

Simulating Sample Paths of Linear Fractional Stable Motion

Wei Biao Wu, George Michailidis, and Danlu Zhang

Abstract—An algorithm for generating sample paths of linear fractional stable motion (LFSM) is introduced. It is based on the approximation of LFSM by a linear process and exhibits low computational complexity. A detailed analysis of the error term involved in the approximation is provided, which in turn guides the user on selecting the size of the generated sequence.

Index Terms—Fast Fourier transform (FFT), heavy-tailed distributions, linear fractional stable motion (LFSM), linear process, long-range dependence, simulation.

I. INTRODUCTION

THE extreme complexity of modern communication and computer networks, coupled with their traffic characteristics—heavy tails, self-similarity, and long-range dependence—[27], [35] makes the characterization of their performance through analytical models an extremely difficult task. Under such circumstances, simulations become one of the most promising tools for understanding the behavior of such networks [26], [34]. One then must be able to generate traffic that exhibits the necessary temporal behavior over large time scales [24], [25].

One of the simplest models exhibiting long-range dependence is fractional Brownian motion (fBm) introduced by Kolmogorov [19] and further developed by Mandelbrot and Van Ness [22]. It is a Gaussian, nonstationary, self-similar process indexed by a parameter H . The self-similar nature of fBm has made it particularly attractive for using it as an input process when simulating queueing networks [28]. However, several traffic measurement studies do not show an agreement with the Gaussian marginal distribution assumption. There exists empirical evidence supporting a heavy tailed assumption [27] backed by theoretical work that explains how the former assumption induces through an appropriate mechanism long-range dependence in the aggregate traffic [20], [33], [23]. Therefore, researchers have focused their attention on a more general process that exhibits in a natural way both the necessary scaling behavior, coupled with heavy tails.

Manuscript received February 28, 2002; revised February 11, 2004. The material in this paper was presented in part at the 2002 Winter Simulation Conference, San Diego, CA, December 2002.

W. B. Wu is with the Department of Statistics, The University of Chicago, Chicago, IL 60637 USA (e-mail: wbwu@galton.uchicago.edu).

G. Michailidis is with the Department of Statistics, The University of Michigan, Ann Arbor, MI 48109 USA (e-mail: gmichail@umich.edu).

D. Zhang was with the Department of Electrical Engineering and Computer Science, The University of Michigan, Ann Arbor, MI 48109 USA. He is now with Qualcomm, Inc., San Diego, CA 92121 USA (e-mail: dzhang@qualcomm.com).

Communicated by G. Battail, Associate Editor At Large.

Digital Object Identifier 10.1109/TIT.2004.828059

Linear fractional stable motion (LFSM, also known as fractional Lévy motion), a process exhibiting such characteristics, is defined next. A random variable η is called *symmetric stable* with index $\alpha \in (0, 2]$ (S α S) if $\eta =_d \sigma \varepsilon$, where $\sigma \geq 0$ is the scale coefficient and ε is a random variable with characteristic function $\mathbf{E} \exp(i\lambda \varepsilon) = \exp(-|\lambda|^\alpha)$ [31] ($i = \sqrt{-1}$ represents the imaginary unit). Write $\|\eta\|_\alpha = \sigma$ for the scale parameter σ . Let $\{M_\alpha(s), s \in \mathcal{R}\}$ denote a Lévy α -stable motion; namely, $M_\alpha(\cdot)$ is a stochastic process that has stationary and independent S α S increments ($M_\alpha(s+h) - M_\alpha(s)$ is S α S) with scale parameter

$$\|M_\alpha(s+h) - M_\alpha(s)\|_\alpha = h^{1/\alpha}, \quad h \geq 0.$$

Let $M_\alpha(ds)$ be the corresponding S α S random measure with control measure ds . When $\alpha = 2$, the rescaled version $\{\sqrt{2}M_\alpha(s), s \in \mathcal{R}\}$ corresponds to standard Brownian motion. Let $-1/\alpha < \beta < 1 - 1/\alpha$, $x_+ = \max(0, x)$, and $x_- = \max(0, -x)$. For $b_1, b_2 \in \mathcal{R}$, define the LFSM

$$\xi_{H,\alpha}^{b_1,b_2} = \left\{ \xi_{H,\alpha}^{b_1,b_2}(t), t \in \mathcal{R} \right\}$$

via the stochastic integral representation

$$\xi_{H,\alpha}^{b_1,b_2}(t) = \int_{\mathcal{R}} \left\{ b_1[(t-s)_+^\beta - (-s)_+^\beta] + b_2[(t-s)_-^\beta - (-s)_-^\beta] \right\} M_\alpha(ds). \quad (1)$$

The resulting process $\xi_{H,\alpha}^{b_1,b_2}$ has stationary increments and is self-similar in the sense that for $c > 0$, $\{\xi_{H,\alpha}^{b_1,b_2}(ct), t \in \mathcal{R}\}$ has the same distribution as $\{c^H \xi_{H,\alpha}^{b_1,b_2}(t), t \in \mathcal{R}\}$, with the self-similarity parameter given by $H := 1/\alpha + \beta$. The LFSM can be thought of as the generalization of fBm [31] and is characterized by two parameters: the Hurst parameter H that measures the degree of the long-range dependence of the process and the Lévy parameter α that measures the heaviness of the tails of the marginal distributions. When $\alpha = 2$ (i.e., the marginal distributions are Gaussian) we recover fBm. Over the last few years, several studies of queueing performance under a self-similar stable motion input have appeared [13]–[15], [17].

