi_0(N_X)a(n).\end{aligned} \tag{7.3}$$

We will choose a such that $N_\xi < \infty$ almost surely.

Setting $N = \max(N_X, N_A, N_\xi)$, we have, for $t \in T$ and $n \geq N$,

$$\begin{aligned}-\log(A_n) + X_n(t) &\leq -\log A_n + \big(a(0)\lambda_0 + \xi_1(1)\big) + \big(\lambda_0 + \xi_0(1)\big)a(n)\\ &\leq -\log(n\gamma) + \big(a(0)\lambda_0 + \xi_1(1)\big) + \big(\lambda_0 + \xi_0(1)\big)a(n)\\ &\leq -\log(A_1) + \inf_{t\in[0,1]} X_1(t, N_X) - \xi_1(N_X) - \xi_0(N_X)a(n)\\ &\leq -\log(A_1) + \inf_{t\in[0,1]} X_1(t)\\ &\leq -\log(A_1) + X_1(t),\end{aligned}$$

and therefore, for $t \in T$,

$$\sup_{n\geq 1}\big\{-\log A_n + X_n(t) + \mu(t)\big\} = \max_{1\leq n\leq N}\big\{-\log A_n + X_n(t) + \mu(t)\big\}. \tag{7.4}$$

If we select an integer $K_\delta \geq N_X$ such that $\xi_1(K_\delta) + \xi_0(K_\delta)a(n) \leq \delta$, then

$$M_\delta(t) = \max_{1\leq n\leq N}\big\{-\log A_n + X_n(t; K_\delta) + \mu(t)\big\}$$

satisfies $\sup_{t\in T} |M(t) - M_\delta(t)| \leq \delta$.

It remains to explain how to simulate N_X jointly with $(X_1, \ldots, X_N)$ and how to construct ξ_0, and ξ_1. For this, we use a variant of the record-breaking technique, but we first need to discuss our assumptions on the $Z_{m,n}$'s.

7.3. Assumptions on the $Z_{m,n}$'s and an example

We introduce some assumptions on the distribution of Z in order to use our record-breaking algorithm. We write $\overline{F}(\cdot)$ for the right tail of the distribution of $|Z|$, that is $\overline{F}(t) = \mathbb{P}(|Z| > t)$ for $t \geq 0$. Assume that we can find: a bounded and nonincreasing function $\overline{H}(\cdot)$ on $[0, \infty)$, an easy-to-evaluate eventually nonincreasing function $\Gamma(\cdot)$ on $\mathbb{N}$, as well as some $\theta_0 > 0$, $b \in (0, 1)$, and $\rho > 0$ satisfying the following assumptions with

$$a(n) = \rho\big(\log(n+1)\big)^b :$$

(C1) For (m, n) satisfying $\theta(m, n) \geq \theta_0$, we have $\overline{F}(a(m) + a(n)) \leq \overline{H}(a(m))\overline{H}(a(n))$.
(C2) We have $\sum_{m=0}^{\infty} \overline{H}(a(m)) < \infty$.
(C3) For $r > \theta_0$, we have $1 > \Gamma(r) \geq \sum_{(m,n):\theta(m,n)>r} \overline{H}(a(m))\overline{H}(a(n))$.
(C4) We have $\sum_r r^{\varepsilon}\Gamma(r) < \infty$ for some $\varepsilon > 0$.

Assumptions (C1), (C2), and (C3) are needed to run the algorithm, and Assumption (C4) to bound moments of the computational complexity.

As an example, we now show that these assumptions are satisfied if X_n is Brownian motion. Similar constructions are possible for fractional Brownian motion (see [5]), but we do not work out the details here. First, $\Lambda_0(t) = tI(t \in [0, 1])$, $\Lambda_1(t) = 2tI(t \in [0, 1/2]) + 2(1-t)I(t \in (1/2, 1])$, $\alpha = 1/2$, and $\lambda_0 = \lambda' = 1$; see [24]. Second, the $Z_{m,n}$'s are i.i.d. standard Gaussian random variables and one can select $\overline{H}(t) = \phi(t)$, the standard normal density, so that we have Assumption (C1) for $\theta_0 = \inf\{\theta : a(\eta_m(\theta)) + a(\eta_n(\theta)) \geq 2\sqrt{2\pi}\}$ and (C2) is evident. Moreover, selecting any $\rho > 4$ and $b = 1/2$ allows us to satisfy Assumptions (C3) and (C4). Indeed, note that

$$\begin{aligned}
&\sum_{\theta(m,n)\geq r} \overline{H}\big(a(m)\big)\overline{H}\big(a(n)\big) \\
&\quad = \sum_{\theta(m,n)\geq r} \left(\frac{2}{\pi}\right) \exp\left(-\rho^2 \frac{\log(m+1) + \log(n+1)}{2}\right) \\
&\quad = \sum_{\theta(m,n)\geq r} \left(\frac{1}{(m+1)(n+1)}\right)^{\rho^2/2} \leq \sum_{\theta(m,n)\geq r} \left(\frac{1}{m+n}\right)^{\rho^2/2}.
\end{aligned}$$

The point (m, n) with $\theta(m, n) = r$ is one of the $\ell(r)$ points on the segment between $(\ell(r), 0)$ and $(1, \ell(r) - 1)$, where $\ell(r) = \lceil \sqrt{2r + 1/4} - 1/2 \rceil$. We therefore continue to bound as follows:

$$\sum_{k\geq \ell(r)} k^{1-\rho^2/2} \leq \int_{\ell(r)-1}^{\infty} x^{1-\rho^2/2}\, dx = \frac{1}{\rho^2/2 - 2}\big(\ell(r) - 1\big)^{2-\rho^2/2}.$$

Thus, in the Brownian case we can define $\Gamma(r)$ to be the right-hand side of the preceding display, so for instance any $\rho > 4$ implies Assumption (C4).

