

**Manuscript version: Author's Accepted Manuscript**

The version presented in WRAP is the author's accepted manuscript and may differ from the published version or Version of Record.

**Persistent WRAP URL:**

<http://wrap.warwick.ac.uk/123272>

**How to cite:**

Please refer to published version for the most recent bibliographic citation information. If a published version is known of, the repository item page linked to above, will contain details on accessing it.

**Copyright and reuse:**

The Warwick Research Archive Portal (WRAP) makes this work by researchers of the University of Warwick available open access under the following conditions.

Copyright © and all moral rights to the version of the paper presented here belong to the individual author(s) and/or other copyright owners. To the extent reasonable and practicable the material made available in WRAP has been checked for eligibility before being made available.

Copies of full items can be used for personal research or study, educational, or not-for-profit purposes without prior permission or charge. Provided that the authors, title and full bibliographic details are credited, a hyperlink and/or URL is given for the original metadata page and the content is not changed in any way.

**Publisher's statement:**

Please refer to the repository item page, publisher's statement section, for further information.

For more information, please contact the WRAP Team at: [wrap@warwick.ac.uk](mailto:wrap@warwick.ac.uk).

## EXACT SIMULATION OF THE EXTREMA OF STABLE PROCESSES

JORGE I. GONZÁLEZ CÁZARES,\* \*\* *University of Warwick & The Alan Turing Institute*

ALEKSANDAR MIJATOVIĆ,\* \*\*\* *University of Warwick & The Alan Turing Institute*

GERÓNIMO URIBE BRAVO,\*\*\*\* *Universidad Nacional Autónoma de México*

### Abstract

We exhibit an exact simulation algorithm for the supremum of a stable process over a finite time interval using dominated coupling from the past (DCFTP). We establish a novel perpetuity equation for the supremum (via the representation of the concave majorants of Lévy processes [27]) and apply it to construct a Markov chain in the DCFTP algorithm. We prove that the number of steps taken backwards in time before the coalescence is detected is finite. We analyse numerically the performance of the algorithm (the code, written in Julia 1.0, is available on GitHub).

*Keywords:* random variate generation; perpetuities; simulation; perfect simulation; dominated coupling from the past; stable process

2010 Mathematics Subject Classification: Primary 65C10

Secondary 65C05

### 1. Introduction

This paper describes an algorithm for generating *exact* samples of the extrema of a stable process (see Algorithm 1 below) based on dominated coupling from the past (DCFTP), a coupling method for exact simulation from an invariant distribution of a Markov chain on an ordered state space (cf. [21] and the references therein). The chain in Algorithm 1 is based on a novel characterisation

---

\* Postal address: The Alan Turing Institute, British Library, 96 Euston Road, London NW1 2DB, UK & University of Warwick, Coventry CV4 7AL, UK

\*\* Email address: jorge.gonzalez-cazares@warwick.ac.uk

\*\*\* Email address: a.mijatovic@warwick.ac.uk

\*\*\*\* Postal address: Instituto de Matemáticas, Ciudad Universitaria, Coyoacán, Ciudad de México, 04510, México

\*\*\*\* Email address: geronimo@matem.unam.mx

for the law of the supremum of a stable process at a fixed time in Theorem 1.1. Perpetuity (1.1) is established via the stochastic representation for concave majorants of Lévy processes [27] and the scaling property of stable laws (see Section 2 below for the proof of Theorem 1.1).

**Theorem 1.1.** *Let  $Y = (Y_t)_{t \in [0, \infty)}$  be a stable process with the stability and positivity parameters  $\alpha$  and  $\rho$ , respectively (see Appendix A). Define  $\bar{Y}_1 = \sup_{s \in [0, 1]} Y_s$  and let  $(B, U, V, S, \bar{Y}_1)$  be a random vector with independent components, where  $U, V$  are uniform on  $(0, 1)$ ,  $B$  is Bernoulli with parameter  $1 - \rho$  and  $S$  has the law of  $Y_1$  conditioned on being positive. Then the following equality in law holds:*

$$\bar{Y}_1 \stackrel{d}{=} \Lambda^{\frac{1}{\alpha}} \left( U^{\frac{1}{\alpha}} \bar{Y}_1 + (1 - U)^{\frac{1}{\alpha}} S \right), \quad (1.1)$$

where  $\Lambda = 1 + B(V^{\frac{1}{\rho}} - 1)$ . Furthermore, the law of  $\bar{Y}_1$  is the unique solution to (1.1).

The universality of stable processes makes them ubiquitous in probability theory and many areas of statistics and natural and social sciences (see the monograph [30] and the references therein). The problem of efficient simulation of stable random variables in the context of statistics was addressed in [14]. Among the path properties, the running supremum  $\bar{Y}_t = \sup_{s \in [0, t]} Y_s \stackrel{d}{=} t^{1/\alpha} \bar{Y}_1$  of a stable process is of special interest (cf. [29, 1, 22, 16]) as it arises in application areas such as optimal stopping, the prediction of the ultimate supremum and risk theory (cf. [2, 29]).

In general, one has no access to the density, distribution or even characteristic function of  $\bar{Y}_1$ , making a rejection sampling algorithm (see [11, Sec. II.3]) for  $\bar{Y}_1$  difficult to construct. More precisely, if  $Y$  has no positive jumps, the strong Markov property and the fact that  $Y$  does not jump over positive levels imply that  $\bar{Y}_1$  has the same law as  $Y_1$  conditioned on being positive [25]. In all other cases, the law of  $\bar{Y}_1$  is not accessible in closed form and the information about it in the literature is obtained via analytical methods based on the Wiener-Hopf factorisation. If  $Y$  has no negative jumps, [1] gives an alternating series expression for the density, while [22, 16] give a double series representation for a dense class of parameters. The coefficients in these representations are complicated and it is not immediately clear how one could use them to design a simulation algorithm. Moreover, in the general case, when  $\alpha$  is rational the series representation is proved to be convergent for finitely many  $\rho$  only [23]. Our simulation algorithm is based on purely probabilistic methods (it may be regarded as a generalization of the exact simulation algorithm for Vervaat perpetuities in [18]) and as such covers the entire class of stable processes.

### 1.1. Exact Simulation Algorithm

The perpetuity in (1.1) above gives rise to an update function  $x \mapsto \phi(x, \Theta)$  of a Markov chain on  $(0, \infty)$ , where the components of the random vector  $\Theta$  are the random variables in Theorem 1.1 (see (3.1) below for the precise definition of  $\phi$ ). The invariant distribution (i.e. invariant probability measure as defined in [24, p. 229]) for the chain  $X' = \{X'_n\}_{n \in \mathbb{Z}}$ , defined by  $X'_n = \phi(X'_{n-1}, \Theta_{n-1})$  with  $\{\Theta_n\}_{n \in \mathbb{Z}}$  a sequence of independent copies of  $\Theta$ , equals that of  $\bar{Y}_1$ . However, since  $x \mapsto \phi(x, \Theta)$  is strictly increasing in  $x$  with probability one, no coalescence occurs, making  $X'$  unusable for DCFTP purposes. Fortunately, the structure of the perpetuity in (1.1) is such that the update function  $\phi$  can be modified to a *multigamma coupler* [26]  $x \mapsto \psi(x, \Theta)$ , which is constant on a subinterval in  $(0, \infty)$  with positive probability and globally non-decreasing. The definition of  $\psi$ , given in Lemma 3.1 below, was inspired by [18] where such a modification was applied to Vervaat perpetuities. The construction requires an addition of a single independent uniform random variable to the vector  $\Theta$  and yields a Markov chain  $X = \{X_n\}_{n \in \mathbb{Z}}$  on  $(0, \infty)$  via  $X_n = \psi(X_{n-1}, \Theta_{n-1})$ , where  $\{\Theta_n\}_{n \in \mathbb{Z}}$  are independent copies of  $\Theta$ . The invariant distribution of  $X$  equals that of  $\bar{Y}_1$  and the coalescence occurs at every step with positive probability. The former follows from Theorem 1.1 and the fact that the chains  $X$  and  $X'$  have the same transition probabilities (see Lemma 3.1 below) and the latter is a consequence of the structure of  $\psi$ .

Our aim is to sample  $X_0$ , whose law equals that of  $\bar{Y}_1$ . By construction of  $\psi$  it follows that  $\psi(x, \Theta) = \psi(a(\Theta), \Theta)$  for any  $x \in (0, a(\Theta)]$ , where  $\theta \mapsto a(\theta)$  is a positive deterministic function explicitly given in (3.3) of Lemma 3.1 below. The coalescence for  $X$  occurs every time the inequality  $X_n \leq a(\Theta_n)$  is satisfied, since, if  $-\sigma$  is such a time, then  $X_{-\sigma+1} = \psi(a(\Theta_{-\sigma}), \Theta_{-\sigma})$  disregards the value  $X_{-\sigma}$  and hence the entire trajectory of  $X$  prior to time  $-\sigma + 1$ .

The task now is to detect whether the event  $\{X_n \leq a(\Theta_n)\}$  occurred without knowing the value of  $X_n$  (if we had access to  $X_n$  for any  $n \in \mathbb{Z}$ , we would have a sample from the law of  $\bar{Y}_1$ !). DCFTP [21] suggests to look for a process  $D = \{D_n\}_{n \in \mathbb{Z}}$  satisfying  $D_n \geq X_n$  for all  $n \in \mathbb{Z}$ , which can be simulated backwards in time (starting at 0) together with the i.i.d. sequence  $\{\Theta_n\}_{n \in \mathbb{Z}}$ . It is possible to define such a process  $D$ , which turns out to be stationary but non-Markovian, by “unwinding” the recursion for  $X$  backwards in time and bounding the terms (see (3.8) in Sec. 3).

The backward simulation of  $\{(D_n, \Theta_n)\}_{n \in \mathbb{Z}}$  in step 1 of Algorithm 1 is discussed in Section 4 below. It relies on two ingredients: (A) the simulation of the indicators of independent events with summable probabilities and (B) the simulation of a random walk with negative drift and its

---

**Algorithm 1** Exact sampling from the law of  $\bar{Y}_1$

---

- 1: Starting at 0, sample  $\{(D_n, \Theta_n)\}_{n \in \mathbb{Z}}$  backwards in time until  $-\sigma = \sup\{n \leq 0 : D_n \leq a(\Theta_n)\}$
  - 2: Put  $X_{-\sigma+1} = \psi(a(\Theta_{-\sigma}), \Theta_{-\sigma})$
  - 3: Compute recursively  $X_n = \psi(X_{n-1}, \Theta_{n-1})$  for  $n = -\sigma + 2, \dots, 0$
  - 4: **return**  $X_0$
- 

future supremum. By the Borel-Cantelli lemma, only finitely many indicators in (A) are non-zero. A simple and efficient algorithm for the simulation of the entire sequence is given in Section 4.1 below. The algorithm for (B) has been developed in [4, Sec. 4]. For completeness, in Section 4.2 below we present the algorithm from [4, Sec. 4] applied to the specific random walk that arises in definition (3.8) of our dominating process  $D$ . The algorithm in [4, Sec. 4] requires the simulation of the walk under the original measure as well as under an exponential change of measure. In our case the increments of the random walk in question are shifted negative exponential random variables. This makes the dynamics of the walk explicit and easy to simulate under both measures (see Section 4.2 below for details), making the implementation of Algorithm 1 quite fast. More precisely, Algorithm 9 below (a version of Algorithm 1) was implemented in Julia, see the GitHub repository [19] for the code and a simple user guide. This implementation outputs approximately  $10^4$  samples every 1.15 seconds (see Section 5 for details).

Note that the random time  $\sigma$  in Algorithm 1 dictates the number of simulations, as steps 2-4 in the algorithm require only deterministic computation. In order to prove that  $\sigma$  is finite, we couple  $D$  with a dominating process  $D'$ , which is a component of a multi-dimensional positive Harris recurrent Markov chain  $\Xi$  (see (3.9) for the definition of  $D'$  and Lemma 3.2 of Section 3 below). Note that we need not be (and in fact are not) able to simulate  $D'$ . We apply the general state space Markov chain theory [24, 28] to prove the following result (see Section 3 below for details).

**Theorem 1.2.** *The random time  $\sigma$  in Algorithm 1 is finite a.s. Moreover,  $\mathbb{E}[\sigma | \Xi_0] < \infty$  a.s.*

In [18, Thm 5.1] the authors provide a sharp estimate on  $\mathbb{E}[\sigma]$  for an analogous algorithm in the context of Vervaat perpetuities. Their analysis is based on the fact that their dominating process  $D$  is a birth-death Markov chain and is hence time-reversible with skip-free increments and an explicit invariant distribution (shifted geometric). In the context of Theorem 1.2, the dominating process  $D$  is non-Markovian, its increments are diffuse, have heavy tails and the multi-dimensional Markov chain  $\Xi$  used to bound  $D$  has a non-explicit invariant probability measure  $\pi$  (which also has heavy

tails). These heavy tails make the chain frequently take large values, which in turn makes the coalescence events and probabilities harder to trace, bound and control. Moreover, the law of the time-reversal of  $\Xi$  (with respect to  $\pi$ ) is very different from that of  $\Xi$ . The key step in the proof of Theorem 1.2 is provided by [28, Thm 8.1.1], which allows us to conclude that the time-reversed chain has a Harris recurrent modification. However, a quantitative bound on the expected number of steps taken backwards in time in Algorithm 1 remains an open problem.

## 1.2. Related literature

Exact simulation algorithms for various instances of a general perpetuity equation  $\mathcal{X} \stackrel{d}{=} A_0\mathcal{X} + A_1$  (with  $(A_0, A_1)$  and  $\mathcal{X}$  independent) have been developed in the literature.

Paper [18] studies the case  $A_0 = A_1 \geq 0$ ,  $\mathbb{E}[A_0] < 1$ , specialising to the Vervaat perpetuity for  $A_0 = U^{1/\beta}$  with  $U$  uniform on  $(0, 1)$  and  $\beta \in (0, \infty)$ , see also [9, 12]. Briefly put, [18] first identifies the update function and constructs a multigamma coupler. The identified dominating process is a simple random walk with a partially absorbing barrier and whose invariant law is that of a shifted geometric random variable. A sped up version of a DCFTP algorithm [12] in the case  $\beta = 1$  (i.e. when  $\mathcal{X}$  follows the Dickman distribution) is given in [13].