The purpose of this paper is to introduce a new efficient algorithm to simulate sample paths of LFSM. The proposed method is based on approximating the LFSM through a linear process. The literature over the last decade has focused on methods for simulating sample paths for fBm (the special case for $\alpha = 2$). For example, there are methods that are based on the properties of fBm, such as through its stochastic representation [22], or the fractional integration of Gaussian white noise [1], or through

matching its covariance function; other methods attempt to first synthesize the increments process and then generate a realization of fBm by calculating the cumulative sums process of fractional Gaussian noise, such as those of Levinson [7] and the fast and exact method of Wood and Chan [11], [36]. These methods take advantage of the fact that the increment process is a stationary one and its covariance matrix is a Toeplitz matrix. Finally, there are methods that rely on approximations of fBm; for example, the method of Flandrin [12] involves computing the wavelet coefficients of the wavelet transform of fBm and then synthesize fBm through the inverse wavelet transformation, while the random midpoint displacement method [21] progressively subdivides the interval over which a sample path is generated and at each subinterval, a Gaussian displacement is used to determine the value of the process at the midpoint. Other approximate methods use queueing models and renewal processes to generate fBm as the limiting process [35]. Unlike fBm, there have been hardly any proposals for efficiently generating sample paths of LFSM.

One major potential difficulty is that covariances can no longer be defined for LFSM since the variances are infinite for $\alpha < 2$. Thus, those simulation techniques for fBm which are based on matching covariances cannot be directly applied to simulate sample paths of LFSM. Nevertheless, we can resort to other approaches such as stochastic integral representations. In their book, Samorodnitsky and Taqqu [31] provide an algorithm (further developed in Stoev and Taqqu (2003)) that approximates sample paths of LFSM based on the discretization of the stochastic integral representation given in (1). The Stoev-Taqqu version of the algorithm is compared to the proposed one in Section III.

Our approach allows one to generate an LFSM efficiently, since it has linear memory requirements and a competitive time complexity (for details see Section II) similar to the efficient algorithm proposed in [36]. The paper is organized as follows: in Section II, the theoretical development and the proposed algorithm are given, while the corresponding proofs are presented in Section IV. Finally, some concluding remarks are drawn in Section III-B.

II. THE ALGORITHM AND A LIMIT THEOREM

A. Preliminaries

Let $b_1 = 1$ and $b_2 = 0$ in (1) and write $\xi_{H,\alpha} = \xi_{H,\alpha}^{1,0}$. The general case with $b_1, b_2 \in \mathcal{R}$ is discussed in Section II-E. Let $\{\varepsilon_i\}_{i \in \mathcal{Z}}$ be a sequence of independent and identically distributed (i.i.d.) standard (namely, the scale $\|\varepsilon_i\|_\alpha = 1$) S α S random variables, with $\alpha \in (0, 2]$. Define the sequence $\{a_n\}_{n \in \mathcal{Z}_+}$ with $a_1 = 1$ and $a_n = n^\beta - (n-1)^\beta$ for $n \geq 2$, and the partial sum $s_n = \sum_{i=1}^n a_i = n^\beta$ for all $n \geq 1$.

Define the one-sided linear process (or MA(∞) process [3])

$$X_n = a_1 \varepsilon_{n-1} + a_2 \varepsilon_{n-2} + a_3 \varepsilon_{n-3} + \cdots = \sum_{i=1}^{\infty} a_i \varepsilon_{n-i}. \quad (2)$$

Under the condition $\sum_{i=1}^{\infty} |a_i|^\alpha < \infty$, which is equivalent to $H = 1/\alpha + \beta < 1$, the process exists almost surely [3] by Kol-

mogorov's Three Series Theorem. Moreover, by the definition of S α S distributions, X_n has the same distribution as

$$\left(\sum_{i=1}^{\infty} |a_i|^\alpha \right)^{1/\alpha} \varepsilon_0$$

and, in addition, we have that

$$\|X_n\|_\alpha = \left(\sum_{i=1}^{\infty} |a_i|^\alpha \right)^{1/\alpha}.$$

Let

$$S_n(t) = \sum_{j=1}^{\lceil nt \rceil} X_j \sum_{i=0}^{\infty} X_i \left(\sum_{j=1-i}^{\lceil nt \rceil - i} a_j \right), \quad 0 < t \leq 1$$

be the partial sum process of X_n and let $S_n = S_n(1)$.

Under an appropriate set of conditions on the coefficients a_i , a properly normalized version of $S_n(t)$ converges in the sense of finite-dimensional distributions to the LFSM $\xi_{H,\alpha}(t)$ [3]–[5]. In the case where the innovations $\{\varepsilon_i\}$ have moments higher than 2, Davydov [10] considers the weak convergence problem and shows that the limiting process corresponds to fBm. Using the algorithm described in Section II-F with $m = 2\,000\,000$ and $n = 10\,000$, sample paths of LFSM for different values of the Hurst and Lévy parameters (H and α) are shown in Fig. 1. It is known that for smaller H , the sample paths are discontinuous (cf. [31, Ch. 10]) due to heavy tails. The heaviness of the tails of the marginal distribution induces very large bursts, especially for $\alpha < 1$.

In what follows we will make use of (2) and give an efficient algorithm for simulating the sample paths of LFSM. However, in principle, the linear process has an infinite number of terms. Therefore, in practice, we are forced to use a finite number of terms, which leads to using a truncated version of the linear process. Moreover, in order to speed up calculations, we embed the coefficients a_i in a circulant matrix (an idea also used in [11]) as shown later. The main issue then becomes to decide on the number of terms to be used in order to achieve a satisfactory approximation and to provide an estimate of the error term.

B. Asymptotic Results

Recall that $M_\alpha(ds)$ denotes an S α S random measure with control measure ds . Then, for each $n \geq 1$ we have that

$$\varepsilon_{k,n} = n^{1/\alpha} [M_\alpha(k/n) - M_\alpha((k-1)/n)], \quad k \in \mathcal{Z} \quad (3)$$

are i.i.d. standard S α S random variables. Define next the following variables:

$$X_{k,n} = \sum_{i=1}^{\infty} a_i \varepsilon_{k-i,n} \quad (4)$$

and

$$Y_{k,n} = n^H \int_{\mathcal{R}} [(k/n - s)_+^\beta - ((k-1)/n - s)_+^\beta] M_\alpha(ds). \quad (5)$$

Then, for each n , the process $(X_{k,n})_{k \in \mathcal{Z}}$ has the same distribution as $(X_k)_{k \in \mathcal{Z}}$.