In the case when X_n is standard Brownian motion we have

$$\sum_{m=0}^{2^r-1} \lambda_m Z_{m,n} \Lambda_m(t) = X_n(t),$$

for every dyadic point $t = j2^{-r}$ with $j = 0, 1, \ldots, 2^r$. Therefore, once we fix any $\delta > 0$ (say $\delta = 1/2$), we can apply the previous strategy to obtain N and we can continue sampling $Z_{m,n}$ for $m \geq K_\delta$ if needed so that we can return

$$M(t) = \max_{1 \leq n \leq N} \left\{ -\log(A_n) + \sum_{m=0}^{2^r-1} \lambda_m Z_{m,n} \Lambda_m(t) \right\}.$$

Consequently, we conclude that at least in the Brownian case the procedure that we present here can be used to evaluate $\{M(j/2^r)\}_{j=0}^d$ with $d = 2^r$ exactly and with expected computational cost of order $O(d \cdot \mathbb{E}[N]) = O(d)$ – because $\mathbb{E}[N]$ does not depend on d and is finite; see Theorem 7.4 below.

7.4. Breaking records for the $Z_{m,n}$'s

Define $T_0 = 0$, and, for $k \geq 1$,

$$T_k = \inf\bigl\{\theta(m,n) > T_{k-1} : |Z_{m,n}| > a(m) + a(n)\bigr\}.$$

In this subsection, given some integer $\theta_0 \geq 0$, we develop a technique to sample the random set $\mathcal{T} = \{T_k : T_k < \infty\} \cap \{\theta_0 + 1, \ldots\}$ jointly with $(Z_{m,n})_{m \geq 0, n \geq 1}$. Indeed, given $\mathcal{T}$, the $Z_{m,n}$ are independent and have the following distributions. For $\theta(m,n) \leq \theta_0$, $Z_{m,n}$ has the nominal (unconditional) distribution. For $\theta(m,n) \in \mathcal{T}$, $Z_{m,n}$ has the conditional distribution of Z given $\{|Z| > a(m) + a(n)\}$, and if $\theta(m,n) \notin \mathcal{T}$, $Z_{m,n}$ has the conditional distribution of Z given $\{|Z| \leq a(m) + a(n)\}$.

We first note that that only finitely many T_k's are finite, so that we can once again apply a record breaking technique, based on the record-breaking epochs T_k. Indeed, applying Assumptions (C1) and (C2), we find that

$$\sum_{m,n} P\bigl(|Z_{m,n}| > a(m) + a(n)\bigr) \leq \sum_{m,n} \overline{H}\bigl(a(m)\bigr)\overline{H}\bigl(a(n)\bigr) = \left(\sum_m \overline{H}\bigl(a(m)\bigr)\right)^2 < \infty,$$

and the claim follows from the Borel–Cantelli lemma.

The function SAMPLERECORDSZ given below, which is directly adapted from Algorithm 2w in [6], allows one to sequentially sample the elements in $\{T_k : T_k < \infty\}$ jointly with the $Z_{m,n}$'s. The function SAMPLERECORDSZ takes as input θ_0 satisfying $\Gamma(\theta_0) < 1$.

Function SAMPLERECORDSZ(θ_0)**: Samples the set** $\mathcal{T} = \{T_k : T_k < \infty\} \cap \{\theta_0 + 1, \ldots\}$

Step 1: Initialize $G \leftarrow \theta_0$ and $\mathcal{T} \leftarrow [\,]$.
Step 2: $u \leftarrow 1, d \leftarrow 0$. $V \leftarrow U(0, 1)$.
Step 3: While $u > V > d$
Step 4: $\quad G \leftarrow G + 1$
Step 5: $\quad d \leftarrow \max(d, (1 - \Gamma(G)) \times u)$
Step 6: $\quad u \leftarrow \mathbb{P}(|Z| \le a(\eta_m(G)) + a(\eta_n(G))) \times u$
Step 7: EndWhile
Step 8: If $V \ge u$, then $\mathcal{T} \leftarrow [\mathcal{T}, G]$ and go to Step 2.
Step 9: If $V \le d$, stop and return $\mathcal{T}$.

The next proposition establishes that the output of the function SAMPLERECORDSZ has the desired distribution.

Proposition 7.1. *The output from* SAMPLERECORDSZ(θ_0) *is a sample of the set* $\mathcal{T} = \{T_k : T_k < \infty\} \cap \{\theta_0 + 1, \ldots\}$. *Moreover, we have* $\mathbb{E}[(\max(0, \sup \mathcal{T}))^\beta] < \infty$ *for some* $\beta > 1$.

Proof. For simplicity, we assume throughout this proof that $\theta_0 = 0$. For the first claim, it suffices to show that SAMPLERECORDSZ(0) returns $\mathcal{T} = \{T_k : T_k < \infty\} \cap \{1, 2, \ldots\}$ without bias. We write $T = T_1$.