In [15], the authors develop the double CFTP algorithm in the case  $A_0 = V$  and  $A_1 = (1 - V)Z$ , where  $V$  takes values in  $[0, 1]$  (and has a computable density) and  $Z$  is independent of  $V$  with support in an interval  $[0, c]$  for some  $c < \infty$ . This structure appears similar to perpetuity (2.1) of Proposition 2.1 below, where  $A_0 = U^{1/\alpha}$  and  $A_1 = (1 - U)^{1/\alpha} \max\{Y_1, 0\}$  with  $Y_1$  an  $\alpha$ -stable random variable independent of the uniform  $U$ . Proposition 2.1 provides a key step in the proof of Theorem 1.1 above, which in turn is the cornerstone of Algorithm 1. The upper bound  $c$  on the support of  $Z$  in [15] is inversely proportional to the coalescence probability of the chain in the double CFTP algorithm, making its direct application to perpetuity (2.1) impossible, since  $\max\{Y_1, 0\}$  not only has infinite support but also a heavy tail. Moreover, even if we could construct a stochastic (rather than constant) upper bound on the relevant support, this bound would necessarily still have a heavy tail making the coalescence in a generalisation of the algorithm in [15] unlikely. This would then yield long (possibly infinite) running times for such a generalisation.

Paper [10] studies the generalised Vervaat perpetuity where  $A_1 = A_0A_2$  for independent  $A_2$  and  $A_0 = U^{1/\beta}$  with  $U$  uniform on  $(0, 1)$ . By calculating the Laplace transform from the perpetuity, it is shown in [10] that  $\mathcal{X}$  has the law of the marginal of a pure jump Lévy process at time  $\beta$  with

Lévy density  $\nu(dx) = |x|^{-1}(\mathbb{P}(A_2 > x)1_{x>0} + \mathbb{P}(A_2 < -x)1_{x<0})dx$ . Techniques similar to those in [6], based on infinite divisibility, are used to devise the simulation algorithm under the conditions  $A_2 \geq 0$  and  $\lim_{x \downarrow 0} \mathbb{P}(A_2 \leq x)/x < \infty$ , without relying on Markov chain techniques. The calculation of Laplace transforms based on perpetuities (2.1) or (1.1), yields complicated equations for the Laplace transform. Furthermore, even if we could solve for the Laplace transform of  $\bar{Y}_1$ , we could not follow the simulation approach from [10] as  $\bar{Y}_1$  is typically not infinitely divisible.

In [4] the authors use a version of a multigamma coupler, allowing  $A_1$  to have a heavy tail but assuming the independence of  $A_0$  and  $A_1$ , a requirement clearly violated by perpetuities (1.1) and (2.1) in the present paper. Moreover, a certain domination condition [4, Eq. (2) in Assumption (B)] for the density of  $A_1$  is stipulated, which plays an important role in constructing the coalescence probability. This dominating condition is hard to establish for the density of a stable law conditioned on being positive, appearing in perpetuity (1.1). Thus, even if one could remove the assumption on the independence of  $A_0$  and  $A_1$  in [4], this technical requirement would make it hard to apply directly the sampling algorithm from [4] in our setting.

The structure of the multigamma coupler used in the present paper is closer to the one in [18] (see also Section 1.1 above and Lemma 3.1 below) than the one in [4]. Despite the differences between the samplers in [4] and the one used here, the construction of our dominating process was inspired by the one presented in [4]. However, we were unable to use directly the dominating process  $V_k^+$  in [4, Eq. (9)], which appears to be bounded from below by the deterministic function  $k \mapsto e^{ak/2}/(1 - e^{-a/2})$  (for all positive integers  $k$  and some constant  $a > 0$ ) tending to infinity exponentially fast and hence suggesting a positive probability of never detecting coalescence. It appears that this issue could be circumvented in the general context of [4] by a simple adaptation of our dominating process defined in (3.8) below, which is based on the idea of adaptive bounds (cf. Figure 4.1).

A perpetuity can be understood as the special case of the stochastic fixed point equation  $\mathcal{X} \stackrel{d}{=} f(\mathcal{X}, U)$  in a general state space for independent  $\mathcal{X}$  and  $U$  and some measurable function  $f$ . See the monograph [20] for a comprehensive survey on the variety of Markov chain techniques, such as CFTP and DCFTP, used to obtain exact samples of  $\mathcal{X}$ .

The problem of the exact simulation of the first passage event of a spectrally positive stable process (resp. a Lévy process with infinite activity and finite variation) is addressed in [8] (resp. [7]). Algorithm 1 solves this problem for all stable processes as follows: for any  $x > 0$ , define the *first*

passage time  $\tau_x := \inf\{t > 0 : Y_t \geq x\}$  and note that the equality of events  $\{\tau_x > t\} = \{\bar{Y}_t < x\}$  for all  $t \in (0, \infty)$  and the scaling property yield the equality in law  $\tau_x \stackrel{d}{=} (x/\bar{Y}_1)^\alpha$ .

We conclude the introduction by noting that Proposition 2.1 easily implies the asymptotic behaviour at infinity of the distribution function of  $\bar{Y}_1$  stated in [3, Prop. VIII.1.4, p. 221]. Excluding the spectrally negative case, perpetuity (2.1) and the Grincevičius-Grey theorem [5, Thm 2.4.3] yield  $\lim_{x \rightarrow \infty} 2\mathbb{P}(Y_1 U^{1/\alpha} > x) / \mathbb{P}(\bar{Y}_1 > x) = 1$ . By Breiman's lemma [5, Lem. B.5.1] we have  $\lim_{x \rightarrow \infty} 2\mathbb{P}(Y_1 U^{1/\alpha} > x) / \mathbb{P}(Y_1 > x) = 1$ , implying  $\lim_{x \rightarrow \infty} \mathbb{P}(\bar{Y}_1 > x) / x^{-\alpha} = \Gamma(\alpha) \sin(\pi\alpha\rho) / \pi$  via the classical tail behaviour of the stable law [30, Sec. 4.3].

The remainder of the paper is structured as follows. In Section 2, we establish perpetuity (2.1) and apply it in the proof of Theorem 1.1. In Section 3 we define the update function  $\psi$  (in Lemma 3.1), construct the dominating process and prove Theorem 1.2 above. Section 4 discusses the backward simulation of  $\{(D_n, \Theta_n)\}_{n \in \mathbb{Z}}$ . Finally, a numerical performance analysis is found in Section 5.

## 2. Stochastic Perpetuities

Let  $Y$  be a stable process with stability and positivity parameters  $\alpha$  and  $\rho$ , respectively (see Appendix A below for definition). Since  $Y_0 = 0$  and the scaling property yield  $\bar{Y}_t = \sup_{s \in [0, t]} Y_s \stackrel{d}{=} t^{1/\alpha} \bar{Y}_1$  for all  $t \in [0, \infty)$ , we may restrict our attention to  $\bar{Y}_1$ . Let  $S(\alpha, \rho)$  and  $\bar{S}(\alpha, \rho)$  denote the laws of  $Y_1$  and  $\bar{Y}_1$ , respectively. Since  $\mathbb{P}(Y_t > 0) = \rho$  for any  $t > 0$ , the extreme cases  $\rho \in \{0, 1\}$  are excluded from our analysis as they correspond to  $Y$  having monotone paths. Let  $U(0, 1)$  denote the uniform law on  $(0, 1)$  and define  $x^+ = \max\{x, 0\}$  for any real number  $x \in \mathbb{R}$ .

**Proposition 2.1.** *Let  $(\bar{Y}_1, Z, U) \sim \bar{S}(\alpha, \rho) \times S(\alpha, \rho) \times U(0, 1)$ . Then the law of  $\bar{Y}_1$  is the unique solution of the following perpetuity:*

$$\bar{Y}_1 \stackrel{d}{=} U^{\frac{1}{\alpha}} \bar{Y}_1 + (1 - U)^{\frac{1}{\alpha}} Z^+. \quad (2.1)$$

To prove this result, we need the next definition. For any  $a < b$ , the *concave majorant* of a function  $f : [a, b] \rightarrow \mathbb{R}$  is defined as the smallest concave function  $c : [a, b] \rightarrow \mathbb{R}$ , such that  $c(t) \geq f(t)$  for every  $t \in [a, b]$ . The proof of Proposition 2.1 exploits the fact that the supremum of a function lies on its concave majorant, at the end of all (if any) faces with positive slope. Following the classical result for the complete description of a concave majorant of random walks, [27] describes



the continuous time analogue of these results for Lévy processes ([27] is phrased in terms of the convex minorant, but through a change of sign their results cover the concave majorant). The idea is as follows: fix a sample path of  $Y$  and pick a random face of its concave majorant above an independent uniform point in  $[0, 1]$ . The length of the chosen face is distributed as  $V \sim U(0, 1)$  and its height is distributed as the increment of a stable process over a time interval of duration  $V$ . Moreover, after removing this face (together with the path underneath it) the remainder of the concave majorant behaves like a concave majorant of a stable process over the time interval  $[0, 1 - V]$ , see [27]. This recursive relation and the scaling property of  $Y$  will yield the perpetuity in (2.1).

*Proof.* A stick-breaking process  $\{\ell_n\}_{n \geq 1}$  on  $[0, 1]$  is defined recursively as follows:

$$\ell_n = V_n (1 - L_{n-1}), \quad n \geq 1,$$

where  $L_{n-1} = \ell_1 + \dots + \ell_{n-1}$ ,  $L_0 = 0$  and  $\{V_n\}_{n \geq 1}$  is a sequence of i.i.d. random variables with law  $U(0, 1)$  (independent of  $Y$ ). Let  $C = (C_t)_{t \in [0, 1]}$  be the concave majorant of the Lévy process  $Y$ . Let  $(d_n - g_n, C_{d_n} - C_{g_n})_{n \geq 1}$  be the lengths and heights of the faces of  $C$  picked at random, uniformly on lengths and without replacement ( $g_n$  and  $d_n$  denote the beginning and end times for the  $n$ -th face). [27, Thm. 1] asserts the equality in law

$$(d_n - g_n, C_{d_n} - C_{g_n})_{n \geq 1} \stackrel{d}{=} (\ell_n, Y_{L_n} - Y_{L_{n-1}})_{n \geq 1}.$$

The concave majorant  $(C_t)_{t \in [0, 1]}$  is piecewise linear, with the corresponding slopes forming a non-increasing piecewise constant function in  $t$ . Hence  $\bar{Y}_1$  is always contained in the image of the function  $C$ . Moreover, the supremum equals the sum of all the positive heights of  $C$ :

$$\bar{Y}_1 = \sum_{n=1}^{\infty} (C_{d_n} - C_{g_n})^+ \stackrel{d}{=} \sum_{n=1}^{\infty} (Y_{L_n} - Y_{L_{n-1}})^+.$$

Conditional on  $\{L_n\}_{n \geq 1}$ , the random variables  $\{Y_{L_n} - Y_{L_{n-1}}\}_{n \geq 1}$  are independent and have the same distribution as the respective  $Y_{\ell_n}$ . Hence, for an independent i.i.d. sequence  $\{Z_n\}_{n \geq 1}$  with law  $S(\alpha, \rho)$  we have

$$(\ell_n, Y_{Z_n} - Y_{Z_{n-1}})_{n \geq 1} \stackrel{d}{=} \left( \ell_n, \ell_n^{\frac{1}{\alpha}} Z_n \right)_{n \geq 1},$$

implying

$$\bar{Y}_1 \stackrel{d}{=} \sum_{n=1}^{\infty} (Y_{L_n} - Y_{L_{n-1}})^+ \stackrel{d}{=} \sum_{n=1}^{\infty} \ell_n^{\frac{1}{\alpha}} Z_n^+. \quad (2.2)$$

It is well-known that  $\left\{\frac{\ell_n}{1-\ell_1}\right\}_{n \geq 2}$  is a stick-breaking process on  $[0, 1]$ , independent of  $\ell_1 \sim U(0, 1)$  (and  $\{Z_n\}_{n \geq 1}$ ). Hence by (2.2) we find the equality in law

$$\bar{Y}_1 \stackrel{d}{=} \sum_{n=2}^{\infty} \left(\frac{\ell_n}{1-\ell_1}\right)^{\frac{1}{\alpha}} Z_n^+,$$

which, together with (2.2), implies the perpetuity

$$\bar{Y}_1 \stackrel{d}{=} \ell_1^{\frac{1}{\alpha}} Z_1^+ + (1-\ell_1)^{\frac{1}{\alpha}} \bar{Y}_1.$$

Finally, the uniqueness of solution follows from [5, Thm 2.1.3].  $\square$

Let  $S^+(\alpha, \rho)$  denote the law of  $Y_1$  conditioned on being positive. For  $n, m \in \mathbb{Z}$  define the sets

$$\mathcal{Z}^n = \{k \in \mathbb{Z} : k < n\}, \quad \mathcal{Z}_m^n = \mathcal{Z}^n \setminus \mathcal{Z}^m. \quad (2.3)$$

*Proof of Theorem 1.1.* Note that the random variable  $Z^+$  in Proposition 2.1 behaves like the product of a Bernoulli random variable and a stable random variable conditioned on being positive, i.e., if  $B \sim \text{Ber}(\rho)$  and  $S \sim S^+(\alpha, \rho)$  are independent, then  $Z^+ \stackrel{d}{=} BS$ . Since  $\mathbb{P}(Z^+ = 0) = 1 - \rho > 0$ , the idea behind the proof of Theorem 1.1 is to iterate perpetuity (2.1) backwards in time until the first time we observe  $Z^+ > 0$ .