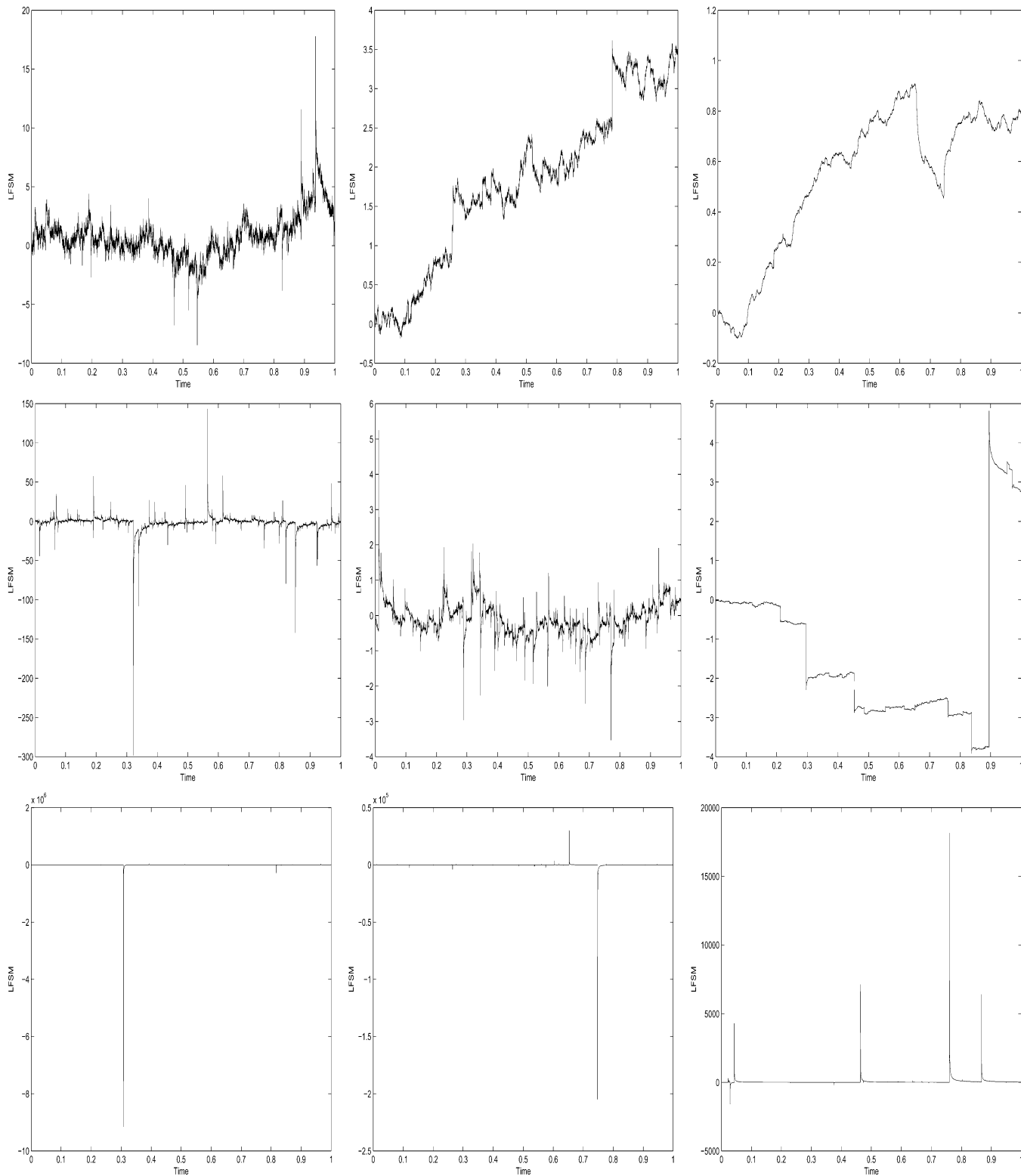


Fig. 1. Top, middle, and bottom panels: realizations of LFSMs for $\alpha = 1.8$, $\alpha = 1.2$, and $\alpha = 0.6$. In all cases, the left panel corresponds to $H = 0.2$, the middle panel to $H = 0.5$, and the right panel to $H = 0.8$. The x -axis represents time ($t = k/n, k = 0, 1, 2, \dots, n$), while on the y -axis the values of the LFSM process are given.

Theorem 1: For $X_{k,n}$ and $Y_{k,n}$ defined in (4) and (5), respectively, we have that

$$\max_{1 \leq K \leq n} \left\| \sum_{k=1}^K (X_{k,n} - Y_{k,n}) \right\|_{\alpha} = \mathcal{O}(1). \quad (6)$$

Notice that $\{\sum_{k=1}^K Y_{k,n}/n^H, 0 \leq K \leq n\}$ has the same distribution as $\{\xi_{H,\alpha}(K/n), 0 \leq K \leq n\}$. The main thrust of Theorem 1 is that, by defining $\varepsilon_{k,n}$ in (3), the partial sum of the linear process $X_{k,n}$ can be used to approximate $\{\sum_{k=1}^K Y_{k,n}/n^H, 0 \leq K \leq n\}$ on the *same* probability space. Therefore, by generating

sample paths of linear processes and then computing their partial sums, we are able to synthesize sample paths of LFSM. The proof of the theorem is given in Section IV.