In Steps 3 through 5, the algorithm iteratively constructs the sequences (u_j) and (d_j) given by

$$u_j = u_{j-1}\mathbb{P}\big(|Z| \le a\big(\eta_m(j)\big) + a\big(\eta_n(j)\big)\big), \qquad d_j = \max\big(d_{j-1}, u_{j-1}\big(1 - \Gamma(j)\big)\big)$$

with $u_0 = 1$ and $d_0 = 0$. It is evident that both sequences are monotone. Moreover, we have $u_j = \mathbb{P}(T > j)$ for $j \ge 0$ and $\lim_{j\to\infty} u_j = \mathbb{P}(T = \infty)$. Similarly, because $\lim_{j\to\infty} \Gamma(j) = 0$ we obtain $\lim_{j\to\infty} d_j = \mathbb{P}(T = \infty)$.

Let $n(V)$ be the number of times Step 3 is executed before either going to Step 8 or Step 9. It suffices to check that when Step 8 is executed then the element added to $\mathcal{T}$ has the law of T given $T < \infty$, and that Step 9 is executed with probability $\mathbb{P}(T = \infty)$. For the former, we note that by definition of $n(V)$ and because $u_j \in (d_{j-1}, u_j)$, we have for $j \ge 1$

$$\begin{aligned}\mathbb{P}\big(n(V) = j | V \ge u_{n(V)}\big) &= \frac{\mathbb{P}(V \in (d_{j-1}, u_{j-1}), V \ge u_j)}{\mathbb{P}(V \ge u_{n(V)})} \\ &= \frac{\mathbb{P}(V \in (u_j, u_{j-1}))}{\mathbb{P}(V \ge u_{n(V)})} \\ &= \frac{u_{j-1} - u_j}{1 - \lim_{k\to\infty} u_k},\end{aligned}$$

which equals $\mathbb{P}(T = j | T < \infty)$ as desired. For the latter, we note that

$$\mathbb{P}(V \le d_{n(V)}) = 1 - \mathbb{P}(V \ge u_{n(V)}) = \mathbb{P}(T = \infty).$$

In preparation for the proof of the second claim of the proposition, we bound the probability that the while loop requires more than $k \geq 1$ iterations:

$$\begin{aligned}\mathbb{P}\big(n(V) > k\big) &\leq \mathbb{P}\big(V \in (d_k, u_k)\big) \leq \mathbb{P}\big(V \in (d_k, u_{k-1})\big) = u_{k-1} - d_k \\ &= u_{k-1} - \max\big\{d_{k-1}, u_{k-1}\big(1 - \Gamma(k)\big)\big\} \leq \Gamma(k).\end{aligned}$$

As a consequence of the inequality

$$\mathbb{E}\big[n(V)^\beta\big] = \sum_{k=0}^{\infty}\big((k+1)^\beta - k^\beta\big)\mathbb{P}\big(n(V) > k\big) \leq 1 + \sum_{k=1}^{\infty}\big((k+1)^\beta - k^\beta\big)\Gamma(k),$$

we find that $\mathbb{E}[n(V)^\beta] < \infty$ if $\sum_k k^{\beta-1}\Gamma(k) < \infty$.

We have a similar finite-moment bound for subsequent calls to the while loop. Writing $n_i(V_1, \ldots, V_i)$ for the number of iterations in the ith execution of the while loop, where $V_1, V_2, \ldots$ are the i.i.d. standard uniform random variables generated in subsequent calls to Step 2. Compared to the above argument for $i = 1$, this quantity only depends on $V_1, \ldots, V_{i-1}$ through a random shift of Γ. Because Γ is eventually nonincreasing, there exists a constant c' such that, for all $i \geq 1$,

$$\mathbb{E}\big[n_i(V_1, \ldots, V_i)^\beta | V_1, \ldots, V_{i-1}\big] \leq c' \sum_k k^{\beta-1}\Gamma(k). \tag{7.5}$$

To prove a bound on the moment of $\sup \mathcal{T}$, we first let Υ be the number of times we execute the while loop. We then note that, for any random variable G and any $\beta \geq 1$, by Jensen's inequality,

$$\begin{aligned}\max(0, \sup \mathcal{T})^\beta &= \left(\sum_{i=1}^{\Upsilon-1} n_i(V_1, \ldots, V_i)\right)^\beta = \left(\sum_{i=1}^{\infty} n_i(V_1, \ldots, V_i) I(\Upsilon > i)\right)^\beta \\ &\leq \sum_{i=1}^{\infty}\left(\frac{n_i(V_1, \ldots, V_i) I(\Upsilon > i)}{\mathbb{P}(G = i)}\right)^\beta \mathbb{P}(G = i) \\ &= \sum_{i=1}^{\infty} n_i(V_1, \ldots, V_i)^\beta I(\Upsilon > i)\mathbb{P}(G = i)^{1-\beta},\end{aligned}$$

because the right-hand side is finite almost surely.