More precisely, by Proposition 2.1 and Kolmogorov's consistency theorem we can construct a stationary Markov chain  $\{(U_n, Z_n, \zeta_n)\}_{n \in \mathbb{Z}^1}$  with invariant law  $U(0, 1) \times S(\alpha, \rho) \times \bar{S}(\alpha, \rho)$ , where  $\{(U_n, Z_n)\}_{n \in \mathbb{Z}^1}$  is an i.i.d. sequence with law  $U(0, 1) \times S(\alpha, \rho)$  and

$$\zeta_{n+1} = U_n^{\frac{1}{\alpha}} Z_n^+ + (1 - U_n)^{\frac{1}{\alpha}} \zeta_n, \quad n \in \mathbb{Z}^0.$$

Define  $V_0 = 1$  and  $V_n = \prod_{m \in \mathbb{Z}_n^0} (1 - U_m)$  for  $n \in \mathbb{Z}^0$ . Then the following equality holds

$$\zeta_0 = \sum_{m \in \mathbb{Z}_n^0} (U_m V_{m+1})^{\frac{1}{\alpha}} Z_m^+ + V_n^{\frac{1}{\alpha}} \zeta_n \quad \text{for all } n \in \mathbb{Z}^0. \quad (2.4)$$

Let  $\tau = \sup\{n \in \mathbb{Z}^0 : Z_n > 0\}$  (with convention  $\sup \emptyset = -\infty$ ) be the last time we see a positive value in the sequence  $\{Z_n\}_{n \in \mathbb{Z}^0}$ . Substituting  $n = \tau$  in equation (2.4), we get

$$\zeta_0 = V_{\tau+1}^{\frac{1}{\alpha}} \left( (1 - U_\tau)^{\frac{1}{\alpha}} \zeta_\tau + U_\tau^{\frac{1}{\alpha}} Z_\tau \right). \quad (2.5)$$

This equality of course yields the same equality in law. It will hence imply the perpetuity in (1.1), if we prove that the random variables involved have the desired laws and independence structure.

The events  $\{Z_n > 0\}$ ,  $n \in \mathcal{Z}^0$ , are independent with probability  $\rho$ , making  $\tau$  a geometric random variable on  $\mathcal{Z}^0$  with parameter  $\rho$ . By construction, the coordinates of the vector  $(U_n, Z_n, \zeta_n)$  are independent for any  $n \in \mathcal{Z}^0$ . Hence we have  $(U_\tau, Z_\tau, \zeta_\tau) \sim U(0, 1) \times S^+(\alpha, \rho) \times \bar{S}(\alpha, \rho)$ . Moreover,  $(U_\tau, Z_\tau, \zeta_\tau)$  is independent of  $(\tau, V_{\tau+1})$ . Hence (2.5) will imply the perpetuity in the theorem if we prove that  $\Lambda$  has the same law as  $V_{\tau+1}$ . Put differently, as  $\tau$  and  $U_0$  are independent, it is sufficient to prove the following equality in law

$$V_{\tau+1} \stackrel{d}{=} 1_{\tau=-1} + 1_{\tau \neq -1} U_0^{\frac{1}{\rho}} = 1 + 1_{\tau \neq -1} \left( U_0^{\frac{1}{\rho}} - 1 \right). \quad (2.6)$$

Since  $-\log(1 - U_1) \sim \text{Exp}(1)$  is exponential with mean one,  $-\log(V_n)$  is gamma distributed with density  $x \mapsto x^{-n-1}e^{-x}/(-n-1)!$  for any  $n \in \mathcal{Z}^0$ . Hence, on the event  $\{\tau \neq -1\}$ , the density of the conditional law  $-\log(V_{\tau+1})|\tau$  is given by  $x \mapsto x^{-\tau-2}e^{-x}/(-\tau-2)!$ . Thus, the conditional law  $-\log(V_{\tau+1})|\{\tau \neq -1\}$  is exponential with density

$$x \mapsto \frac{1}{1-\rho} \sum_{k=2}^{\infty} \rho(1-\rho)^{k-1} \frac{x^{k-2}}{(k-2)!} e^{-x} = \rho e^{-\rho x}, \quad x > 0. \quad (2.7)$$

Since  $-\log(V_{\tau+1})$  takes the value 0 when  $\tau = -1$ , which happens with probability  $\rho$ , and is otherwise exponential with mean  $1/\rho$ , the distributional identity in (2.6) follows.

Finally, the uniqueness of the solution for perpetuity (1.1) follows from [5, Thm 2.1.3].  $\square$

### 3. The Markov chain $X$ and the dominating process $D$ in Algorithm 1

Let  $\mathcal{A} = (0, \infty) \times (0, 1) \times (0, 1) \times (0, 1]$  and define the function  $\phi : (0, \infty) \times \mathcal{A} \rightarrow (0, \infty)$  by

$$\phi(x, \theta) = \lambda^{\frac{1}{\alpha}} \left( u^{\frac{1}{\alpha}} x + (1-u)^{\frac{1}{\alpha}} s \right), \quad x \in (0, \infty), \quad \theta = (s, u, w, \lambda) \in \mathcal{A}. \quad (3.1)$$

Note that the map  $x \mapsto \phi(x, \theta)$  is increasing and linear in  $x$  for all  $\theta \in \mathcal{A}$  and does not depend on  $w$ . Let  $W \sim U(0, 1)$  be independent of random variables  $S, U$ , and  $\Lambda$  defined in Theorem 1.1. Then, by Theorem 1.1, we have  $\zeta \stackrel{d}{=} \phi(\zeta, \Theta)$ , where  $\zeta \sim \bar{S}(\alpha, \rho)$  is independent of  $\Theta = (S, U, W, \Lambda)$ . Hence a Markov chain with the update function  $\phi$  has the correct invariant law but does not allow for coalescence: if for any  $x, y \in (0, \infty)$  we have  $\phi(x, \Theta) = \phi(y, \Theta)$ , by (3.1) it follows  $x = y$ . But the structure of  $\phi$  and the additional randomness in  $W$  allow us to modify the update function  $x \mapsto \phi(x, \theta)$  so that coalescence can be achieved, while keeping the law of the chain unchanged.

**Lemma 3.1.** Define the functions  $\psi : (0, \infty) \times \mathcal{A} \rightarrow (0, \infty)$  and  $a : \mathcal{A} \rightarrow (0, \infty)$  by the formulae

$$\psi(x, \theta) = 1_{\{a(\theta) \geq x\}} w^{\frac{1}{\alpha\rho}} (1-u)^{\frac{1}{\alpha}} s + 1_{\{a(\theta) < x\}} \lambda^{\frac{1}{\alpha}} \left( u^{\frac{1}{\alpha}} x + (1-u)^{\frac{1}{\alpha}} s \right), \quad (3.2)$$

$$a(\theta) = \left( \lambda^{-\frac{1}{\alpha}} - 1 \right) \left( \frac{1-u}{u} \right)^{\frac{1}{\alpha}} s. \quad (3.3)$$

The map  $x \mapsto \psi(x, \theta)$  is non-decreasing in  $x$  for all  $\theta \in \mathcal{A}$ . Moreover, for  $\zeta$  and  $\Theta$  as in the paragraph above, we have  $\phi(x, \Theta) \stackrel{d}{=} \psi(x, \Theta)$  for all  $x > 0$  and  $\bar{S}(\alpha, \rho)$  is the unique solution of the distributional equation  $\zeta \stackrel{d}{=} \psi(\zeta, \Theta)$ .

*Proof.* The function  $\psi$  takes constant value of  $w^{\frac{1}{\alpha\rho}} (1-u)^{\frac{1}{\alpha}} s$  for  $x \in (0, a(\theta)]$  and increases linearly on the interval  $(a(\theta), \infty)$  with the right limit satisfying  $\lim_{x \searrow a(\theta)} \psi(x, \theta) = (1-u)^{\frac{1}{\alpha}} s > \psi(a(\theta), \theta)$ . Hence the desired monotonicity follows.

We now prove that  $\phi(x, \Theta) \stackrel{d}{=} \psi(x, \Theta)$  for all  $x > 0$ , i.e the transition probabilities for the update functions  $\phi$  and  $\psi$  coincide. Pick  $x > 0$  and note that  $\{\phi(x, \Theta) = \psi(x, \Theta)\} \supset \{a(\Theta) < x\}$ . Thus, for any  $y > 0$  we have  $\mathbb{P}(\phi(x, \Theta) \leq y, a(\Theta) < x) = \mathbb{P}(\psi(x, \Theta) \leq y, a(\Theta) < x)$ . Define

$$v(u, s) = \left( \frac{(1-u)^{\frac{1}{\alpha}} s}{u^{\frac{1}{\alpha}} x + (1-u)^{\frac{1}{\alpha}} s} \right)^{\alpha\rho} \in (0, 1),$$

and note that  $\{a(\Theta) \geq x\} = \{\Lambda^\rho \leq v(U, S)\}$ . On this event, the definition of  $\Lambda$  in Theorem 1.1 implies the inequality  $\Lambda < 1$ , in which case  $\Lambda^\rho$  is uniform on  $(0, 1)$ . Hence the conditional law of  $\Lambda$ , given  $(U, S)$  and  $\{a(\Theta) \geq x\}$ , is uniform on the interval  $(0, v(U, S))$ . Moreover, the conditional law of  $v(U, S)W$ , given  $(U, S)$  and on  $\{a(\Theta) \geq x\}$ , is also uniform on  $(0, v(U, S))$ . Hence for any  $y > 0$  the following equalities hold:

$$\begin{aligned} \mathbb{P}(\phi(x, \Theta) \leq y, a(\Theta) \geq x | U, S) &= \mathbb{P}\left(\Lambda^\rho \leq \left(\frac{y}{U^{\frac{1}{\alpha}} x + (1-U)^{\frac{1}{\alpha}} S}\right)^{\alpha\rho}, a(\Theta) \geq x \mid U, S\right) \\ &= \mathbb{P}\left(v(U, S)W \leq \left(\frac{y}{U^{\frac{1}{\alpha}} x + (1-U)^{\frac{1}{\alpha}} S}\right)^{\alpha\rho}, a(\Theta) \geq x \mid U, S\right) \\ &= \mathbb{P}\left(W^{\frac{1}{\alpha\rho}} (1-U)^{\frac{1}{\alpha}} S \leq y, a(\Theta) \geq x \mid U, S\right) \\ &= \mathbb{P}(\psi(x, \Theta) \leq y, a(\Theta) \geq x | U, S). \end{aligned}$$

Taking expectations in this identity yields the unconditional equality  $\mathbb{P}(\phi(x, \Theta) \leq y, a(\Theta) \geq x) = \mathbb{P}(\psi(x, \Theta) \leq y, a(\Theta) \geq x)$ . Hence we get  $\mathbb{P}(\phi(x, \Theta) \leq y) = \mathbb{P}(\psi(x, \Theta) \leq y)$  for all  $y > 0$ , implying the equality in law  $\phi(x, \Theta) \stackrel{d}{=} \psi(x, \Theta)$  for arbitrary  $x > 0$ .

Pick  $y > 0$ . Since  $\Theta$  and  $\zeta$  are independent, by Theorem 1.1 we have

$$\begin{aligned} \mathbb{P}(\zeta \leq y) = \mathbb{P}(\phi(\zeta, \Theta) \leq y) &= \int_{[0, \infty)} \mathbb{P}(\phi(x, \Theta) \leq y) \mathbb{P}(\zeta \in dx) \\ &= \int_{[0, \infty)} \mathbb{P}(\psi(x, \Theta) \leq y) \mathbb{P}(\zeta \in dx) = \mathbb{P}(\psi(\zeta, \Theta) \leq y), \end{aligned}$$

implying  $\zeta \stackrel{d}{=} \psi(\zeta, \Theta)$ . Moreover, if there exists some  $\zeta'$  (independent of  $\Theta$ ) satisfying  $\zeta' \stackrel{d}{=} \psi(\zeta', \Theta)$ , this calculation implies the equality  $\zeta' \stackrel{d}{=} \phi(\zeta', \Theta)$ . By Theorem 1.1 we get  $\zeta' \stackrel{d}{=} \zeta$ , as claimed.  $\square$

By Lemma 3.1 and Kolmogorov's consistency theorem, there exists a probability space supporting a sequence  $\{\Theta_n\}_{n \in \mathbb{Z}}$  of independent copies of  $\Theta$  and a stationary Markov chain  $\{X_n\}_{n \in \mathbb{Z}}$ , satisfying  $X_{n+1} = \psi(X_n, \Theta_n)$  for all  $n \in \mathbb{Z}$ . In the remainder of the paper,  $\{(X_n, \Theta_n)\}_{n \in \mathbb{Z}}$  denotes the corresponding Markov chain on  $(0, \infty) \times \mathcal{A}$ . In order to detect coalescence in Algorithm 1, we now construct a dominating process  $\{D_n\}_{n \in \mathbb{Z}}$ .

With this in mind, fix constants  $\delta$  and  $d$  satisfying  $0 < \delta < d < \frac{1}{\alpha\rho}$ . Let  $I_k^n = 1_{\{S_k > e^{\delta(n-1-k)}\}}$  for all  $n \in \mathbb{Z}$ ,  $k \in \mathbb{Z}^n$  (see (2.3) above), where  $S_k \sim S^+(\alpha, \rho)$  is the first component of  $\Theta_k$  (see the first paragraph of Section 3). Fix  $\gamma > 0$  such that  $\mathbb{E}S_1^\gamma < \infty$  (see (A.2)). Markov's inequality implies

$$p(m) = \mathbb{P}(S_1 \leq e^{\delta m}) \geq 1 - e^{-\delta\gamma m} \mathbb{E}S_1^\gamma, \quad m \geq 0, \quad (3.4)$$

and hence  $\sum_{m=0}^{\infty} (1-p(m)) < \infty$ . Since  $\{S_k\}_{k \in \mathbb{Z}}$  are independent, the Borel-Cantelli lemma ensures that, for a fixed  $n \in \mathbb{Z}$ , the events  $\{S_k > e^{\delta(n-1-k)}\} = \{I_k^n = 1\}$  occur for only finitely many  $k \in \mathbb{Z}^n$  a.s. Let  $\chi_n$  be the smallest time beyond which the indicators  $I_k^n$  are all zero:

$$\chi_n = (n-1) \wedge \inf \{k \in \mathbb{Z}^n : I_k^n = 1\}, \quad (3.5)$$

with convention  $\inf \emptyset = \infty$ . Note that  $-\infty < \chi_n \leq n-1$  holds a.s. for all  $n \in \mathbb{Z}$ . Since the integers are countable, we have  $n-1 \geq \chi_n > -\infty$  for all  $n \in \mathbb{Z}$  a.s.