C. Approximations

Since there are infinitely many terms in the representation given in (2), for simulation purposes based on such linear processes, it is necessary to adopt an appropriate truncation scheme. We assume throughout that the sequence $m = m_n$ satisfies $m/n \rightarrow \infty$ as $n \rightarrow \infty$. The first step is to approximate the tail processes

$$\psi_{k,m,n} = \sum_{j=m+1-n}^{\infty} a_{k+j}\varepsilon_{-j}, \quad 1 \leq k \leq n$$

by $\psi_{1,m,n}$ (cf Lemma 1), which can be easily generated and does not depend on k . This observation is crucial since it is difficult to quickly and accurately generate the tail processes. Next, the ‘‘main part’’ of the linear process $\sum_{j=1-k}^{m-n} a_{k+j}\varepsilon_{-j}$ is approximated by a process $W_{k,m,n}$ with a circulant representation (cf (14)) whose computation can be carried out in a sufficiently fast manner (cf Section II -D).

Lemma 1: Assume $m, n \rightarrow \infty$, and $m/n \rightarrow \infty$. Then, for the truncated process

$$\psi_{k,m,n} = \sum_{j=m+1-n}^{\infty} a_{k+j}\varepsilon_{-j} \quad (7)$$

we have that

$$\max_{1 \leq k \leq n} \left\| \sum_{k=1}^K \psi_{k,m,n} - K\psi_{1,m,n} \right\|_{\alpha} = \mathcal{O}(n^2 m^{\beta-2+1/\alpha}). \quad (8)$$

Proof: Observe that

$$\begin{aligned} & \left\| \sum_{k=1}^K \psi_{k,m,n} - K\psi_{1,m,n} \right\|_{\alpha} \\ &= \sum_{j=m+1-n}^{\infty} |(K+j)^{\beta} - j^{\beta} - K[(1+j)^{\beta} - j^{\beta}]|^{\alpha}. \quad (9) \end{aligned}$$

Note that $n/j \rightarrow 0$ since $n/m \rightarrow 0$. By Taylor’s expansion, we have $(1+\delta)^{\beta} = 1 + \delta\beta + \mathcal{O}(\delta^2)$ as $\delta \rightarrow 0$, and, therefore,

$$|(K+j)^{\beta} - j^{\beta} - K[(1+j)^{\beta} - j^{\beta}]|/j^{\beta} = \mathcal{O}(n^2/j^2)$$

Hence, we get that

$$\begin{aligned} & \sum_{j=m+1-n}^{\infty} |(K+j)^{\beta} - j^{\beta} - K[(1+j)^{\beta} - j^{\beta}]|^{\alpha} \\ &= \sum_{j=m+1-n}^{\infty} \mathcal{O}(n^2 j^{\beta-2})^{\alpha} \\ &= \mathcal{O}[n^{2\alpha} m^{1+(\beta-2)\alpha}] \end{aligned}$$

which establishes the lemma. \square

Lemma 2: Define the following matrix (see (10) at the bottom of the page). Then

$$\max_{1 \leq k \leq n} \left\| \sum_{k=1}^K U_{k,m,n} \right\|_{\alpha} = \mathcal{O}(n^{1/\alpha+1} m^{\beta-1}). \quad (11)$$

Proof: Since $\sum_{j=N}^m a_j = m^{\beta} - (N-1)^{\beta}$ for $m \geq N \geq 2$, (11) follows from

$$\begin{aligned} & \left\| \sum_{k=n-l}^{n-1} U_{k,m,n} \right\|_{\alpha}^{\alpha} = \sum_{j=1}^l |m^{\beta} - (m-j)^{\beta}|^{\alpha} \\ &= \mathcal{O}(n(nm^{\beta-1})^{\alpha}) \end{aligned}$$

in view of $|m^{\beta} - (m-l)^{\beta}| = \mathcal{O}(lm^{\beta-1})$, $0 \leq l \leq n$, and $n/m \rightarrow 0$. \square

Lemma 3: Let $\{\varepsilon_n, n \in \mathcal{Z}\}$ be a sequence of i.i.d. standard S α S random variables; further, let $\{c_n, n \in \mathcal{Z}\}$ and $\{d_n, n \in \mathcal{Z}\}$ be two sequences of real numbers such that

$$\sum_{n \in \mathcal{Z}} (|c_n|^{\alpha} + |d_n|^{\alpha}) < \infty.$$

Finally, define $\eta_1 = \sum_{n \in \mathcal{Z}} c_n \varepsilon_{-n}$ and $\eta_2 = \sum_{n \in \mathcal{Z}} d_n \varepsilon_{-n}$. We then have that

$$\|\eta_1 + \eta_2\|_{\alpha} \leq \max(2, 2^{1/\alpha})(\|\eta_1\|_{\alpha} + \|\eta_2\|_{\alpha}). \quad (12)$$

Proof: Using $|x+y|^{\alpha} \leq c_{\alpha}(|x|^{\alpha} + |y|^{\alpha})$, where $c_{\alpha} = \max(2^{\alpha-1}, 1)$, we have that

$$\begin{aligned} \|\eta_1 + \eta_2\|_{\alpha}^{\alpha} &= \sum_{n \in \mathcal{Z}} |c_n + d_n|^{\alpha} \\ &\leq c_{\alpha} \sum_{n \in \mathcal{Z}} (|c_n|^{\alpha} + |d_n|^{\alpha}) \\ &= c_{\alpha} (\|\eta_1\|_{\alpha}^{\alpha} + \|\eta_2\|_{\alpha}^{\alpha}) \quad (13) \end{aligned}$$

which yields (12) since

$$\|\eta_1\|_{\alpha}^{\alpha} + \|\eta_2\|_{\alpha}^{\alpha} \leq 2(\|\eta_1\|_{\alpha} + \|\eta_2\|_{\alpha})^{\alpha}. \quad \square$$

Theorem 2: See the matrix (14) at the top of the following page. Then

$$\begin{aligned} & \max_{1 \leq k \leq n} \left\| \sum_{k=1}^K (W_{k,m,n} + \psi_{1,m,n} - X_k) \right\|_{\alpha} \\ &= \mathcal{O}(n^{1/\alpha+1} m^{\beta-1} + n^2 m^{\beta-2+1/\alpha}). \quad (15) \end{aligned}$$