Because the event $\{\Upsilon > i - 1\}$ only depends on $V_1, \ldots, V_{i-1}$, we have by (7.5),

$$\begin{aligned}\mathbb{E}\big[n_i(V_1, \ldots, V_i)^\beta I(\Upsilon > i)\big] &\leq \mathbb{E}\big[n_i(V_1, \ldots, V_i)^\beta I(\Upsilon > i - 1)\big] \\ &= \mathbb{E}\big[I(\Upsilon > i - 1)\mathbb{E}\big[n_i(V_1, \ldots, V_i)^\beta | V_1, \ldots, V_{i-1}\big]\big] \\ &\leq c'\left(\sum_k k^{\beta-1}\Gamma(k)\right) P(\Upsilon > i - 1) \\ &\leq c'\left(\sum_k k^{\beta-1}\Gamma(k)\right) P(T_1 < \infty)^{i-1},\end{aligned}$$

where we use the fact that Υ is stochastically dominated by a geometric random variable with success parameter $P(T_1 = \infty) > 0$. Combining the preceding displays, we deduce that, for $\beta \geq 1$,

$$\mathbb{E}\big[\max(0, \sup \mathcal{T})^{\beta}\big] \leq c'\bigg(\sum_{k} k^{\beta-1}\Gamma(k)\bigg)\sum_{i=1}^{\infty} P(T_1 < \infty)^{i-1}\mathbb{P}(G = i)^{1-\beta},$$

which is seen to be finite for some $\beta > 1$ by Assumption (C4) upon choosing G geometric with a suitably chosen success probability. □

7.5. Truncation error of the infinite series

We next write, for $k \geq 0$

$$X_n(t) = X_n(t; k) + \sum_{m>k} \lambda_m Z_{m,n} \Lambda_m(t),$$

and it is our objective to study the truncation error, i.e., the second term.

The next proposition controls the truncation error in terms of functions ξ_0 and ξ_1 defined for $r \geq 1$ through

$$\xi_0(r) = \lambda'\big(1 - 2^{-\alpha}\big)^{-1} 2^{-\alpha\lfloor \log_2(r)\rfloor},$$
$$\xi_1(r) = \frac{\rho}{\log_2(e)}\bigg(\big\lfloor \log_2(r)\big\rfloor + \frac{2^{-\alpha}}{1 - 2^{-\alpha}} + 2\bigg)\xi_0(r).$$

Note that $\xi_0(r), \xi_1(r) \to 0$ as $r \to \infty$. We also write

$$N_X = \max\{\sup \mathcal{T}, \theta_0 - 1\}.$$

If $\mathcal{T}$ is empty, then $\sup \mathcal{T} = -\infty$ and therefore $N_X = \theta_0 - 1$; otherwise, if $\mathcal{T}$ is non-empty, then $\sup \mathcal{T} \geq \theta_0$ and therefore $N_X \geq \theta_0$.

Proposition 7.2. *For all $k \geq N_X$ and $n \geq 1$, we have* (7.1), *and for all $n \geq N_X$,* (7.2).

Proof. We observe that

$$\big|X_n(t) - X_n(t; k)\big| \leq \sum_{m>k} \lambda_m a(m)\big|\Lambda_m(t)\big| + a(n)\sum_{m>k} \lambda_m \big|\Lambda_m(t)\big|.$$

If $m > k \geq N_X$, because $\theta(m, n) \geq m$, we have from the definition of N_X, that

$$\big|\lambda_m Z_{m,n} \Lambda_m(t)\big| \leq \lambda_m\big(a(m) + a(n)\big)\big|\Lambda_m(t)\big|.$$

We bound the summand of the second sum by noting that, for $r \geq 1$,

$$\sup_{t\in T}\sum_{m=r}^{\infty}\lambda_m\bigl|\Lambda_m(t)\bigr| \leq \sup_{t\in T}\sum_{j=\lfloor\log_2(r)\rfloor}^{\infty}\sum_{k=0}^{2^j-1}\lambda_{2^j}\bigl|\Lambda_{2^j+k}(t)\bigr| \leq \sum_{j=\lfloor\log_2(r)\rfloor}^{\infty}\lambda' 2^{-\alpha j} = \xi_0(r).$$

We similarly bound the summand in the first sum, using the definition of $a(\cdot)$ and the fact that

$$\sum_{j=k}^{\infty} j s^j = s^k\frac{k(1-s)+s}{(1-s)^2}$$

for $|s| < 1$. These bounds establish (7.1).

Now we turn to the proof of (7.2). For $n \geq N_X$,

$$\bigl|X_n(t)\bigr| \leq \sum_{m=0}^{\infty}\bigl|\lambda_m Z_{m,n}\Lambda_m(t)\bigr| \leq \bigl(a(0)+a(n)\bigr)\lambda_0 + \sum_{m=1}^{\infty}\lambda_m\bigl(a(m)+a(n)\bigr)\bigl|\Lambda_m(t)\bigr|,$$

because $\theta(m,n) \geq N_X$ for each $n \geq N_X$. The sum over m is bounded by $\xi_1(1) + \xi_0(1)a(n)$ as shown in the proof of (7.1). □

7.6. Construction of M_δ

Now we are ready to provide the final algorithm for computing M_δ.

Algorithm TES: Samples M_δ given $\delta > 0$.