Define the i.i.d. sequence  $\{F_n\}_{n \in \mathbb{Z}}$  by  $F_n = d + \frac{1}{\alpha} \log(\Lambda_n U_n)$ , where  $U_n$  and  $\Lambda_n$  are the second and fourth components of  $\Theta_n$ , respectively (see the first paragraph of Section 3). Note that  $d - F_n$  has the same law as a sum of (random) geometrically many independent exponential random variables and is hence exponentially distributed with mean  $\mathbb{E}[d - F_n] = \frac{1}{\alpha\rho}$ . Let  $C = \{C_n\}_{n \in \mathbb{Z}}$  be a random walk defined by  $C_0 = 0$  and

$$C_{n+1} = C_n - F_n, \quad n \in \mathbb{Z}. \quad (3.6)$$

Recall definition (2.3) and let  $R = \{R_n\}_{n \in \mathbb{Z}}$  be the reflected process of the walk  $\{C_n\}_{n \in \mathbb{Z}}$ , that is

$$R_n = \sup_{k \in \mathcal{Z}^{n+1}} C_k - C_n, \quad n \in \mathbb{Z}. \quad (3.7)$$

For any  $n \in \mathbb{Z}$ , define the following random variables

$$D_n = \exp(R_n) \left( \frac{e^{(d-\delta)(\chi_n-n)}}{1 - e^{\delta-d}} + \sum_{k \in \mathcal{Z}_{\chi_n}^n} e^{-(n-1-k)d} S_k (1 - U_k)^{\frac{1}{\alpha}} \right), \quad (3.8)$$

$$D'_n = \exp(R_n) \left( \frac{1}{1 - e^{\delta-d}} + D''_n \right), \quad \text{where} \quad D''_n = \sum_{k \in \mathcal{Z}^n} e^{-(n-1-k)d} S_k. \quad (3.9)$$

The sum in (3.8) is taken to be zero if  $\mathcal{Z}_{\chi_n}^n = \emptyset$ , i.e. if  $\chi_n = n$ . Note that the series in  $D''_n$  is absolutely convergent by the Borel-Cantelli lemma, but  $D'_n$  cannot be simulated directly as it depends on an infinite sum. Finally, define the random element  $\Xi_n = (\Theta_n, R_n, D'_n)$  for any  $n \in \mathbb{Z}$ .

**Lemma 3.2.** (a)  $X_n \leq D_n \leq D'_n$  for all  $n \in \mathbb{Z}$  a.s.

(b) The processes  $R = \{R_n\}_{n \in \mathbb{Z}}$  and  $\Xi = \{\Xi_n\}_{n \in \mathbb{Z}}$  are Markov, stationary and  $\varphi$ -irreducible (see definition [24, p. 82]) with respect to the respective invariant distributions.

*Proof.* (a) Since  $\mathbb{E}F_1 < 0$ , by the strong law of large numbers we have  $C_{-n} \rightarrow -\infty$  a.s. as  $n \rightarrow \infty$ . Hence  $R_n < \infty$  for all  $n \in \mathbb{Z}$  a.s. and a direct termwise comparison yields  $D'_n \geq D_n$  for all  $n \in \mathbb{Z}$ . It remains to prove that  $X_n \leq D_n$  for all  $n \in \mathbb{Z}$ .

Recall that the function  $\theta \mapsto a(\theta)$  is defined in (3.3). Let  $\tau_n = \sup \{k \in \mathcal{Z}^n : X_k \leq a(\Theta_k)\}$  (with convention  $\sup \emptyset = -\infty$ ) be the last time the coalescence occurred before  $n \in \mathbb{Z}$ . If  $\tau_n > -\infty$ , the value  $X_{1+\tau_n}$  does not depend on  $X_{\tau_n}$ , and neither do the values of the chain taken at subsequent times. In particular,

$$X_n = \psi(X_{n-1}, \Theta_{n-1}) = \underbrace{\psi \left( \dots \psi \left( W_{\tau_n}^{\frac{1}{\alpha\rho}} (1 - U_{\tau_n})^{\frac{1}{\alpha}} S_{\tau_n}, \Theta_{\tau_n+1} \right), \dots, \Theta_{n-1} \right)}_{n-1-\tau_n}.$$

In general, by (3.2) and (2.3),  $X_n$  can be expressed as

$$\begin{aligned} X_n &= \sum_{k \in \mathcal{Z}_{\tau_n+1}^n} \exp \left( \frac{1}{\alpha} \sum_{j \in \mathcal{Z}_{k+1}^n} \log(\Lambda_j U_j) \right) \Lambda_k^{\frac{1}{\alpha}} (1 - U_k)^{\frac{1}{\alpha}} S_k \\ &+ \mathbf{1}_{\{\tau_n > -\infty\}} \exp \left( \frac{1}{\alpha} \sum_{j \in \mathcal{Z}_{\tau_n+1}^n} \log(\Lambda_j U_j) \right) W_{\tau_n}^{\frac{1}{\alpha\rho}} (1 - U_{\tau_n})^{\frac{1}{\alpha}} S_{\tau_n}, \end{aligned} \quad (3.10)$$

where sums over empty sets in (3.10) are defined to be equal to zero and, if  $\tau_n = -\infty$ , we define  $\mathcal{Z}_{\tau_n+1}^n = \mathcal{Z}^n$ . A termwise comparison then yields

$$\begin{aligned} X_n &\leq \sum_{k \in \mathcal{Z}^n} e^{C_{k+1} - C_n - (n-1-k)d} (1 - U_k)^{\frac{1}{\alpha}} S_k \\ &\leq e^{R_n} \sum_{k \in \mathcal{Z}^n} e^{-(n-1-k)d} (1 - U_k)^{\frac{1}{\alpha}} S_k \quad \text{for all } n \in \mathbb{Z} \text{ a.s.} \end{aligned} \quad (3.11)$$

Recall that  $S_k (1 - I_k^n) \leq e^{\delta(n-1-k)} (1 - I_k^n)$  for all  $k \in \mathcal{Z}^n$ . Since  $I_k^n = 0$  for  $k < \chi_n$ , we get

$$\begin{aligned} \sum_{k \in \mathcal{Z}^n} e^{-(n-1-k)d} (1 - U_k)^{\frac{1}{\alpha}} S_k &\leq \sum_{k \in \mathcal{Z}^{\chi_n}} e^{-(n-1-k)(d-\delta)} (1 - U_k)^{\frac{1}{\alpha}} + \sum_{k \in \mathcal{Z}_{\chi_n}^n} e^{-(n-1-k)d} (1 - U_k)^{\frac{1}{\alpha}} S_k \\ &\leq \frac{e^{(\chi_n - n)(d-\delta)}}{1 - e^{\delta-d}} + \sum_{k \in \mathcal{Z}_{\chi_n}^n} e^{-(n-1-k)d} (1 - U_k)^{\frac{1}{\alpha}} S_k. \end{aligned} \quad (3.12)$$

The inequalities in (3.11)–(3.12) and the definition in (3.8) imply  $X_n \leq D_n$  for all  $n \in \mathbb{Z}$  a.s.

(b) Note that  $C_k - C_n = \sum_{i=k}^{n-1} F_i$  for all  $k \in \mathcal{Z}^n$ . Hence  $R_n = \sup\{C_k - C_n : k \in \mathcal{Z}^{n+1}\}$  and  $F_n$  are independent and the Markov property for  $\{R_n\}_{n \in \mathbb{Z}}$  follows from

$$R_n = \max \left\{ \sup_{k \in \mathcal{Z}^n} C_k - C_n, 0 \right\} = \max \{R_{n-1} + F_{n-1}, 0\}.$$

By (3.9) we have  $D_n'' = S_{n-1} + e^{-d} D_{n-1}''$ . Hence  $(R_n, D_n')$  is a function of  $\Xi_{n-1} = (\Theta_{n-1}, R_{n-1}, D_{n-1}')$  (recall that  $S_{n-1}$  is the first component of the random vector  $\Theta_{n-1}$ ). Since the random elements  $\Xi_{n-1}$  and  $\Theta_n$  are independent, the process  $\{\Xi_n\}_{n \in \mathbb{Z}}$  is Markov.

The vector  $\Xi_n = (\Theta_n, R_n, D_n')$  is in a bijective correspondence with  $(\Theta_n, R_n, D_n'')$ .

Since  $\{\Theta_n\}_{n \in \mathbb{Z}}$  are i.i.d., the following equality in law holds

$$(R_{n+1}, D_{n+1}'') = \left( \sup_{j \in \mathcal{Z}^1} \sum_{k \in \mathcal{Z}_j^1} F_{n+k}, \sum_{k \in \mathcal{Z}^1} e^{kd} S_{n+k} \right) \stackrel{d}{=} \left( \sup_{j \in \mathcal{Z}^1} \sum_{k \in \mathcal{Z}_j^1} F_k, \sum_{k \in \mathcal{Z}^1} e^{kd} S_k \right),$$

implying the stationarity of  $\{(\Theta_n, R_n, D_n'')\}_{n \in \mathbb{Z}}$  and hence of  $R$  and  $\Xi$ .

The process  $R$  can jump to 0 in a single step and has positive jumps of size at most  $1/(\alpha\rho) - d$ , both with positive probability. Hence it will hit any subinterval of its state space  $[0, \infty)$  from any starting point in a finite number of steps with positive probability, making it  $\varphi$ -irreducible [24, p. 82] with respect to its invariant law.

Since  $\Theta_n$  is independent of  $(R_n, D_n'')$ , the  $\varphi$ -irreducibility of  $\{\Xi_n\}_{n \in \mathbb{Z}}$  follows if, starting from an arbitrary point, we can prove that the process  $\{(R_n, D_n'')\}_{n \in \mathbb{Z}}$  hits any rectangle in the product

$[0, \infty) \times (0, \infty)$  with positive probability. Since we already know that  $R$  hits intervals and has (arbitrarily) small positive jumps with positive probability, the independence of  $\{D''_n\}_{n \in \mathbb{Z}}$  and  $R$ , together with the fact that  $D''_n$  has a positive density, imply the final statement of the lemma.  $\square$

*Proof of Theorem 1.2.* By Lemma 3.2(ii),  $\Xi$  is  $\pi$ -irreducible, where  $\pi$  denotes the invariant law of  $\Xi$ . Hence, by [24, Prop. 10.1.1],  $\Xi$  is recurrent, meaning that the expected number of visits of the chain  $\Xi$  to any set charged by  $\pi$  is infinite for all starting points. By [24, Thm 9.0.1], the chain  $\Xi$  is Harris recurrent on a complement of a  $\pi$ -null set. Put differently, for any starting point, the number of visits  $\Xi$  makes to any set charged by  $\pi$  is infinite almost surely.

Consider the Markov chain  $\Psi = \{\Psi_n\}_{n \in \mathbb{N}}$ , where  $\mathbb{N} = \{0, 1, \dots\}$  and  $\Psi_n = \Xi_{-n}$ . In the language of [28],  $\Psi$  is a chain dual to  $\Xi$  with respect to  $\pi$ . In particular, the invariant law of  $\Psi$  equals  $\pi$ . Since  $\Xi$  is Harris recurrent on a state space with a countably generated  $\sigma$ -algebra, [28, Thm 8.1.1] implies that there exists a modification of  $\Psi$  (again denoted by  $\Psi$ ) that is also Harris recurrent. Since  $\mathbb{P}(a(\Theta_{-n}) \geq D'_{-n}) > 0$  for any  $n \in \mathbb{N}$ , it follows that the  $\Psi$ -stopping time  $\sigma' = \inf\{n > 0 : a(\Theta_{-n}) \geq D'_{-n}\}$  is finite almost surely. Moreover, by [24, Thm 11.1.4] we have  $\mathbb{E}[\sigma' | \Psi_0] < \infty$  almost surely.

Recall that  $\sigma = \inf\{n > 0 : a(\Theta_{-n}) \geq D_{-n}\}$  is the number of steps taken backwards in time in Algorithm 1. By Lemma 3.2(i) we have  $\sigma \leq \sigma'$ . Since, by definition  $\Psi_0 = \Xi_0$ , the claim follows.  $\square$

#### 4. Backward Simulation of $\{(D_n, \Theta_n)\}_{n \in \mathbb{Z}}$

A key step in Algorithm 1 consists of simulating the process  $\{(D_n, \Theta_n)\}_{n \in \mathbb{Z}}$  backwards in time until the random time  $\sigma = \inf\{n > 0 : a(\Theta_{-n}) \geq D_{-n}\}$  (see (2.3) and (3.3) for the definitions of  $\mathcal{Z}^1$  and  $a(\theta)$ , respectively). The forthcoming Algorithm 2 is responsible for this step. Recall that  $\{\Theta_n\}_{n \in \mathbb{Z}}$  is an i.i.d. sequence with  $\Theta_n = (S_n, U_n, W_n, \Lambda_n)$  having independent components, where  $S_n, U_n$  and  $\Lambda_n$  are distributed as in Theorem 1.1 and  $W_n \sim U(0, 1)$ .

At time  $n \in \mathbb{Z}$ , the dominating process  $D$  in (3.8) depends on three components: the sequence  $(\chi_n, \{S_k\}_{k \in \mathcal{Z}_{\chi_n}^0})$ , the all-time maximum  $\sup_{k \in \mathcal{Z}^{n+1}} \{C_k\}$  and  $C_n$  (via the reflected process  $R$ , see (3.6)-(3.7)) and the uniform random variables  $\{U_k\}_{k \in \mathcal{Z}_{\chi_n}^0}$ . The time  $\chi_n$  in (3.5) is the last time before  $n$  the random variables  $\{S_k\}_{k \in \mathcal{Z}^0}$  exceed a certain adaptive exponential bound. Algorithm 3 for sampling  $(\chi_n, \{S_k\}_{k \in \mathcal{Z}_{\chi_n}^0})$  is given in Section 4.1 below. A sample for  $(R_n, C_n)$  requires the joint forward simulation of the dual random walk  $-C$  and its ultimate maximum. This problem



was solved in [4]. The algorithm in [4], stated for completeness as Algorithm 7 of Section 4.2 below for the random walk  $C$  in (3.6), requires the simulation of the walk under the exponential change of measure.

Since the increments of  $C$  are shifted negative exponential random variables under the original measure, they remain in the same class under the exponential change of measure, making the simulation in Algorithm 7 simple. Finally, heaving simulated  $(R, C)$  backwards in time, we need to recover the random variables  $\Lambda_k$  and  $U_k$ , conditional on the values of increments  $F_k = d + (1/\alpha) \log(U_n \Lambda_n)$  we have observed. Algorithm 8 in Section 4.3 below describes this step.