Proof: Observe that

$$X_k - \psi_{k,m,n} = \sum_{j=1-k}^{m-n} a_{k+j}\varepsilon_{-j} = W_{k,m,n} - U_{k,m,n}.$$

Hence (15) follows from Lemmas 1–3. \square

Remark 1: The inclusion of the term $\psi_{1,m,n}$ in (8) of Lemma 1 significantly improves the accuracy of the approximation. The

$$\begin{pmatrix} U_{1,m,n} \\ U_{2,m,n} \\ \dots \\ U_{n-1,m,n} \\ U_{n,m,n} \end{pmatrix} = \begin{pmatrix} a_{m-n+2} & a_{m-n+3} & \dots & a_{m-1} & a_m & 0 \\ a_{m-n+3} & a_{m-n+4} & \dots & a_m & 0 & 0 \\ \dots & \dots & \dots & 0 & 0 & 0 \\ a_m & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & \dots & 0 & 0 \end{pmatrix} \begin{pmatrix} \varepsilon_{n-1} \\ \varepsilon_{n-2} \\ \dots \\ \varepsilon_1 \\ \varepsilon_0 \end{pmatrix}. \quad (10)$$

$$\begin{pmatrix} W_{1,m,n} \\ W_{2,m,n} \\ \dots \\ W_{n-1,m,n} \\ W_{n,m,n} \end{pmatrix} = \begin{pmatrix} a_{m-n+2} & a_{m-n+3} & \dots & a_m & a_1 & a_2 & \dots & a_{m-n+1} \\ a_{m-n+3} & a_{m-n+4} & \dots & a_1 & a_2 & a_3 & \dots & a_{m-n+2} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ a_m & a_1 & \dots & a_{n-1} & a_n & a_{n+1} & \dots & a_{m-1} \\ a_1 & \dots & a_{n-1} & a_n & a_{n+1} & \dots & a_{m-1} & a_m \end{pmatrix} \begin{pmatrix} \varepsilon_{n-1} \\ \varepsilon_{n-2} \\ \dots \\ \varepsilon_{n-m+1} \\ \varepsilon_{n-m} \end{pmatrix}. \quad (14)$$

intuition is clear since as $n/m \rightarrow 0$, $a_{k+j}/a_{1+j} \rightarrow 1$ uniformly over $k = 1, \dots, n$ as $j \geq m+1-n \rightarrow \infty$ and, hence, $\psi_{k,m,n}$ is close to $\psi_{1,m,n}$. Without subtracting the term $\psi_{1,m,n}$, the error bound would be

$$\max_{1 \leq K \leq n} \left\| \sum_{k=1}^K \psi_{k,m,n} \right\|_{\alpha} \geq \left\| \sum_{k=1}^n \psi_{k,m,n} \right\|_{\alpha}.$$

The latter has the same order of magnitude as

$$n \|\psi_{1,m,n}\|_{\alpha} = n \left(\sum_{j=m+1-n}^{\infty} |a_{1+j}|^{\alpha} \right)^{1/\alpha} \geq cnm^{\beta-1+1/\alpha}$$

for some constant $c > 0$, in view of

$$\begin{aligned} n \|\psi_{1,m,n}\|_{\alpha} &\leq 2^{1/\alpha} \left(\left\| \sum_{k=1}^n \psi_{k,m,n} - n\psi_{1,m,n} \right\|_{\alpha} + \left\| \sum_{k=1}^n \psi_{k,m,n} \right\|_{\alpha} \right) \quad (16) \end{aligned}$$

due to (12). Clearly, $n^2 m^{\beta-2+1/\alpha} = o(nm^{\beta-1+1/\alpha})$ since $n/m \rightarrow 0$. Thus, the error bound is substantially improved by subtracting $\psi_{1,m,n}$ in (8). \square

We now apply Theorems 1 and 2 to obtain an error bound. Let $\varepsilon_k = \varepsilon_{k,n}$ be defined as in (3), namely, we embed ε_k into the Lévy α -stable motion $M_{\alpha}(\cdot)$.

Proposition 1: Let $S_n^*(t)$, $0 \leq t \leq 1$ be a stepwise constant function such that

$$S_n^*(t) = \sum_{i=1}^k W_{i,m,n} + k\psi_{1,m,n}$$

when $k/n \leq t < (k+1)/n$, $0 \leq k \leq n$. Then

$$\begin{aligned} \sup_{0 \leq u \leq 1} \|n^{-H} S_n^*(u) - \xi_{H,\alpha}(u)\|_{\alpha} &= \mathcal{O} \left[n^{-H} + (n/m)^{\min(2-H, 1-H+1/\alpha)} \right]. \quad (17) \end{aligned}$$

Proof: Observe that

$$\xi_{H,\alpha}(K/n) = n^{-H} \sum_{k=1}^K Y_{k,n}.$$

Define

$$\eta_1 = \sum_{k=1}^K (W_{k,m,n} + \psi_{1,m,n} - X_{k,n})$$

and

$$\eta_2 = \sum_{k=1}^K (X_{k,n} - Y_{k,n}).$$

Then by using relationship (12), and Theorems 1 and 2 we get

$$\begin{aligned} \max_{1 \leq K \leq n} \|n^{-H} S_n^*(K/n) - \xi_{H,\alpha}(K/n)\|_{\alpha} &= \mathcal{O} \left[n^{-H} + (n/m)^{\min(2-H, 1-H+1/\alpha)} \right] \quad (18) \end{aligned}$$

since $H = \beta + 1/\alpha$ and $n/m \rightarrow 0$. By the stationarity of the increments of the LFSM processes, we have that

$$\begin{aligned} \{\xi_{H,\alpha}(u) - \xi_{H,\alpha}(\lfloor nu \rfloor/n), k/n \leq u \leq (k+1)/n\} &=_d \{\xi_{H,\alpha}(v), 0 \leq v \leq 1/n\}, \text{ for all } k = 0, 1, \dots, n-1. \end{aligned}$$