Step 1: $\mathcal{T} \leftarrow$ Sample SAMPLERECORDSZ(θ_0)
Step 2: $N_X \leftarrow \max\{\sup\mathcal{T}, \theta_0 - 1\}$
Step 3: Sample $Z_{m,n}$ from the nominal distribution if $\theta(m,n) \leq \theta_0$
Step 4: For $0 \leq m \leq N_X$ and $\theta(m,1) > \theta_0$
Step 5: If $\theta(m,1) \in \mathcal{T}$: sample $Z_{m,1}$ from the law of Z given $\{|Z| > a(m) + a(1)\}$
Step 6: Else If: sample $Z_{m,1}$ from the law of Z given $\{|Z| \leq a(m) + a(1)\}$
Step 7: EndFor
Step 8: Sample $A_1, \ldots, A_{N_A}$ using Steps 1–8 from **Algorithm S** with $S_n = \gamma n - A_n$.
Step 9: Compute N_ξ, the smallest n for which (7.3) holds, and let $N \leftarrow \max(N_X, N_A, N_\xi)$
Step 10: Sample $A_{N_A+1}, \ldots, A_N$ as in Step 10 from **Algorithm S** with $S_n = \gamma n - A_n$.
Step 11: Compute the smallest $K_\delta \geq N_X$ such that $\xi_1(K_\delta) + \xi_0(K_\delta)a(N) \leq \delta$.
Step 12: For $2 \leq n \leq N$, $0 \leq m \leq K_\delta$, $\theta(m,n) > \theta_0$ and also for $n = 1$, $N_X < m \leq K_\delta$, $\theta(m,n) > \theta_0$
Step 13: If $\theta(m,n) \in \mathcal{T}$: sample $Z_{m,n}$ from the law of Z given $\{|Z| > a(m) + a(n)\}$
Step 14: Else: sample $Z_{m,n}$ from the law of Z given $\{|Z| \leq a(m) + a(n)\}$
Step 15: EndFor
Step 16: Return $M_\delta(t) = \max\{X_n(t; K_\delta) - \log(A_n)\}$.

7.7. Exponential moments of $\sup_{t\in[0,1]}|X(t)|$

We need a bound on the exponential moments of $\sup_{t\in[0,1]}|X(t)|$ in order to analyze N_ξ. If X is Gaussian and continuous, then such a bound immediately follows from Borell's inequality [1], Thm. 2.1.1. The following proposition establishes the existence of exponential moments in the generality of the present section.

Proposition 7.3. *For any $p > 0$, we have*

$$\mathbb{E}\exp\Big(p\sup_{t\in[0,1]}\big|X(t)\big|\Big) < \infty.$$

Proof. We first note that

$$\sup_{t\in[0,1]}\big|X_n(t)\big| \le \lambda_0 Z_{0,n} + \sum_{j=1}^{\infty}\lambda' 2^{-\alpha j}\max_{k=0,\dots,2^j-1}|Z_{2^j+k,n}|.$$

It suffices to prove that the tail of the infinite sum in this expression is ultimately lighter than any exponential. A union bound leads to, for $y \ge 0$,

$$\begin{aligned}
&\mathbb{P}\left(\sum_{j=1}^{\infty}\lambda' 2^{-\alpha j}\max_{k=0,\dots,2^j-1}|Z_{2^j+k,n}| > y\right)\\
&\le \sum_{j=1}^{\infty}\mathbb{P}\Big(\lambda' 2^{-\alpha j}\max_{k=0,\dots,2^j-1}|Z_{2^j+k,n}| > \big(2^{\alpha/2}-1\big)2^{-\alpha j/2}y\Big)\\
&\le \sum_{j=1}^{\infty}\mathbb{P}\left(\max_{k=0,\dots,2^j-1}|Z_{2^j+k,n}| > \frac{(2^{\alpha/2}-1)2^{\alpha j/2}}{\lambda'}y\right).
\end{aligned}$$

Assumptions (C1) and (C2) imply that $C' := \mathbb{E}\exp(|Z/\rho|^{1/b}) < \infty$ and therefore we have by Markov's inequality, for $t \ge 0$,

$$\mathbb{P}\Big(\max_{k=0,\dots,2^j-1}|Z_{2^j+k,n}| > 2^{\alpha j/2}t\Big) \le 2^j\mathbb{P}\big(|Z| > 2^{\alpha j/2}t\big) \le C'2^j e^{-(t2^{\alpha j/2}/\rho)^{1/b}}.$$

Select some $t_0 > 0$ and $\kappa \in (1, 1/b)$ such that $(t2^{\alpha j/2}/\rho)^{1/b} \ge j + t^\kappa$ for all $j \ge 1$ and $t \ge t_0$. Using this bound results in a tail estimate that is summable over j and lighter than any exponential distribution. □

7.8. Complexity analysis

We conclude this section with the following result which summarizes the performance guarantee of **Algorithm TES**. Higher moment bounds on the computational costs are readily found using the same arguments and a stronger version of Assumption (C4).

Theorem 7.4. *Assume that the conditions* (A1), (A2), (C1)–(C4) *are in force. Given* $\delta \in (0,1)$, *the output* $(M_\delta(t))_{t\in T}$ *of* ***Algorithm TES*** *satisfies*

$$\sup_{t\in T}\big|M_\delta(t) - M(t)\big| \le \delta.$$

Moreover, we have

$$\mathbb{E}[K_\delta] = O\big(\big(\delta/\log(1/\delta)\big)^{-1/\alpha}\big),$$

where α *is determined by the series representation of* X. *Finally, the total computational costs of running* ***Algorithm TES*** *has expectation at most* $O((\delta/\log(1/\delta))^{-1/\alpha})$.

Proof. The first claim follows by construction, see Section 7.2.