---

**Algorithm 2** Backward simulation of  $(\sigma, \{(D_n, \Theta_n)\}_{n \in \mathcal{Z}_{-\sigma}^0})$

---

- 1: Sample  $\chi_{-1}$  and  $\{S_k\}_{k \in \mathcal{Z}_{\chi_{-1}}^0}$  ▷ Algorithm 3
  - 2: Sample  $\{(R_k, C_k, \Lambda_k, U_k)\}_{k \in \mathcal{Z}_{N_{-1}}^0}$  for some  $N_{-1} \leq \chi_{-1}$  ▷ Algorithms 7 & 8
  - 3: Bundle up  $\{\Theta_k\}_{k \in \mathcal{Z}_{\chi_{-1}}^0}$  and compute  $D_{-1}$
  - 4: Put  $n := -1$
  - 5: **while**  $D_n > a(\Theta_n)$  **do**
  - 6:   Put  $n := n - 1$
  - 7:   Sample  $\chi_n$  and  $\{S_k\}_{k \in \mathcal{Z}_{\chi_n}^{x_{n+1}}}$  conditional on  $(\chi_{n+1}, \{S_k\}_{k \in \mathcal{Z}_{\chi_n}^{x_{n+1}}})$  ▷ Algorithm 3
  - 8:   Sample  $\{(R_k, C_k, \Lambda_k, U_k)\}_{k \in \mathcal{Z}_{N_n}^{N_{n+1}}}$  for some  $N_n \leq \chi_n$  ▷ Algorithms 7 & 8
  - 9:   Bundle up  $\{\Theta_k\}_{k \in \mathcal{Z}_{\chi_n}^{x_{n+1}}}$ , and compute  $D_n$
  - 10: **end while**
  - 11: Put  $\sigma = -n$
  - 12: **return**  $(\sigma, \{\Theta_k\}_{k \in \mathcal{Z}_{-\sigma}^0})$
- 

The number of steps  $N_{-1}$  (resp.  $N_n$ ) in line 2 (resp. 8) of Algorithm 2 is random since Algorithm 7, which outputs the all-time maximum of the random walk, may need more values of the random walk than required to recover the previous value of the dominating process  $D_{-1}$  (resp.  $D_n$ ). (In the notation of Section 4.2 below, the integers  $N_n$  take the form  $\Delta(\tau_m)$ .) The running time of Algorithm 3 is random but has moments of all orders (see Lemma 4.1 in Section 4.1 below). Algorithm 8 executes a loop of length equal to the number of steps in the random walk  $C$  the algorithm is applied to, with each step sampling one Poisson and one Beta random variables (see Section 4.3 below). Hence both Algorithms 3 and 8 are fast (see Section 5). Algorithm 7 of [4]

(see Section 4.2 below) runs sequentially Algorithms 4, 5 and 6. Each of these algorithms is reliant on rejection sampling and has a finite expected running time, which is easy to quantify in terms of the increments of the walk  $C$ .

#### 4.1. Simulation of $(\mathcal{X}_n, \{\mathbf{S}_k\}_{k \in \mathcal{Z}_{\mathcal{X}_n}^0})$

Consider independent Bernoulli random variables  $\{J_n\}_{n=1}^\infty$  with computable  $p_n = \mathbb{P}(J_n = 0)$ ,  $n \geq 1$ , satisfying  $\sum_{n=1}^\infty (1 - p_n) < \infty$ . By the Borel-Cantelli Lemma the random time  $\tau = \sup\{n \geq 0 : J_n = 1\}^+$  (with convention  $\sup \emptyset = -\infty$ ) satisfies  $\tau \in \mathbb{N}$  a.s. Clearly,  $J_n = 0$  for all  $n > \tau$ , and  $\{\tau < n\} = \bigcap_{k=n}^\infty \{J_k = 0\}$  implies  $\mathbb{P}(\tau < n) = \prod_{k=n}^\infty p_k$ . If there exists  $n^* \geq 1$  such that for all  $n \geq n^*$  we have a positive computable lower bound  $q_n \leq \prod_{k=n}^\infty p_k$ , then we can simulate  $(\tau, \{J_k\}_{k \in \{0, \dots, \tau\}})$  as follows.

Define the auxiliary function  $F : (0, 1) \times (0, 1) \rightarrow \{0, 1\} \times (0, 1)$  by the formula

$$F(u, p) = \begin{cases} \left(0, \frac{u}{p}\right) & \text{if } u \leq p, \\ \left(1, \frac{u-p}{1-p}\right) & \text{if } u > p. \end{cases}$$

The following observation is simple but crucial: for any  $p \in (0, 1)$  and  $U \sim U(0, 1)$ , the components of the vector  $(J, V) = F(U, p)$  are independent,  $J$  is Bernoulli with  $\mathbb{P}(J = 0) = p$  and  $V \sim U(0, 1)$ .

Sample  $\{J_n\}_{n \in \mathcal{Z}_1^{n^*}}$  and an independent  $U^{(n^*)} \sim U(0, 1)$ . Let  $(J_{n^*}, U^{(n^*+1)}) = F(U^{(n^*)}, p_{n^*})$ . Hence  $J_{n^*}$  has the correct distribution and is independent of  $U^{(n^*+1)} \sim U(0, 1)$ . Thus,  $J_{n^*}$  is independent of  $F(U^{(n^*+1)}, p_{n^*+1}) = (J_{n^*+1}, U^{(n^*+2)})$ . Define recursively  $(J_n, U^{(n+1)}) = F(U^{(n)}, p_n)$  for  $n \geq n^* + 2$  and note that the sequence  $\{J_n\}_{n \in \mathbb{N}}$  of Bernoulli random variables is i.i.d. Moreover, the sequence  $\{U^{(n)}\}_{n \geq n^*}$  detects the value of  $\tau$  since  $\{U^{(n)} \leq q_n\} \subseteq \{U^{(n)} \leq \prod_{k=n}^\infty p_k\} = \{\tau < n\}$ .

Algorithm 3 samples a single uniform random variable and performs a binary search. Its running time  $\varsigma = \inf\{n \geq n^* : U^{(n)} \leq q_n\} \geq \tau + 1$  (with convention  $\inf \emptyset = \infty$ ) has the following properties.

**Lemma 4.1.** (a) *If  $\lim_{n \rightarrow \infty} q_n = 1$  then  $\mathbb{P}(\varsigma < \infty) = 1$ .*

(b) *If  $\sum_{n=n^*}^\infty (1 - q_n) < \infty$  then  $\mathbb{E}\varsigma < \infty$ .*

(c) *If  $\sum_{n=n^*}^\infty (1 - q_n) e^{tn} < \infty$  for some  $t > 0$ , then  $\mathbb{E}e^{t\varsigma} < \infty$ .*

(d) *If  $q_n p_{n-1} \geq q_{n-1}$  for  $n > n^*$ , then the converses of (a), (b) and (c) are also true.*

**Remark 4.1.** At the cost of additional operations, one may always construct a sequence  $\{q'_n\}_{n=n^*}^\infty$  that satisfies (d). Indeed, let  $q'_{n^*} = q_{n^*}$  and define recursively  $q'_n = \max\{q_n, q'_{n-1}/p_{n-1}\}$  for  $n > n^*$ ,

---

**Algorithm 3** Simulation of  $(\tau, \{J_k\}_{k \in \{1, \dots, \tau\}})$ 


---

- 1: Sample  $J_1, \dots, J_{n^*-1}$  and put  $n := n^* - 1$
  - 2: Sample  $U \sim U(0, 1)$
  - 3: **loop**
  - 4:   Put  $n := n + 1$
  - 5:   **if**  $U > p_n$  **then**
  - 6:     Put  $J_n := 1$  and update  $U := \frac{U - p_n}{1 - p_n}$
  - 7:   **else if**  $U \leq q_n$  **then**
  - 8:     Compute  $\tau$  from  $J_1, \dots, J_{n-1}$  and **exit loop**
  - 9:   **else**
  - 10:    Put  $J_n := 0$  and update  $U := \frac{U}{p_n}$
  - 11:   **end if**
  - 12: **end loop**
  - 13: **return**  $(\tau, \{J_k\}_{k \in \{1, \dots, \tau\}})$
- 

then these satisfy condition (d), are computable and inductively satisfy  $q'_n \leq \prod_{k=n}^{\infty} p_k$  for  $n \geq n^*$ . This consideration shows that our conditions are sharp.

- Proof.* (a) For all  $n \geq n^*$  we have  $\{\zeta \leq n\} \supseteq \{U^{(n)} \leq q_n\}$ , then  $\mathbb{P}(\zeta > n) \leq \mathbb{P}(U^{(n)} > q_n) = 1 - q_n$ . Hence  $\mathbb{P}(\zeta = \infty) = \lim_{n \rightarrow \infty} \mathbb{P}(\zeta > n) \leq \lim_{n \rightarrow \infty} (1 - q_n) = 0$  and the sufficiency follows.
- (b) Similarly,  $\mathbb{E}\zeta = \sum_{n=0}^{\infty} \mathbb{P}(\zeta > n) \leq n^* + \sum_{n=n^*}^{\infty} (1 - q_n) < \infty$  and the claim follows.
- (c) Note that  $(e^t - 1) \sum_{m=0}^{n-1} e^{tm} = e^{tn} - 1$ . Exchanging the order of summation in the third equality of the following estimate implies (c):

$$\begin{aligned} \mathbb{E}e^{t\zeta} &= \sum_{n=0}^{\infty} \mathbb{P}(\zeta = n) e^{tn} = \sum_{n=0}^{\infty} \mathbb{P}(\zeta = n) \left( 1 + (e^t - 1) \sum_{m=0}^{n-1} e^{tm} \right) \\ &= 1 + (e^t - 1) \sum_{m=0}^{\infty} e^{tm} \mathbb{P}(\zeta > m) \leq e^{tn^*} + (e^t - 1) \sum_{n=n^*}^{\infty} (1 - q_n) e^{tn} < \infty. \end{aligned}$$

- (d) Condition (d) and the relation  $(\tau + 1) \vee n^* = \inf\{k \geq n^* : U^{(k)} \leq \prod_{j=k}^{\infty} p_j\}$  imply for  $n \geq k \geq n^*$ ,

$$\{(\tau + 1) \vee n^* = k, \zeta \leq n\} = \begin{cases} \left\{ U^{(k-1)} \in \left[ p_{k-1}, p_{k-1} + (1 - p_{k-1})q_n \prod_{j \in \mathcal{Z}_k^n} p_j \right] \right\} & k > n^*, \\ \left\{ U^{(n^*)} \in \left[ 0, q_n \prod_{j \in \mathcal{Z}_{n^*}^n} p_j \right] \right\} & k = n^*. \end{cases}$$

Thus, a simple calculation yields

$$\mathbb{P}(\varsigma \leq n) = q_n \prod_{j \in \mathcal{Z}_{n^*}^n} p_j + \sum_{k \in \mathcal{Z}_{n^*}^n} q_n(1-p_k) \prod_{j \in \mathcal{Z}_{k+1}^n} p_j = q_n,$$

and the result follows from standard probability theory.  $\square$

In Algorithm 2 we are required to sample  $(\chi_0, \{S_k\}_{k \in \mathcal{Z}_{\chi_0}^0})$ , and then, iteratively for  $n \in \mathcal{Z}^0$ ,  $\chi_n$  and the remaining  $\{S_k\}_{k \in \mathcal{Z}_{\chi_n}^n}$ , given the known values  $(\chi_{n+1}, \{S_k\}_{k \in \mathcal{Z}_{\chi_{n+1}}^0})$ . To apply Algorithm 3, we need a computable lower bound on the product of probabilities  $p(m) = \mathbb{P}(S_1 \leq e^{\delta m})$ ,  $m \in \mathbb{N}$ . Recall the exponential lower bound on  $p(m)$  in (3.4) and define  $m^* = \left\lfloor \frac{1}{\delta\gamma} \log \mathbb{E}S_1^\gamma \right\rfloor^+ + 1$  (here  $\lfloor x \rfloor = \sup\{n \in \mathbb{Z} : n \leq x\}$  for any  $x \in \mathbb{R}$ ). Note that for any  $m \geq m^*$  we have  $e^{-\delta\gamma m} \mathbb{E}S_1^\gamma < 1$  and may hence define  $\bar{p}(m) = \exp\left(-\frac{1}{1-e^{-\delta\gamma}} \frac{e^{-\delta\gamma m} \mathbb{E}S_1^\gamma}{1-e^{-\delta\gamma m} \mathbb{E}S_1^\gamma}\right) \in (0, 1)$ . The inequality in (3.4) implies

$$\begin{aligned} \prod_{j=m}^{\infty} p(j) &\geq \prod_{j=m}^{\infty} (1 - e^{-\delta\gamma j} \mathbb{E}S_1^\gamma) = \exp\left(\sum_{j=m}^{\infty} \log(1 - e^{-\delta\gamma j} \mathbb{E}S_1^\gamma)\right) \\ &= \exp\left(-\sum_{j=m}^{\infty} \sum_{k=1}^{\infty} \frac{1}{k} e^{-\delta\gamma jk} (\mathbb{E}S_1^\gamma)^k\right) \geq \exp\left(-\sum_{k=1}^{\infty} \frac{e^{-\delta\gamma mk} (\mathbb{E}S_1^\gamma)^k}{1 - e^{-\delta\gamma k}}\right) \geq \bar{p}(m). \end{aligned}$$

Since for any  $k \in \mathcal{Z}_{\chi_0}^0$  we have  $\mathbb{P}(I_k^0 = 0) = \mathbb{P}(S_k \leq e^{-(k+1)\delta}) = p(-(k+1))$ , Algorithm 3 can be applied (with  $n^* = m^*$ ) to sample the sequence  $\{I_k^0\}_{k \in \mathcal{Z}_{\chi_0}^0}$ . Moreover, for  $m \in \mathbb{N}$  we get

$$\bar{p}(m^* + m) \geq \exp(-re^{-\delta\gamma m}) \geq 1 - re^{-\delta\gamma m}, \quad \text{where } r = \frac{e^{-\delta\gamma m^*} \mathbb{E}S_1^\gamma}{(1 - e^{-\delta\gamma})(1 - e^{-\delta\gamma m^*} \mathbb{E}S_1^\gamma)} > 0.$$

Hence, for any  $t \in (0, \delta\gamma)$ , Lemma 4.1(c) implies that the running time  $\varsigma$  satisfies  $\mathbb{E}[e^{t\varsigma}] < \infty$  and therefore possesses moments of all orders. Having obtained  $(\chi_0, \{I_k^0\}_{k \in \mathcal{Z}_{\chi_0}^0})$ , for  $k \in \mathcal{Z}_{\chi_0}^0$ , we sample  $S_k$  as  $S^+(\alpha, \rho)$  conditional on  $S_k \leq e^{-\delta(k+1)}$  (if  $I_k^0 = 0$ ) or  $S_k > e^{-\delta(k+1)}$  (if  $I_k^0 = 1$ ), yielding a sample of  $(\chi_n, \{S_k\}_{k \in \mathcal{Z}_{\chi_n}^n})$ .