Therefore,

$$\begin{aligned} \sup_{0 \leq u \leq 1} \|\xi_{H,\alpha}(u) - \xi_{H,\alpha}(\lfloor nu \rfloor/n)\|_{\alpha}^{\alpha} &= \sup_{0 \leq v \leq 1/n} \|\xi_{H,\alpha}(v)\|_{\alpha}^{\alpha} \\ &= n^{-H\alpha} \|\xi_{H,\alpha}(1)\|_{\alpha}^{\alpha} \\ &= \mathcal{O}(n^{H\alpha}) \end{aligned}$$

where in the second equality above, we used the self-similarity of the process $\xi_{H,\alpha}$. \square

Remark 2: In Lemma 2, we introduce additional dependence artifacts to ensure the circular representation (14). In our FFT-based algorithm (cf Section II-D), such a circular structure is needed for a fast implementation. The additional dependence yields the error term $(n/m)^{1-H+1/\alpha}$ in (17). Notice that if $0 < \alpha \leq 1$, the term $(n/m)^{1-H+1/\alpha}$ is dominated by $(n/m)^{2-H}$ since $1 - H + 1/\alpha \geq 2 - H$. On the other hand, if $1 < \alpha \leq 2$, then the overall error rate is $\mathcal{O}[n^{-H} + (n/m)^{1-H+1/\alpha}]$. Finally, if m is sufficiently large such that $n^{(1+1/\alpha)/(1-H+1/\alpha)} = o(m)$ then $(n/m)^{1-H+1/\alpha}$ is dominated by n^{-H} . In summary, the overall order of approximation is negatively affected by the introduced dependence if $1 < \alpha \leq 2$ and $m = o[n^{(1+1/\alpha)/(1-H+1/\alpha)}]$. \square

Samorodnitsky and Taquq [31, Proposition 3.5.1] showed that the stochastic integrals $\int_{\mathcal{R}} f_n(s) M_{\alpha}(ds)$ converge in probability to $\int_{\mathcal{R}} f(s) M_{\alpha}(ds)$ if and only if

$$\left\| \int_{\mathcal{R}} [f_n(s) - f(s)] M_{\alpha}(ds) \right\|_{\alpha}^{\alpha} = \int_{\mathcal{R}} |f_n(s) - f(s)|^{\alpha} ds \rightarrow 0$$

where f_n and f are real-valued measurable functions. Hence, by Proposition 1, we get that $n^{-H} S_n^*(\cdot)$ converges in the sense of finite-dimensional distributions to the LFSM $\xi_{H,\alpha}(\cdot)$, and the bound (17) can be interpreted as the rate of convergence.

The error bound provides a theoretical justification on how to select the parameters m and n . Given an error level $\delta > 0$ and parameters α and H , a simple way (up to a multiplicative constant) to choose m and n is given by

$$n^{-H} + (n/m)^{\min(2-H, 1-H+1/\alpha)} \leq \delta. \quad (19)$$

Let $p = p(\alpha, H) = \min(2 - H, 1 - H + 1/\alpha)$ and $L = m/n$ denote the oversampling rate. The expression in (19) can be used

in the following two ways to determine the sample size n and the embedding dimension m . If one specifies first a sample size n that satisfies $n > \delta^{-1/H}$, then m is naturally given by $m = Ln$ with $L = (\delta - n^{-H})^{-1/p}$. If, on the other hand, one specifies first the oversampling rate L that satisfies $L > \delta^{-1/p}$, then n is given by $n = (\delta - L^{-p})^{-1/H}$ and, as before, $m = Ln$. Observe that $L \rightarrow \infty$ as $n \rightarrow \delta^{-1/H}$, whereas $n = (\delta - L^{-p})^{-1/H} \rightarrow \infty$ as $L \rightarrow \delta^{-1/p}$. Thus, these expressions capture the tradeoff between L and n that is necessary for making the procedure computationally tractable. We elaborate next on this tradeoff.

Since $0 < H < 1$ and $p > 0$, it is easily seen that the function $f(u) = u^{-H} + (u/m)^p$ reaches its minimum at

$$u_0 = (H/p)^{1/(H+p)} m^{p/(H+p)}$$

which solves $\frac{df}{du}|_{u=u_0} = 0$. Let $f(u_0) = \delta$. Then we obtain

$$m(\delta; \alpha, H) = \{\delta^{-1}[(H/p)^{-H/(H+p)} + (H/p)^{p/(H+p)}]\}^{1/H+1/p}. \quad (20)$$

Therefore, m and n can be chosen to be $\lceil m(\delta; \alpha, H) \rceil$ and $\lceil [(H/p)^{1/(H+p)} m^{p/(H+p)}] \rceil$, respectively, where $\lceil x \rceil$ is the smallest integer no less than x . For example, for $H = 0.8$, $\alpha = 1$, and $\delta = 0.005$, we have $m = 252779$ and $n = 1425$. The quantity $m(\delta; \alpha, H)$ provides a lower bound for the embedding dimension m . We believe that $m(\delta; \alpha, H)$ is a good choice in the sense that it reaches the minimal memory complexity (cf Section II-D).

D. The Algorithm

Observe that $\{W_{k,m,n}, 1 \leq k \leq n\}$ and $\psi_{1,m,n}$ defined in (14) and (7), respectively, are independent. Thus, by simulating $\{W_{k,m,n}, 1 \leq k \leq n\}$ and $\psi_{1,m,n}$ independently, the partial sum process $S_n^*(\cdot)$ defined in Proposition 1 can be obtained by combining the above two quantities. The former can be easily computed by the fast Fourier transforms (FFT), while the latter is identically distributed to $(\sum_{j=m+1-n}^{\infty} |a_{1+j}|^\alpha)^{1/\alpha} \varepsilon_{n-m-1}$. Elementary manipulations show that

$$\begin{aligned} & \left(\sum_{j=m+1-n}^{\infty} |a_{1+j}|^\alpha \right)^{1/\alpha} \\ &= \frac{|\beta|}{[\alpha(1-H)]^{1/\alpha}} (m-n)^{H-1} [1 + \mathcal{O}(m^{-1})] \end{aligned} \quad (21)$$

since $a_j = \beta j^{\beta-1} (1 + \mathcal{O}(j^{-1}))$ as $j \rightarrow \infty$. For generating the SoS variates one can use the algorithm proposed in Chambers *et al.* [8], [9].