From Proposition 7.1, we have $\mathbb{E}[N_X^\beta] < \infty$ for some $\beta > 1$. In order to analyze N_ξ, we use Proposition 7.3. In fact, N_ξ only has to be sufficiently large so that we have

$$\big(\lambda_0 + \xi_0(1) + \xi_0(N_X)\big)\rho\big(\log(n+1)\big)^b < \frac{1}{2}\log n$$

and

$$-\frac{1}{2}\log n \le \inf_t X_1(t, N_X) - \log A_1 - a(0)\lambda_0 - \xi_1(1) - \xi_1(N_X) + \log\gamma$$

for any $n \ge N_\xi$. With simple calculations, it follows from Proposition 7.3 and Assumption (A1) that $\mathbb{E}[N_\xi^p] < \infty$ for every $p > 0$. We have argued in Section 6 that $\mathbb{E}[N_A^p] < \infty$, so we conclude that $\mathbb{E}[N^\beta] < \infty$. Finally, using the definition of $\xi_0(r)$ and $\xi_1(r)$ we can see that it there is a constant $\kappa > 0$ such that

$$K_\delta = O\left(\left[\frac{\delta}{(\log N)^b + \kappa\log(1/\delta)}\right]^{-1/\alpha}\right).$$

This leads to the bound on the first moment of K_δ. The expected running time of the algorithm is order $\mathbb{E}[K_\delta \times N]$, which is finite because $\mathbb{E}[N^\beta] < \infty$. The complexity bound follows. □

8. Numerical results

In this section, we show some simulation results to empirically validate **Algorithm M**. We also compare numerically the computational cost of our record-breaking method, noted as RB in the following charts, with the existing exact sampling algorithm developed in [13] by Dieker and Mikosch (DM) and the exact simulation algorithm using extremal function proposed in [14] (EF). We implemented all three algorithms in Matlab. For our algorithm, we choose the values of a and C according to our discussion in Section 6.2. We let $C = 0$, then choose the largest $a \in (0,1)$ such that (6.2) holds.

We generated the Brown–Resnick processes, $M(t) = \sup_{n\ge1}\{-\log A_n + X_n(t) - \sigma^2(t)/2\}$, on compact sets. If X is a Brownian motion it was shown in [10] that M has a stationary sample

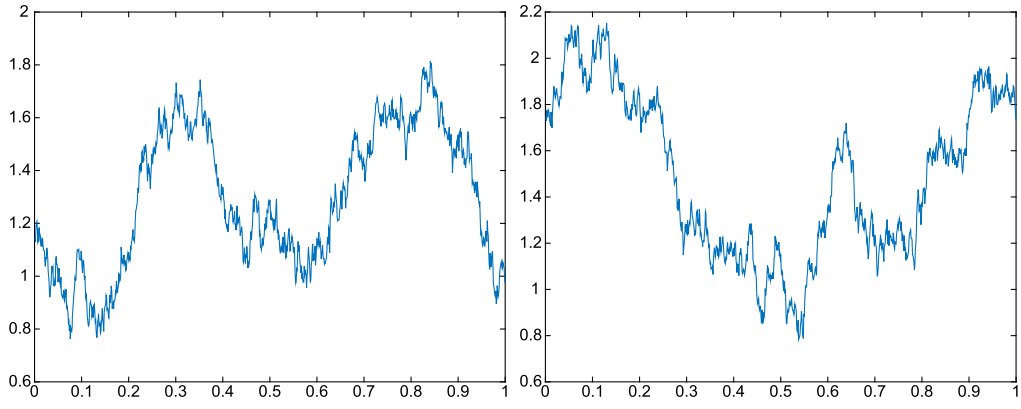

Figure 1. The Brown–Resnick process on [0, 1] with Brownian motion input. The grid mesh is 0.001.

path on [0, 1]. Figure 1 shows sample paths of M in this case. Figure 2 presents two samples of the Brown–Resnick random field on $[0, 1]^2$ when X is a Brownian sheet.

Next, we will compare the computational cost in CPU time of our algorithm with the algorithm proposed in [13]. We conducted both algorithms to generate 200 samples of the Brown–Resnick process M with fractional Brownian motion inputs. We recorded both the average CPU time for generating a single sample and the 95% confidence interval for the mean based on our 200 samples, for different grid numbers $d = 1000, 2000, 5000$ and $10\,000$, and with different Hurst parameters $H \in \{1/4, 1/2, 3/4\}$ of the fractional Brownian motion. The sample estimates and the 95% confidence intervals for the mean CPU times to generate a single sample are shown in Table 1. They illustrate that when the number of grids increases, the computational cost of our algorithm appears to increase almost linearly, while the cost for the algorithm proposed in

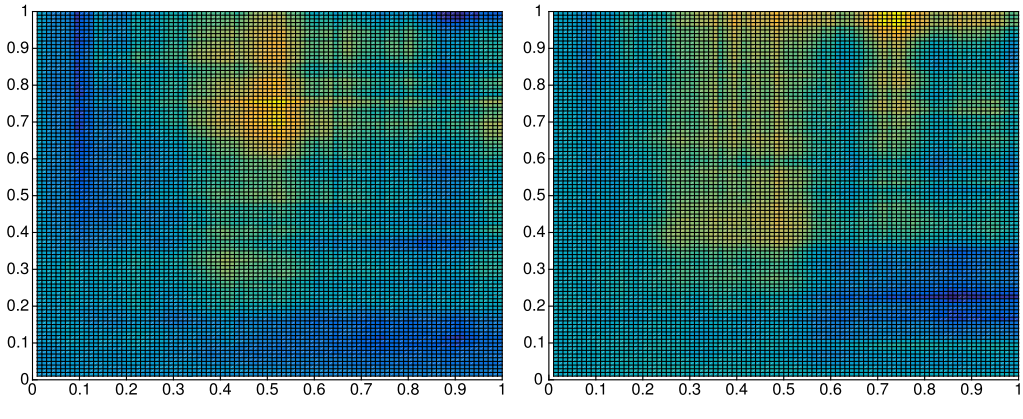

Figure 2. The Brown–Resnick field on $[0, 1]^2$ with Brownian sheet input. The grid mesh is 0.001.