Assume now that we have already sampled  $(\chi_{n+1}, \{S_k\}_{k \in \mathcal{Z}_{\chi_{n+1}}^0})$ . The adaptive exponential bounds in the indicators  $I_k^{n+1}$  and  $I_k^n$  are different (see Figure 4.1) and the relevant probabilities take the form

$$p'(m) = \mathbb{P}\left(S_1 \leq e^{\delta m} \mid S_1 \leq e^{\delta(m+1)}\right), \quad m \in \mathbb{N}.$$

Since  $\{S_1 \leq e^{\delta m}\} \subset \{S_1 \leq e^{\delta(m+1)}\}$ , the inequality  $p'(m) \geq p(m)$  holds for any  $m \in \mathbb{N}$ . Thus

$$\prod_{j=m}^{\infty} p'(j) \geq \prod_{j=m}^{\infty} p(j) \geq \bar{p}(m)$$

and Algorithm 3 can be applied with  $n^* = \max\{m^*, n - \chi_{n+1}\}$ . The same argument as above shows that the running time  $\varsigma$  has moments of all orders.

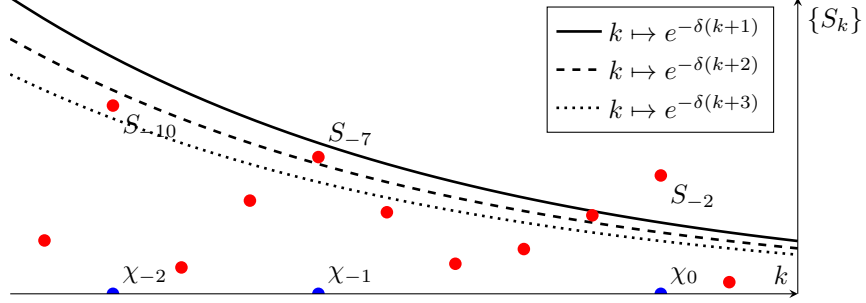


FIGURE 4.1: The adaptive exponential bounds  $k \mapsto e^{\delta(n-k-1)}$  for  $n \in \{0, -1, -2\}$  with the corresponding stable random variables conditioned to be positive  $\{S_k\}_{k \in \mathcal{Z}^0}$  and the times  $\{\chi_k\}_{k \in \mathcal{Z}^1}$  used for the construction of the dominating process  $\{D_n\}_{n \in \{0, -1, -2\}}$  in (3.8).

#### 4.2. Simulation of the Random Walk and Its Reflected Process from [4]

In this section we present an overview of the algorithm in [4] for the joint simulation of  $(C, R)$  defined in (3.6)-(3.7). We refer to [4] and [17] for the proofs (the latter paper contains the simulation algorithm for the ultimate maximum of a random walk with negative drift and provides a basis for the simulation algorithm in [4]).

Let  $\eta = \eta(d)$  be the unique positive root of  $\psi_d(\eta) = 0$ , where  $\psi_d(t) = \log(\mathbb{E}e^{tF_0}) = dt - \log(1 + t/(\alpha\rho))$ . Note that  $\psi'_d(\eta) = d - \frac{1}{\alpha\rho + \eta} > 0$  and  $\eta = -\alpha\rho - W_{-1}(-\alpha\rho d e^{-\alpha\rho d})/d$ , where  $W_{-1}$  is the secondary branch of the Lambert W function. Since  $\mathbb{E}[\exp(\eta F_n)] = 1$  for all  $n \in \mathbb{Z}$ , the process  $\{\exp(\eta C_n)\}_{n \in \mathcal{Z}^1}$  is a positive backward martingale started at one, thus inducing a probability measure  $\mathbb{P}^\eta$  on  $\sigma$ -algebras  $\sigma(C_k; k \in \mathcal{Z}_n^1)$ ,  $n \in \mathcal{Z}^1$ , by the formula  $\mathbb{P}^\eta(A) = \mathbb{E}[1_A e^{\eta C_n}]$  where  $A \in \sigma(C_k; k \in \mathcal{Z}_n^1)$ . Under  $\mathbb{P}^\eta$ , the process  $C$  remains a random walk with i.i.d. increments satisfying  $(\alpha\rho + \eta)(d - F_n) \sim \text{Exp}(1)$ . Hence  $\mathbb{E}^\eta[C_{-1}] = \psi'_d(\eta) > 0$ , implying  $\lim_{n \rightarrow -\infty} C_n = \infty$ ,  $\mathbb{P}^\eta$ -a.s. by the strong law of large numbers.

For any  $k \in \mathbb{Z}$  define (with convention  $\sup \emptyset = -\infty$ )

$$T_x^k = \begin{cases} \sup \{n \in \mathcal{Z}^k : C_n - C_k > x\} & \text{if } x > 0, \\ \sup \{n \in \mathcal{Z}^k : C_n - C_k < x\} & \text{if } x < 0. \end{cases} \quad (4.1)$$

For ease of notation we let  $T_x = T_x^0$ . Let  $E$  be an independent exponential random variable with

mean one. Then, for  $x > 0$ , we have  $\mathbb{P}(R_0 > x) = \mathbb{P}^\eta(L_{E/\eta} > x)$ , where  $L_x = \inf\{y \geq 0 : C_{T_y} > x\}$  is the right inverse of  $x \mapsto C_{T_x}$ , see e.g. [17]. Hence for  $x \in (0, x')$ , where  $x' \leq \infty$ , sampling  $1_{\{R_0 > x\}} = 1_{\{T_x > -\infty\}}$ , conditional on  $1_{\{R_0 \leq x'\}} = 1_{\{T_{x'} = -\infty\}}$ , in finite time amounts to sampling  $E$  and  $C_{-1}, \dots, C_{T_{E/\eta}}$  under  $\mathbb{P}^\eta$ , see Algorithm 4 below.

---

**Algorithm 4** Simulation of  $1_{\{R_0 > x\}}$  conditional on  $\{R_0 \leq x'\}$

---

**Require:**  $\infty \geq x' > x > 0$

```

1: loop
2:   Sample  $E \sim \text{Exp}(1)$ 
3:   if  $E/\eta \leq x$  then
4:     return 0
5:   else
6:     Sample  $C_0 = 0, C_{-1}, \dots, C_{T_{E/\eta}}$  under  $\mathbb{P}^\eta$ 
7:     Compute  $L_{E/\eta}$ 
8:     if  $L_{E/\eta} \leq x'$  then ▷ Accept sample
9:       return  $1_{\{L_{E/\eta} > x\}}$ 
10:    end if
11:  end if
12: end loop

```

---

**Remark 4.2.** Since  $L_x \leq x$ , then the condition  $E/\eta \leq x$  implies  $L_{E/\eta} \leq x$ , thus identifying  $1_{\{L_{E/\eta} > x\}} = 0$  (see line 3) and saving the computational effort of running all subsequent lines. This algorithm repeats independent experiments with success probability  $\mathbb{P}^\eta(L_{E/\eta} \leq x') > 0$ . The expected runtime of each iteration in the loop is bounded above by  $(\eta^{-1} + d)/\psi'_d(\eta)$ , see [17, Eq. (2.3)]. Hence the expected running time of Algorithm 4 is finite.

In Algorithm 7 below we need to sample the path of the random walk  $\{C_k\}_{k \in \mathbb{Z}_{T_x}^1}$  conditioned on the event  $\{R_0 \in (x, x')\}$ , where  $0 < x < x' \leq \infty$ . By a rejection sampling method under  $\mathbb{P}^\eta$  and Algorithm 4 (see [4, Lemma 3]), this can be achieved as follows.

**Remark 4.3.** Since  $L_x \leq x$ , we have  $\mathbb{P}(R_0 \leq z) \geq \mathbb{P}(E/\eta \leq z) = 1 - \exp(-z\eta)$  for all  $z \geq 0$ . Since the overshoot  $C_{T_x} - x$  is in the interval  $(0, d)$ , the expected running time of Algorithm 5 (i.e. one over the acceptance probability) is smaller than  $\exp(\eta(x+d))/(1 - \exp(-\eta(x' - x - d)))$  if  $x' > x + d$ .

---

**Algorithm 5** Simulation of  $C_0, \dots, C_{T_x}$  conditional on  $\{T_x > -\infty = T_{x'}\}$

---

**Require:**  $\infty \geq x' > x > 0$

- 1: **loop**
  - 2:   Sample  $C_0 = 0, C_{-1}, \dots, C_{T_x}$  under  $\mathbb{P}^\eta$
  - 3:   Given  $C_{T_x}$ , sample independent  $1_{\{R'_0 \leq x' - C_{T_x}\}}$  and  $U \sim U(0, 1)$  ▷ Algorithm 4
  - 4:   **if**  $U \leq \exp(-\eta C_{T_x})$  and  $1_{\{R'_0 \leq x' - C_{T_x}\}} = 1$  **then** ▷ Accept sample
  - 5:     **return**  $\{C_n\}_{n \in \mathcal{Z}_{T_x}^1}$
  - 6:   **end if**
  - 7: **end loop**
- 

In Algorithm 7 we also need to simulate the path of the walk reaching a negative level  $-x$ , while staying below a given positive level forever. Algorithm 6 achieves this (see [4, Lemma 3]). Its expected running time is bounded above by  $1/((1 - \exp(-\eta(x' + x)))\mathbb{P}(T_{-x} < T_{x'})) < \infty$ .

---

**Algorithm 6** Simulation of  $C_0, \dots, C_{T_{-x}}$  conditional on  $\{T_{x'} = -\infty\}$

---

**Require:**  $x \in (0, \infty)$  &  $x' \in (0, \infty]$

- 1: **loop**
  - 2:   Sample  $C_0 = 0, C_{-1}, \dots, C_{T_{-x}}$  under  $\mathbb{P}$
  - 3:   Given  $C_{T_{-x}}$ , sample an independent  $1_{\{R'_0 \leq x' - C_{T_{-x}}\}}$  ▷ Algorithm 4
  - 4:   **if**  $1_{\{R'_0 \leq x' - C_{T_{-x}}\}} = 1$  and  $\max_{n \in \mathcal{Z}_{T_{-x}}^1} \{C_n\} \leq x'$  **then** ▷ Accept sample
  - 5:     **return**  $\{C_n\}_{n \in \mathcal{Z}_{T_{-x}}^1}$
  - 6:   **end if**
  - 7: **end loop**
- 

We now give a brief overview of the algorithm in [4] for the simulation of  $\{(C_n, R_n)\}_{n \in \mathcal{Z}^1}$ . Pick  $\kappa > \max\{\log(2)/(3\eta), 1/(\alpha\rho)\}$  (see assumption in [4, Prop. 3]). [4] constructs sequences  $\Delta = \{\Delta(k)\}_{k \geq 0}$  and  $\tau = \{\tau_k\}_{k \geq 0}$  of decreasing negative and increasing positive times, respectively:

1. at the start of each iteration of the algorithm we are given

$$\left( \{\tau_k\}_{k \in \{0, \dots, m\}}, \{\Delta(k)\}_{k \in \{0, \dots, \tau_m\}}, \{C_n\}_{n \in \mathcal{Z}_{\Delta(\tau_m)}^1}, \{R_n\}_{n \in \mathcal{Z}_{\Delta(\tau_{m-1})}^1} \right),$$

2. at each iteration we sample

$$\left( \tau_{m+1}, \{\Delta(k)\}_{k \in \{\tau_m+1, \dots, \tau_{m+1}\}}, \{C_n\}_{n \in \mathcal{Z}_{\Delta(\tau_{m+1})}^{\Delta(\tau_m)}}, \{R_n\}_{n \in \mathcal{Z}_{\Delta(\tau_{m+1}-1)}^{\Delta(\tau_{m-1})}} \right).$$

Note that at the  $m$ -th iteration we have  $\Delta(\tau_m) - \Delta(\tau_m - 1)$  more values of the walk than of the reflected process. More precisely, the algorithm starts by setting  $\Delta(0) = 0$  and repeats the following steps: given  $\{\tau_k\}_{k \in \{0, \dots, m\}}$  and  $\{\Delta(k)\}_{k \in \{0, \dots, \tau_m\}}$ , then put  $\Delta(\tau_m + 1) = T_{-2\kappa}^{\Delta(\tau_m)}$ . Next, if  $\Delta(k)$  is the last known value of  $\Delta$  and if  $R_{\Delta(k)} > \kappa$ , then put  $\Delta(k + 1) = T_{\kappa}^{\Delta(k)}$  and  $\Delta(k + 2) = T_{-2\kappa}^{\Delta(k+1)}$ . If instead  $R_{\Delta(k)} \leq \kappa$  then put  $\tau_{m+1} = k$ . Repeat the previous two steps until we can compute  $\tau_{m+1}$ , that is, until  $R_{\Delta(k)} \leq \kappa$ . After computing  $\tau_{m+1}$  go back and repeat. By construction (see Proposition 3 in [4]) we have

$$\sup_{n \in \mathcal{Z}_{\Delta(\tau_m)+1}^{\Delta(\tau_m)+1}} \{C_n\} \leq C_{\Delta(\tau_m-1)} - \kappa, \quad \text{implying} \quad R_n = \max_{k \in \mathcal{Z}_{\Delta(\tau_m)+1}^{\Delta(\tau_m)+1}} \{C_k\} - C_n, \quad n \in \mathcal{Z}_{\Delta(\tau_m-1)}^{\Delta(\tau_m-1)}.$$

Hence, we may compute  $R_n$ ,  $n \in \mathcal{Z}_{\Delta(\tau_m-1)}^1$ , from the simulated values  $\tau_m, \Delta(\tau_m-1), \Delta(\tau_m), \{C_n\}_{n \in \mathcal{Z}_{\Delta(\tau_m)}^1}$ .