For $m \geq 1$ let $\mathbf{a}_m = (a_1, \dots, a_m)$ and define its $m \times m$ circulant matrix by

$$A_m = \begin{pmatrix} a_1 & a_2 & \dots & a_{m-1} & a_m \\ a_2 & a_3 & \dots & a_m & a_1 \\ \dots & \dots & \dots & \dots & \dots \\ a_m & a_1 & \dots & a_{m-2} & a_{m-1} \end{pmatrix}. \quad (22)$$

Let the random vector $\mathbf{e}_m = (e_1, e_2, \dots, e_m)$ be such that $e_j = \varepsilon_{1-j}$ for $1 \leq j \leq m-n+1$ and $e_j = \varepsilon_{1-j+m}$ for $m-n+1 < j \leq m$. Then

$$\begin{pmatrix} W_{1,m,n} \\ W_{2,m,n} \\ \dots \\ W_{m-1,m,n} \\ W_{m,m,n} \end{pmatrix} = A_m \mathbf{e}_m^T \quad (23)$$

where the notation \mathbf{T} stands for transpose. Let $\omega_j = 2\pi j/m$. In Matlab, the discrete Fourier transform (DFT) of the vector $\mathbf{u}_m = (u_1, \dots, u_m)$ is given by $\mathbf{v}_m = (v_1, \dots, v_m) = \text{DFT}(\mathbf{u}_m)$, with

$$v_j = \sum_{t=1}^m u_t \exp[-i(t-1)\omega_{j-1}], \quad 1 \leq j \leq m. \quad (24)$$

The inverse discrete Fourier transform (IDFT) $\mathbf{u}_m = \text{IDFT}(\mathbf{v}_m)$ satisfies

$$u_j = \frac{1}{m} \sum_{t=1}^m v_t \exp[i(t-1)\omega_{j-1}], \quad 1 \leq j \leq m. \quad (25)$$

Using the celebrated FFT algorithm, one can obtain $\text{DFT}(\mathbf{u}_m)$ and $\text{IDFT}(\mathbf{v}_m)$ with time complexity $\mathcal{O}(m \log m)$ steps and $\mathcal{O}(m)$ memory complexity. Let

$$\mathbf{b}_m = \text{DFT}(\mathbf{a}_m) = (b_1, \dots, b_m)$$

$$\mathbf{f}_m = \text{DFT}(e_1, e_m, e_{m-1}, \dots, e_2) = (f_1, \dots, f_m)$$

and

$$\mathbf{p}_m = (b_1 f_1, \dots, b_m f_m).$$

Then, due to the fact that the $m \times m$ matrix A is circulant [11], we have

$$A_m \mathbf{e}_m^T = \text{IDFT}(\mathbf{p}_m). \quad (26)$$

Actually, by (25), the ℓ th ($1 \leq \ell \leq m$) element in $\text{IDFT}(\mathbf{p}_m)$ is

$$\begin{aligned} & \frac{1}{m} \sum_{k=1}^m b_k f_k \exp[i(k-1)\omega_{\ell-1}] \\ &= \frac{1}{m} \sum_{k=1}^m \sum_{t,t'=1}^m a_t e^{[(m+1-t') \bmod m]} \\ & \quad + 1 \exp[-i(t-1)\omega_{k-1} - i(t'-1)\omega_{k-1} + i(k-1)\omega_{\ell-1}] \end{aligned}$$

where $q \bmod m \in \{0, 1, \dots, m-1\}$ is the remainder of q when divided by m . Notice that $(k-1)\omega_{\ell-1} = (\ell-1)\omega_{k-1}$ and

$$\sum_{k=1}^m \exp[-i(t-1)\omega_{k-1} - i(t'-1)\omega_{k-1} + i(\ell-1)\omega_{k-1}]$$

is nonzero if and only if $\{-(t-1) - (t'-1) + (\ell-1)\} \bmod m = 0$, in which case the sum is m . Hence, the ℓ th element is equal to $\sum_{t,t'}^* a_t e^{[(m+1-t') \bmod m] + 1}$ which is exactly $W_{\ell,m,n}$, where the summation is taken over all pairs $1 \leq t \leq m$ and $1 \leq t' \leq m$ such that $(-t - t' + \ell + 1) \bmod m = 0$. We refer to [11] for more details. We next summarize the proposed algorithm.

Algorithm 1:

- Step 1: Use the FFT to compute the DFT of \mathbf{a}_m . Let $\mathbf{b}_m = \text{DFT}(\mathbf{a}_m) = (b_1, \dots, b_m)$.

- Step 2: Generate an m -dimensional random vector \mathbf{e}_m whose elements are i.i.d. S α S random variables employing the algorithm of Chambers *et al.* [8].
- Step 3: Use the FFT to compute the DFT of \mathbf{e}_m . Let $\mathbf{f}_m = \text{DFT}(\mathbf{e}_m) = (f_1, \dots, f_m)$.
- Step 4: Let $\mathbf{p}_m = (b_1 f_1, \dots, b_m f_m)$.
- Step 5: Use the FFT to compute the IDFT of \mathbf{p}_m . Let $(W_1, \dots, W_m) = \text{IDFT}(\mathbf{p}_m)$.
- Step 6: Compute the cumulative sum of the $W_j + c^* \varepsilon_1^*$, $1 \leq j \leq n$, and rescale it by n^H , where

$$c^* = |\beta|[\alpha(1-H)]^{-1/\alpha}(m-n)^{H-1}$$

and ε_1^* is a standard S α S random variable independent of (W_1, \dots, W_m) .