Table 1. Comparison of running time of our algorithm (RB) vs. [13] (DM) vs. [14] (EF)

d	$H = 1/4$	$H = 1/2$	$H = 3/4$
Average cost per sample (second) (RB) ($\pm$ half-width of confidence interval)			
1000	0.03 ± 0.003	0.03 ± 0.002	0.03 ± 0.001
2000	0.08 ± 0.020	0.06 ± 0.007	0.06 ± 0.002
5000	0.19 ± 0.071	0.13 ± 0.004	0.13 ± 0.008
10 000	0.32 ± 0.027	0.26 ± 0.009	0.27 ± 0.008
Average cost per sample (second) (DM)			
1000	0.40 ± 0.04	0.28 ± 0.03	0.43 ± 0.05
2000	1.23 ± 0.13	1.00 ± 0.13	1.37 ± 0.15
5000	7.32 ± 0.88	4.82 ± 0.67	5.97 ± 0.79
10 000	28.98 ± 3.18	21.42 ± 2.64	19.14 ± 2.67
Average cost per sample (second) (EF)			
1000	0.15 ± 0.02	0.13 ± 0.02	0.15 ± 0.02
2000	0.49 ± 0.06	0.46 ± 0.05	0.66 ± 0.09
5000	2.83 ± 0.32	2.34 ± 0.28	3.39 ± 0.43
10 000	10.81 ± 1.46	9.67 ± 1.24	12.17 ± 1.70

[13] increases quadratically. Because we are using the circulant embedding method to generate the fractional Brownian vectors, which has a complexity of order $O(d \log d)$, it is consistent with expectations. It is worth noting that for this method the computational cost to generate a d-dimensional Gaussian vector is the same as for generating a $2^{\lceil \log_2 d \rceil}$-dimensional Gaussian vector. However, this consideration will not affect our comparison because we used this method in both algorithms.

Next, we compare the number of Gaussian vectors generated in our algorithm with the algorithms of [13] and [14]. We generate samples of the Brown–Resnick process with fractional Brownian motion generator, with $H = 3/4$. We used the grid numbers $d = 1000, 3000, 5000, 7000, 9000$. To get comparable relative error, we simulate 1000 times for algorithms DM and EF, and 10 000 times for RB. We calculated the sample average of the number of Gaussian vectors

Table 2. Comparison of number of Gaussian vectors generated in our algorithm (RB) v.s. [13] (DM) v.s. [14] (EF), $H = 3/4$

	Number of Gaussian vectors		
d	RB	DM	EF
1000	29.5 ± 2.0	1522.1 ± 83.3	1040.4 ± 60.5
3000	28.7 ± 2.1	4440.1 ± 248.9	3101.1 ± 194.2
5000	32.5 ± 4.2	7648.0 ± 436.3	5056.3 ± 298.2
7000	31.4 ± 2.9	$10\,642.0 \pm 638.4$	6961.4 ± 423.8
9000	26.5 ± 1.5	$13\,570.0 \pm 796.1$	8886.6 ± 510.3

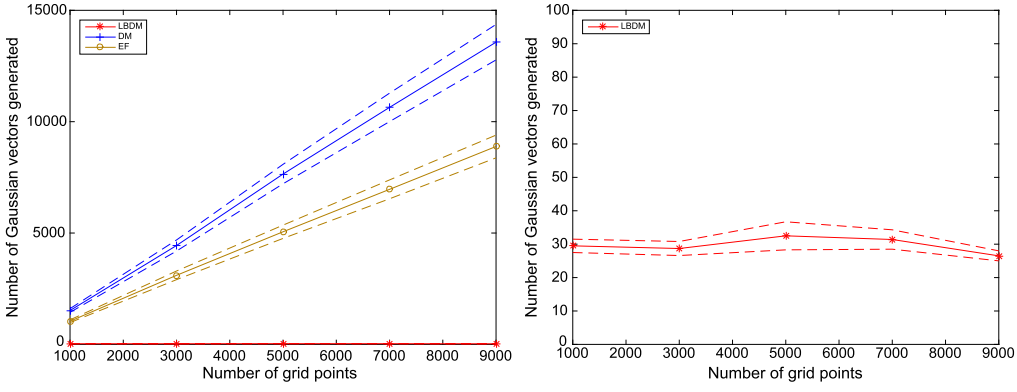

Figure 3. Comparison of number of Gaussian vectors generated in our algorithm (RB) v.s. [13] (DM) v.s. [14] (EF), $H = 3/4$.

generated in each of the algorithms, and the 95% confidence bounds. Table 2 illustrates our main result. On the left, Figure 3 exhibits the plot corresponding to Table 2 for all three algorithms. On the right, Figure 3 focuses on the algorithm RB. The number of Gaussian vectors generated increases linearly in both the algorithms of [13] and [14], with a reduction of constant factor using the extremal function algorithm from [14]. In our algorithm this number stays roughly at the same level.