---

**Algorithm 7** Simulation of the random walk and its reflected process

---

**Require:**  $\kappa > \max\{\frac{\log(2)}{3\eta}, \frac{1}{\alpha\rho}\}$ ,  $d \in (0, 1)$ ,  $\infty \geq x > 0$  and  $m \geq 1$      $\triangleright x$  is an upper bound for  $R_0$

```

1: Put  $t := C_0 := \Delta(0) := \tau_0 := 0$ 
2: for  $k \in \{1, \dots, m\}$  do
3:   Put  $t := \tau_{k-1}$ 
4:   loop
5:     Sample  $C_{\Delta(t)-1}, \dots, C_{T_{-2\kappa}^{\Delta(t)}}$  conditioned on  $\{R_{\Delta(t)} < x - C_{\Delta(t)}\}$      $\triangleright$  Algorithm 6
6:     Put  $\Delta(t+1) := T_{-2\kappa}^{\Delta(t)}$  and  $t := t+1$ 
7:     Sample  $1_{\{R_{\Delta(t)} > \kappa\}}$  given  $\{R_{\Delta(t)} < x - C_{\Delta(t)}\}$      $\triangleright$  Algorithm 4
8:     if  $1_{\{R_{\Delta(t)} > \kappa\}} = 1$  then
9:       Sample  $C_{\Delta(t)-1}, \dots, C_{T_{\kappa}^{\Delta(t)}}$  from  $\mathbb{P}^\eta$      $\triangleright$  Algorithm 5
10:      Put  $\Delta(t+1) := T_{\kappa}^{\Delta(t)}$  and  $t := t+1$ 
11:     else
12:       Put  $x := \kappa + C_{\Delta(t)}$ ,  $\tau_k := t$  and exit loop
13:     end if
14:   end loop
15: end for
16: Compute  $\{R_n\}_{n \in \mathcal{Z}_{\Delta(\tau_m-1)}^1}$ 
17: return  $\left( \{\tau_k\}_{k \in \{0, \dots, m\}}, \{\Delta(k)\}_{k \in \{0, \dots, \tau_m\}}, \{C_n\}_{n \in \mathcal{Z}_{\Delta(\tau_m)}^1}, \{R_n\}_{n \in \mathcal{Z}_{\Delta(\tau_m-1)}^1} \right)$ 

```

---



### 4.3. Sampling $(U_n, \Lambda_n)$ given $F_n$

Algorithm 2 requires the knowledge of  $\{(U_n, \Lambda_n)\}_{n \in \mathbb{Z}^0}$ , given the increments  $\{F_n\}_{n \in \mathbb{Z}^0}$  of the random walk  $C$ . Since  $\log(U_n \Lambda_n) = \alpha(F_n - d)$  for all  $n \in \mathbb{Z}$ , by independence, we may restrict attention to  $n = 1$ . It follows from (2.6) above that  $\Lambda_1 \stackrel{d}{=} \prod_{i=2}^T U_i$  for an independent geometric random variable  $T$  with parameter  $\rho$  on the positive integers (if  $T = 1$  the right-hand side is defined to equal one). Hence, by independence, we have  $U_1 \Lambda_1 \stackrel{d}{=} \prod_{i=1}^T U_i$ . By (2.7),  $-\log \Lambda_1$  conditioned on being positive is exponential with mean  $1/\rho$ . Hence for any  $n \geq 1$  and  $y > 0$  we obtain

$$\mathbb{P} \left[ T = n \mid -\sum_{i=1}^T \log(U_i) = y \right] = \frac{\rho(1-\rho)^{n-1} \frac{y^{n-1} e^{-y}}{(n-1)!}}{\rho e^{-\rho y}} = \frac{[(1-\rho)y]^{n-1} e^{-(1-\rho)y}}{(n-1)!}.$$

Thus the conditional law of  $T - 1$  given  $\sum_{i=1}^T \log(U_i) = -y$  is Poisson with mean  $(1-\rho)y$ . If  $T = 1$ , then  $-\log(U_1) = y$  and  $\Lambda_1 = 1$ . If  $T > 1$ , then for  $x \in (0, y)$  we get

$$\mathbb{P} \left[ -\log(U_1) \in dx \mid T = n, -\sum_{i=1}^T \log(U_i) = y \right] = \frac{e^{-x} \frac{(y-x)^{n-2} e^{-(y-x)}}{(n-2)!}}{\frac{y^{n-1} e^{-y}}{(n-1)!}} dx = (n-1) \frac{(y-x)^{n-2}}{y^{n-1}} dx.$$

Hence, conditional on  $T = n$  and  $\log\left(\prod_{i=1}^T U_i\right) = -y$ , the law of  $-\frac{1}{y} \log(U_1)$  is *Beta*(1,  $n-1$ ) (understood as the Dirac measure  $\delta_1$  when  $n = 1$ ). Finally we set  $\Lambda_1 = \exp(\alpha(F_1 - d))/U_1$ .

---

**Algorithm 8** Simulation of  $\{(U_k, \Lambda_k)\}_{k \in \mathbb{Z}_m^n}$  given  $\{F_k\}_{k \in \mathbb{Z}_m^n}$

---

**Require:**  $\{F_k\}_{k \in \mathbb{Z}_m^n}$  for  $m, n \in \mathbb{Z}$  and  $m < n$ .

- 1: **for**  $k \in \mathbb{Z}_m^n$  **do**
  - 2:     Sample  $T - 1 \sim \text{Poisson}(-\alpha(F_k - d)(1 - \rho))$
  - 3:     Sample  $L \sim \text{Beta}(1, T - 1)$
  - 4:     Let  $U_k := \exp(L\alpha(F_k - d))$  and  $\Lambda_k := \exp((1 - L)\alpha(F_k - d))$
  - 5: **end for**
  - 6: **return**  $\{(U_k, \Lambda_k)\}_{k \in \mathbb{Z}_m^n}$
- 

## 5. Implementation

Recall the definitions of the process  $\{(C_n, F_n)\}_{n \in \mathbb{Z}}$  in 3.6, of  $\{\Theta_n\}_{n \in \mathbb{Z}}$  in the first paragraph of Section 4 and of  $\mathbb{P}^\eta$  in the second paragraph of Section 4.2. Before providing a concrete and concise algorithm and testing it, we will introduce a practical improvement based on a simple consideration.

Note that simulating the iid variables  $\{\Theta_n\}_{n \in \mathcal{Z}^0}$  is clearly quicker and easier than employing the full machinery of our algorithms. Recall that the dominating process was introduced only to detect coalescence for the chain  $\{X_n\}_{n \in \mathcal{Z}^0}$ . Thus, given  $\{\Theta_n\}_{n \in \mathcal{Z}_{\Delta(0)}^0}$  for some burn-in parameter  $\Delta(0) \in \mathcal{Z}^0$  and an upper bound  $X'_{\Delta(0)} = D_{\Delta(0)} \geq X_{\Delta(0)}$  (recall the definition of  $\{D_n\}$  in (3.8)), one could recursively construct  $X'_{n+1} = \psi(X'_n, \Theta_n)$  for  $n \in \mathcal{Z}_{\Delta(0)}^0$  and if any coalescence were detected, we would be certain that  $X'_0 = X_0$ . Our objective is hence to take an appropriate  $\Delta(0)$  that increases the probability  $\mathbb{P}(X'_0 = X_0)$ . Algorithm 9 is a complete and compact simulation algorithm of  $X_0$ , which makes use of this.

It is known that spectrally negative stable processes of infinite variation ( $\alpha > 1$  and  $\rho = 1/\alpha$ ) satisfy  $\bar{S}(\alpha, \rho) = S^+(\alpha, \rho)$  [25, Thm 1]. As a simple application and sanity-check, we now present a comparison between the empirical distribution function of  $N = 10^4$  samples against the actual distribution function in this case. To validate the samples, we compute the Kolmogorov-Smirnov statistic and test the hypothesis. (These graphs can be replicated following the guide available in [19].) In all three cases the null hypothesis of all samples coming from their respective distribution functions is not rejected (see Figure 5.1).

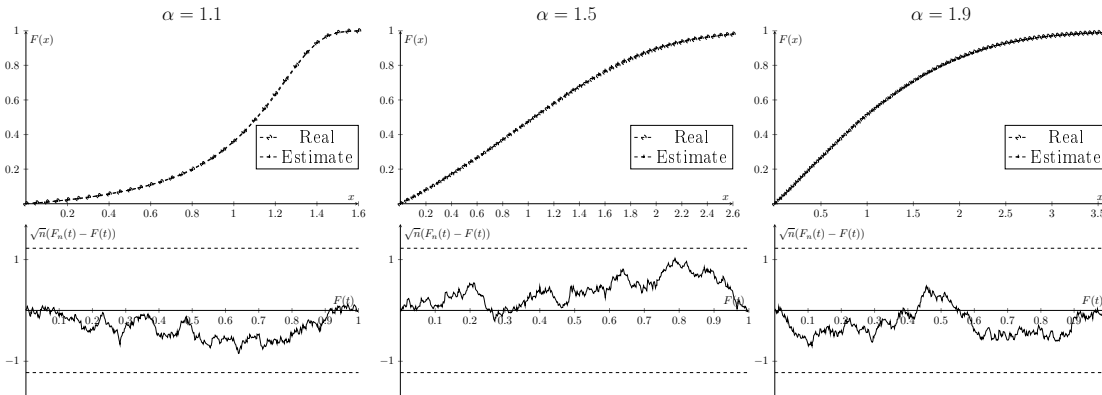


FIGURE 5.1: Empirical distribution functions for spectrally negative infinite variation stable process with parameters  $\rho = 1/\alpha$  and, from left to right,  $\alpha = 1.1$ ,  $\alpha = 1.5$  and  $\alpha = 1.9$ . The top graphs show the empirical distribution functions  $F_N$  for  $N = 10^4$  samples and compare it to the distribution function  $F = \bar{S}(\alpha, \rho) = S^+(\alpha, \rho)$  [25, Thm 1]. The bottom graphs show  $s \mapsto \sqrt{N}(F_N \circ F^{-1}(s) - s)$  on  $[0, 1]$  (or equivalently, the curve  $t \mapsto (F(t), \sqrt{N}(F_N(t) - F(t)))$  in  $\mathbb{R}^2$  for  $t > 0$ ), which converges weakly to a Brownian bridge. The dashed lines are the 0.05 and 0.95 quantiles of the Kolmogorov-Smirnov statistic, derived from the distribution of the signed maximum modulus of the Brownian bridge.

---

**Algorithm 9** Perfect simulation of  $X_0 \stackrel{d}{=} \bar{Y}_1$ 


---

**Require:** Parameters  $0 < \delta < d < \frac{1}{\alpha\rho}$ ,  $\kappa > \max\{\frac{\log(2)}{3\eta}, \frac{1}{\alpha\rho}\}$ ,  $\gamma > 0$  and  $\Delta(0) \in \mathcal{Z}^0$

- 1: Put  $x := \infty$ ,  $t := 1$ ,  $s := \Delta(0)$  and  $m := n := \Delta(0) + 1$   $\triangleright x$  is an upper bound on  $\{C_k\}_{k \in \mathcal{Z}^{\Delta(0)}}$
  - 2: Sample  $\{\Theta_k\}_{k \in \mathcal{Z}^0_{\Delta(0)}}$   $\triangleright$  Recall its definition in Section 4, paragraph 1
  - 3: **loop**
  - 4:   Sample  $(\chi_{m-1}, \{S_k\}_{k \in \mathcal{Z}^s_{\chi_{m-1}}})$   $\triangleright$  Algorithm 3
  - 5:   **while**  $n = m$  or  $\Delta(t) > \chi_{m-1}$  **do**
  - 6:     Sample  $C_{\Delta(t)}, \dots, C_{T_{-2\kappa}^{\Delta(t)}}$  conditional on  $\{R_{\Delta(t)} < x - C_{\Delta(t)}\}$   $\triangleright$  Algorithm 6
  - 7:     Put  $\Delta(t+1) := T_{-2\kappa}^{\Delta(t)}$  and  $t := t+1$   $\triangleright$  Recall its definition in (4.1)
  - 8:     Sample  $1_{\{R_{\Delta(t)} > \kappa\}}$  given  $\{R_{\Delta(t)} < x - C_{\Delta(t)}\}$   $\triangleright$  Algorithm 4
  - 9:     **if**  $1_{\{R_{\Delta(t)} > \kappa\}} = 0$  **then**
  - 10:       Compute  $\{R_k\}_{k \in \mathcal{Z}^n_{\Delta(t-1)}}$  and put  $n := \Delta(t-1)$  and  $x := C_{\Delta(t)} + \kappa$
  - 11:     **else**
  - 12:       Sample  $C_{\Delta(t)-1}, \dots, C_{T_{\kappa}^{\Delta(t)}}$  from  $\mathbb{P}^\eta$   $\triangleright$  Algorithm 5
  - 13:       Put  $\Delta(t+1) := T_{\kappa}^{\Delta(t)}$  and  $t := t+1$
  - 14:     **end if**
  - 15:   **end while**
  - 16:   Sample  $\{(U_k, \Lambda_k)\}_{k \in \mathcal{Z}^s_{\chi_{m-1}}}$  from  $\{F_k\}_{k \in \mathcal{Z}^s_{\chi_{m-1}}}$  and put  $s := \chi_{m-1}$   $\triangleright$  Algorithm 8
  - 17:   Compute  $D_{m-1}$  and put  $m := m-1$   $\triangleright$  Recall its definition in (3.8)
  - 18:   **if**  $D_m \leq a(\Theta_m)$  **then**
  - 19:     **return**  $X_0 := \psi(\dots \psi(D_m, \Theta_m), \dots, \Theta_{-1})$   $\triangleright$  In this case  $\sigma = m$
  - 20:   **else if**  $m = \Delta(0)$  **then**
  - 21:     Put  $X_0 := \psi(\dots \psi(D_m, \Theta_m), \dots, \Theta_{-1})$
  - 22:     **if** coalescence was detected **then**
  - 23:       **return**  $X_0$
  - 24:     **end if**
  - 25:   **end if**
  - 26: **end loop**
-

### 5.1. Parameter choice and numerical performance

As explicitly stated in Algorithm 9, and if one allows  $m^*$  (recall its definition in paragraph 1, p. 14) to vary over  $\lfloor \frac{1}{\delta\gamma} \log \mathbb{E}S_1^\gamma \rfloor^+ + \mathbb{N}$ , our simulation procedure has 6 different parameters. A full theoretical optimisation is infeasible as it heavily depends on, among other things, the way the algorithm is coded, the computational cost of simulating each variable, the cost of each calculation, memory accessing cost, the quality and state of the RAM and  $(\alpha, \rho)$ . However, for the sake of presenting its practical feasibility, we have implemented the algorithm in the Julia programming language (see [19]) and ran it on a macOS Mojave 10.14.3 (18D109) with a 4.2 GHz Intel®Core™i7 processor and a 8 GB 2400 MHz DDR4 memory. This implementation is far from optimal, but still outputs  $10^4$  samples in approximately 1.15 seconds (without multithreading) for the suggested parameters  $(d, \delta, \gamma, \kappa, \Delta(0), m^*) = \varpi$  where

$$\varpi = \varpi(\alpha, \rho) = \left( \frac{2}{3\alpha\rho}, \frac{1}{3\alpha\rho}, \frac{19}{20}\alpha, 4 + \max \left\{ \frac{\log(2)}{3\eta(\frac{2}{3\alpha\rho})}, \frac{1}{\alpha\rho} \right\}, 40, 12 + \left\lfloor \frac{60}{19} \rho \log \mathbb{E}S_1^{\frac{19}{20}\alpha} \right\rfloor^+ \right).$$

This performance varies slightly for different choices of  $(\alpha, \rho)$ . To put things in perspective, Algorithm 8 outputs, for the parameter choice  $\varpi$ ,  $10^6$  samples in approximately 0.4322 seconds and drawing  $10^6$  samples from  $S^+(\alpha, \rho)$  takes 0.1833 seconds. On the other hand, the first iteration of Algorithm 3 (which simulates the indicators  $\{I_k^0\}_{k \in \mathcal{Z}_{x_0}^0}$  and the conditionally positive stable random variables  $\{S_k\}_{k \in \mathcal{Z}_{x_0}^0}$ ) simulates  $10^4$  samples in about 0.8125 seconds and is, although fast, the most computationally costly component of Algorithm 9. The main sources of this cost are the calculation of the probabilities  $\{p(m)\}$  (see their definition in (3.4)) and the simulation of  $\{S_k\}_{k \in \mathcal{Z}_{x_0}^0}$  conditioned on the values of  $\{I_k^0\}_{k \in \mathcal{Z}_{x_0}^0}$ .