Acknowledgements

The authors are grateful for the careful reading and comments of the editors and the referees, all of which significantly improved the quality of our paper. Support from NSF grant DMS-132055, NSF grant DMS-1838576 and NSF grant CMMI-1538217 is gratefully acknowledged by J. Blanchet. Thomas Mikosch was partly supported by an Alexander von Humboldt Research Award. A.B. Dieker gratefully acknowledges support from NSF grant CMMI-1252878.

References

[1] Adler, R.J. and Taylor, J.E. (2007). *Random Fields and Geometry*. *Springer Monographs in Mathematics*. New York: Springer. MR2319516

[2] Asmussen, S. (2003). *Applied Probability and Queues*: *Stochastic Modelling and Applied Probability*, 2nd ed. *Applications of Mathematics* (*New York*) **51**. New York: Springer. MR1978607

[3] Asmussen, S., Glynn, P. and Pitman, J. (1995). Discretization error in simulation of one-dimensional reflecting Brownian motion. *Ann*. *Appl*. *Probab*. **5** 875–896. MR1384357

[4] Asmussen, S. and Glynn, P.W. (2007). *Stochastic Simulation*: *Algorithms and Analysis*. *Stochastic Modelling and Applied Probability* **57**. New York: Springer. MR2331321

[5] Ayache, A. and Taqqu, M.S. (2003). Rate optimality of wavelet series approximations of fractional Brownian motion. *J*. *Fourier Anal*. *Appl*. **9** 451–471. MR2027888

[6] Blanchet, J. and Chen, X. (2015). Steady-state simulation of reflected Brownian motion and related stochastic networks. *Ann. Appl. Probab.* **25** 3209–3250. MR3404635
[7] Blanchet, J., Chen, X. and Dong, J. (2017). ε-strong simulation for multidimensional stochastic differential equations via rough path analysis. *Ann. Appl. Probab.* **27** 275–336. MR3619789
[8] Blanchet, J. and Wallwater, A. (2015). Exact sampling of stationary and time-reversed queues. *ACM Trans. Model. Comput. Simul.* **25** Art. 26. MR3432614
[9] Blanchet, J.H. and Sigman, K. (2011). On exact sampling of stochastic perpetuities. *J. Appl. Probab.* **48A** 165–182. MR2865624
[10] Brown, B.M. and Resnick, S.I. (1977). Extreme values of independent stochastic processes. *J. Appl. Probab.* **14** 732–739. MR0517438
[11] de Haan, L. (1984). A spectral representation for max-stable processes. *Ann. Probab.* **12** 1194–1204. MR0757776
[12] de Haan, L. and Zhou, C. (2008). On extreme value analysis of a spatial process. *REVSTAT* **6** 71–81. MR2386300
[13] Dieker, A.B. and Mikosch, T. (2015). Exact simulation of Brown–Resnick random fields at a finite number of locations. *Extremes* **18** 301–314. MR3351818
[14] Dombry, C., Engelke, S. and Oesting, M. (2016). Exact simulation of max-stable processes. *Biometrika* **103** 303–317. Available at arXiv:1506.04430. MR3509888
[15] Embrechts, P., Klüppelberg, C. and Mikosch, T. (1997). *Modelling Extremal Events*: *For Insurance and Finance*. *Applications of Mathematics* (*New York*) **33**. Berlin: Springer. MR1458613
[16] Gut, A. (2009). *Stopped Random Walks*: *Limit Theorems and Applications*, 2nd ed. *Springer Series in Operations Research and Financial Engineering*. New York: Springer. MR2489436
[17] Kabluchko, Z., Schlather, M. and de Haan, L. (2009). Stationary max-stable fields associated to negative definite functions. *Ann. Probab.* **37** 2042–2065. MR2561440
[18] Kenealy, B. (2013, August 11). New York's MTA buys $200 million cat bond to avoid storm surge losses. *Bus. Insur.* Available at https://www.businessinsurance.com.
[19] Oesting, M., Schlather, M. and Zhou, C. (2018). Exact and fast simulation of max-stable processes on a compact set using the normalized spectral representation. *Bernoulli* **24** 1497–1530. MR3706800
[20] Pollock, M., Johansen, A.M. and Roberts, G.O. (2016). On the exact and ε-strong simulation of (jump) diffusions. *Bernoulli* **22** 794–856. MR3449801
[21] Schilling, R.L. and Partzsch, L. (2012). *Brownian Motion*: *An Introduction to Stochastic Processes*. Berlin: de Gruyter. With a chapter on simulation by Björn Böttcher. MR2962168
[22] Schlather, M. (2002). Models for stationary max-stable random fields. *Extremes* **5** 33–44. MR1947786
[23] Smith, R.L. (1990). Max-stable processes and spatial extremes. Unpublished manuscript.
[24] Steele, J.M. (2001). *Stochastic Calculus and Financial Applications*. *Applications of Mathematics* (*New York*) **45**. New York: Springer. MR1783083
[25] Thibaud, E., Aalto, J., Cooley, D.S., Davison, A.C. and Heikkinen, J. (2016). Bayesian inference for the Brown–Resnick process, with an application to extreme low temperatures. *Ann. Appl. Stat.* **10** 2303–2324. Available at arXiv:1506.07836. MR3592058

Received September 2016 and revised September 2018