Next we show the local marginal behaviour of the number of samples outputted with confidence intervals, for a few different choices of parameters  $(\alpha, \rho)$ . We will see that, although  $\varpi$  may not be optimal, it is a simple and yet efficient choice. Moreover, the variation in performance for parameters close to this one is small, thus showing that this choice is relatively robust.

It should be noted that the data presented in Figure 5.2 is dependent on the characteristics of the hardware and software used. Hence, these exact numbers are not easily replicated. For instance, these times scale sub-linearly as a function of the batch size, and are not replicated despite using the garbage collector and the same random seed. It is readily seen that the exact value of the parameters  $d, \delta, \Delta(0)$  and  $m^*$  is not too important in so far as they remain at a reasonable distance from their boundaries (where  $\infty$  is a right-boundary for  $\Delta(0)$  and  $m^*$ ). The value of  $\kappa$  is slightly

more sensitive, as is  $\gamma$ . Other choices of  $(\alpha, \rho)$  have slightly different behaviours. The shapes of these curves are similar, but the apparent minima change. Thus, we argue that  $\varpi$  is a simple yet sensible choice.

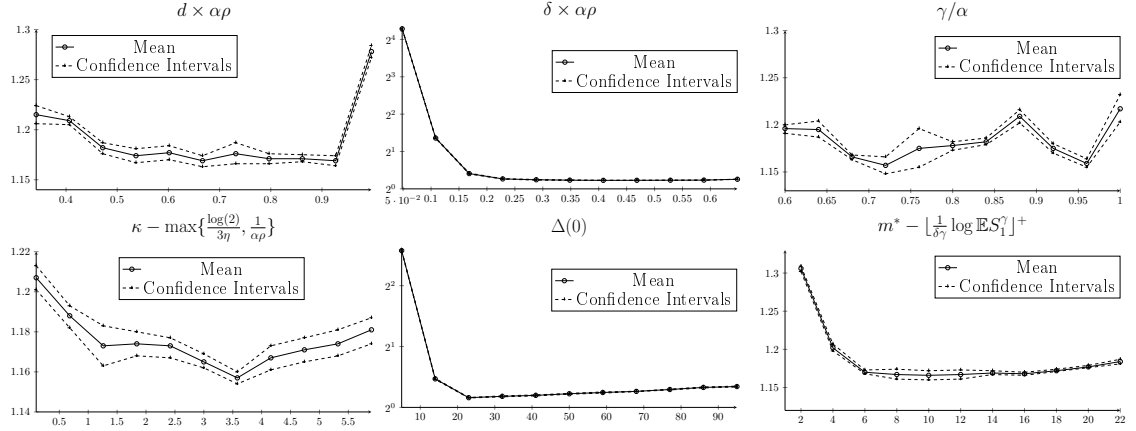


FIGURE 5.2: Time taken (in seconds) to simulate  $N = 10^4$  samples of  $\bar{S}(1.3, 1/2)$  with parameters moving about  $\varpi$ . In each plot, one parameter moves and all others are kept constant at the respective value of  $\varpi$ . We took 100 batches of samples, each with  $N = 10^4$  independent simulations, to construct asymptotic 95% confidence intervals based on the central limit theorem.

### Acknowledgements

JGC and AM are supported by The Alan Turing Institute under the EPSRC grant EP/N510129/1; AM supported by EPSRC grant EP/P003818/1 and the Turing Fellowship funded by the Programme on Data-Centric Engineering of Lloyd's Register Foundation; GUB supported by CoNaCyT grant FC-2016-1946 and UNAM-DGAPA-PAPIIT grant IN115217; JGC supported by CoNaCyT scholarship 2018-000009-01EXTF-00624. We thank Stephen Connor for the reference [9].

### Appendix A. Sampling the marginals of stable processes

A Lévy process  $Y = (Y_t)_{t \in [0, \infty)}$  in  $\mathbb{R}$  is *strictly stable* with index  $\alpha \in (0, 2]$  if for any constant  $c \geq 0$  the processes  $(Y_{ct})_{t \in [0, \infty)}$  and  $(c^{1/\alpha} Y_t)_{t \in [0, \infty)}$  have the same law. For brevity, we call  $Y$  a *stable process*. Sampling the increments of  $Y$  hence reduces to sampling  $Y_1$ . Using Zolotarev's (C)

form [30], up to a scaling constant the law of  $Y_1$  is parametrised by  $(\alpha, \beta) \in (0, 2] \times [-1, 1]$  via

$$\mathbb{E}e^{itY_1} = \exp\left(-|t|^\alpha e^{-i\frac{\pi\alpha}{2}\theta\text{sgn}(t)}\right), \quad \text{where } t \in \mathbb{R}, \theta = \beta\left(1_{\alpha \leq 1} + \frac{\alpha-2}{\alpha}1_{\alpha > 1}\right), \quad (\text{A.1})$$

and  $\text{sgn}(t)$  equals 1 (resp.  $-1$ ) if  $t \geq 0$  (resp.  $t < 0$ ). The Mellin transform of  $Y_1$  equals

$$\mathbb{E}Y_1^s 1_{Y_1 > 0} = \rho \frac{\Gamma(1+s)\Gamma(1-\frac{s}{\alpha})}{\Gamma(1+s\rho)\Gamma(1-s\rho)}, \quad (\text{A.2})$$

where  $\rho = \frac{1+\theta}{2}$  and  $\Gamma(\cdot)$  denotes the gamma function (see [30] Section 5.6). Taking  $s = 0$  in (A.2) implies that the stable law is uniquely determined by  $\alpha$  and its positivity parameter  $\rho = \mathbb{P}(Y_1 > 0)$ . If  $\alpha > 1$ , the pair  $(\alpha, \rho) \in (0, 2] \times [0, 1]$  must satisfy  $\rho \in [1 - \frac{1}{\alpha}, \frac{1}{\alpha}]$ , since  $\theta \in [1 - \frac{2}{\alpha}, \frac{2}{\alpha} - 1]$ .

Let  $S(\alpha, \rho)$  and  $S^+(\alpha, \rho)$  denote the laws of  $Y_1$  and  $Y_1$  conditioned on being positive, respectively. As  $\rho, \alpha\rho \in [0, 1]$  and the Mellin transform determines the law uniquely, (A.2) implies that  $(Z'/Z'')^\rho$  follows  $S^+(\alpha, \rho)$ , where  $Z' \sim S(\alpha\rho, 1)$  and  $Z'' \sim S(\rho, 1)$  are independent. Since  $P'B + P''(1-B)$  follows  $S(\alpha, \rho)$ , where  $P' \sim S^+(\alpha, \rho)$ ,  $P' \sim S^+(\alpha, 1-\rho)$  and  $B \sim \text{Ber}(\rho)$  are independent, we need only be able to simulate a positive stable random variable with law  $S(\alpha, 1)$  for any  $\alpha \in (0, 1]$ . If  $\alpha = 1$ , then by (A.1),  $Y_1$  is a constant equal to one. If  $\alpha \in (0, 1)$ , Kanter's factorisation states

$$\left(\sin(\alpha\pi U)^\alpha \sin((1-\alpha)\pi U)^{1-\alpha} / \sin(\pi U)\right)^{\frac{1}{\alpha}} E^{1-\frac{1}{\alpha}} \sim S(\alpha, 1),$$

where  $E$  is exponential with mean one, independent of  $U$ , which is uniform on  $(0, 1)$  (see [30, Sec 4.4]). For alternative ways of sampling from the laws  $S(\alpha, \rho)$  and  $S^+(\alpha, \rho)$  we refer to [14].

## References

- [1] BERNYK, V., DALANG, R. C. AND PESKIR, G. (2008). The law of the supremum of a stable Lévy process with no negative jumps. *Ann. Probab.* **36**, 1777–1789.
- [2] BERNYK, V., DALANG, R. C. AND PESKIR, G. (2011). Predicting the ultimate supremum of a stable Lévy process with no negative jumps. *Ann. Probab.* **39**, 2385–2423.
- [3] BERTOIN, J. (1996). *Lévy processes* vol. 121 of *Cambridge Tracts in Mathematics*. Cambridge University Press, Cambridge.
- [4] BLANCHET, J. H. AND SIGMAN, K. (2011). On exact sampling of stochastic perpetuities. *J. Appl. Probab.* **48A**, 165–182.

- [5] BURACZEWSKI, D., DAMEK, E. AND MIKOSCH, T. (2016). *Stochastic models with power-law tails*. Springer Series in Operations Research and Financial Engineering. Springer, [Cham].
- [6] CHI, Z. (2012). On exact sampling of nonnegative infinitely divisible random variables. *Adv. in Appl. Probab.* **44**, 842–873.
- [7] CHI, Z. (2012). On exact sampling of the first passage event of Lévy process with infinite Lévy measure and bounded variation. *Stochastic Processes and their Applications* **126**, 1124–1144.
- [8] CHI, Z. (2018). Law and exact sampling of the first passage of a spectrally positive strictly stable process. *arXiv:1801.06891v1*.
- [9] CLOUD, K. AND HUBER, M. (2017). Fast perfect simulation of Vervaat perpetuities. *J. Complexity* **42**, 19–30.
- [10] DASSIOS, A., LIM, J. AND QU, Y. (2019). Exact simulation of generalised Vervaat perpetuities. *Journal of Applied Probability*.
- [11] DEVROYE, L. (1986). *Non-Uniform Random Variate Generation*(originally published with Springer-Verlag.
- [12] DEVROYE, L. (2001). Simulating perpetuities. *Methodol. Comput. Appl. Probab.* **3**, 97–115.
- [13] DEVROYE, L. AND FAWZI, O. (2010). Simulating the Dickman distribution. *Statist. Probab. Lett.* **80**, 242–247.
- [14] DEVROYE, L. AND JAMES, L. (2014). On simulation and properties of the stable law. *Stat. Methods Appl.* **23**, 307–343.
- [15] DEVROYE, L. AND JAMES, L. F. (2011). The double cftp method. *ACM Trans. Model. Comput. Simul.* **21**, 10:1–10:20.
- [16] DONEY, R. A. (2008). A note on the supremum of a stable process. *Stochastics* **80**, 151–155.
- [17] ENSOR, K. B. AND GLYNN, P. W. (2000). Simulating the maximum of a random walk. *J. Statist. Plann. Inference* **85**, 127–135.
- [18] FILL, J. A. AND HUBER, M. L. (2010). Perfect simulation of Vervaat perpetuities. *Electron. J. Probab.* **15**, no. 4, 96–109.

- [19] GONZÁLEZ CÁZARES, J., MIJATOVIĆ, A. AND URIBE BRAVO, G. Code for the simulation of the stable supremum. <https://github.com/jorgeignaciogc/SupStable.jl> 2018. GitHub repository.
- [20] HUBER, M. L. (2016). *Perfect simulation* vol. 148 of *Monographs on Statistics and Applied Probability*. CRC Press, Boca Raton, FL.
- [21] KENDALL, W. S. AND MØLLER, J. (2000). Perfect simulation using dominating processes on ordered spaces, with application to locally stable point processes. *Adv. in Appl. Probab.* **32**, 844–865.
- [22] KUZNETSOV, A. (2011). On extrema of stable processes. *Ann. Probab.* **39**, 1027–1060.
- [23] KUZNETSOV, A. (2013). On the density of the supremum of a stable process. *Stochastic Process. Appl.* **123**, 986–1003.
- [24] MEYN, S. AND TWEEDIE, R. L. (2009). *Markov chains and stochastic stability* second ed. Cambridge University Press, Cambridge.
- [25] MICHNA, Z. (2013). Explicit formula for the supremum distribution of a spectrally negative stable process. *Electron. Commun. Probab.* **18**, no. 10, 6.
- [26] MURDOCH, D. J. AND GREEN, P. J. (1998). Exact sampling from a continuous state space. *Scand. J. Statist.* **25**, 483–502.
- [27] PITMAN, J. AND URIBE BRAVO, G. (2012). The convex minorant of a Lévy process. *Ann. Probab.* **40**, 1636–1674.
- [28] REVUZ, D. (1984). *Markov chains* second ed. vol. 11 of *North-Holland Mathematical Library*. North-Holland Publishing Co., Amsterdam.
- [29] SONG, R. AND VONDRAČEK, Z. (2008). On suprema of Lévy processes and application in risk theory. *Ann. Inst. Henri Poincaré Probab. Stat.* **44**, 977–986.
- [30] UCHAIKIN, V. V. AND ZOLOTAREV, V. M. (1999). *Chance and stability*. Modern Probability and Statistics. VSP, Utrecht.