Proposition 1 ensures that the obtained sample path converges in the sense of finite-dimensional distributions to the LFSM $\{\xi_{H,\alpha}(t), 0 \leq t \leq 1\}$ with error bound given by (17). It is important to note that this algorithm requires $\mathcal{O}(m)$ memory space and has $\mathcal{O}(m \log m)$ time complexity (as a result of the use of the FFT).

Another interesting feature of this algorithm is that it allows one to *simultaneously* obtain $L = \lfloor m/n \rfloor$ sample paths of LFSM, thus, significantly reducing the cost for producing input traces for network simulations. Actually, the L vectors

$$V_\ell = (W_j + c^* \varepsilon_\ell^*, 1 + (\ell-1)n \leq j \leq \ell n), \quad \ell = 1, 2, \dots, L$$

are identically distributed by appropriate permutations of ε_i , where ε_ℓ^* are i.i.d. and are independent of $\{\varepsilon_i\}_{i \in \mathcal{Z}}$. Then L identically distributed LFSM are obtained by taking cumulative sums of each block. However, it is important to note that these vectors are *not independent*.

Remark 3: The dependence between the various blocks being generated is difficult to characterize. Nevertheless, as the next calculation shows, the entry-wise dependence is in a certain sense weak. Let $W_1 + c^* \varepsilon_1$ and $W_n + c^* \varepsilon_2$ be the first elements in V_1 and V_2 . Elementary but tedious manipulations (that are omitted) show that as $n \rightarrow \infty$

$$\begin{aligned} & \sup_{\lambda_1, \lambda_2 \in \mathcal{R}} |\mathbf{E} \exp[i\lambda_1(W_1 + c^* \varepsilon_1) + i\lambda_2(W_n + c^* \varepsilon_2)] \\ & - \mathbf{E} \exp[i\lambda_1(W_1 + c^* \varepsilon_1)] \mathbf{E} \exp[i\lambda_2(W_n + c^* \varepsilon_2)]| \\ & \leq \sup_{\lambda_1, \lambda_2} |\mathbf{E} \exp[i\lambda_1 W_1 + i\lambda_2 W_n]| \\ & - \mathbf{E} \exp(i\lambda_1 W_1) \times \mathbf{E} \exp(i\lambda_2 W_n)| \\ & \rightarrow 0. \end{aligned}$$

E. Two-Sided Linear Processes

In the Gaussian case ($\alpha = 2$), for a given $H \in (0, 1)$ there is essentially a unique, in the sense of finite-dimensional distributions, self-similar process with stationary increments, which up to a multiplicative scale factor corresponds to the fBm with Hurst's index H . However, there are infinitely many essentially different such processes when $0 < \alpha < 2$; see Samorodnitsky and Taqqu [31]. For example, if $b_1 b_2' - b_1' b_2 \neq 0$, then the LFSM $\xi_{H,\alpha}^{b_1', b_2'}$ and $\xi_{H,\alpha}^{b_1, b_2}$ are different.

A slightly modified algorithm can be used to generate $\xi_{H,\alpha}^{b_1, b_2}$ for all $b_1, b_2 \in \mathcal{R}$. To this end, we introduce the coefficients $a'_n = b_1 a_n$ and $a'_{-n} = b_2 a_n$ for $n \geq 1$, $a'_0 = 0$. As in (3)–(5), let

$$\varepsilon_{k,n} = n^{1/\alpha} [M_\alpha(k/n) - M_\alpha((k-1)/n)]$$

$$X'_{k,n} = \sum_{i=-\infty}^{\infty} a'_i \varepsilon_{k-i,n}$$

and

$$\begin{aligned} Y'_{k,n} = n^H \int_{\mathcal{R}} & b_1 [(k/n - s)_+^\beta - ((k-1)/n - s)_+^\beta] \\ & + b_2 [(k/n - s)_-^\beta - ((k-1)/n - s)_-^\beta] M_\alpha(ds). \end{aligned}$$

A similar version of Theorem 1 implies that

$$\max_{1 \leq K \leq n} \left\| \sum_{i=1}^K (X'_{k,n} - Y'_{k,n}) \right\|_\alpha = \mathcal{O}(1).$$

As in Lemma 1, we can approximate the tail

$$\psi'_{k,m,n} = \sum_{j=m+1-n}^{\infty} a'_{k+j} \varepsilon_{-j,n}$$

by $\psi'_{1,m,n}$ and the other tail

$$\phi'_{k,m,n} = \sum_{j=-\infty}^{n-m-1} a'_{k+j} \varepsilon_{-j,n}$$

by $\phi'_{1,m,n}$. Then

$$\max_{1 \leq K \leq n} \left\| \sum_{k=1}^K \psi'_{k,m,n} - K \psi'_{1,m,n} \right\| = \mathcal{O}(n^2 m^{\beta-2+1/\alpha})$$

and

$$\max_{1 \leq K \leq n} \left\| \sum_{k=1}^K \phi'_{k,m,n} - K \phi'_{1,m,n} \right\| = \mathcal{O}(n^2 m^{\beta-2+1/\alpha}).$$

As in Theorem 2, for $1 \leq k \leq m$, the “main part”

$$X_k - \psi'_{k,m,n} - \phi'_{k,m,n} = \sum_{j=n-m}^{m-n} a'_{k+j} \varepsilon_{-j,n}$$

can be approximated by $W'_{k,m,n}$ with approximation error $\mathcal{O}(n^{1/\alpha+1} m^{\beta-1})$ in view of Lemma 2. Here, as in (23), we can write the vector

$$(W'_{1,m,n}, \dots, W'_{m,m,n})^T = A'_m \mathbf{e}_{2m+1}^T$$

where A'_m is a circulant matrix generated by the vector $\mathbf{a}' = (a'_{-m}, a'_{-m+1}, \dots, a'_m)$ and \mathbf{e}_{2m+1} is a $2m+1$ -dimensional random vector with i.i.d. standard S α S random variables as elements. Thus, Algorithm 1 can similarly be applied to generate the random vector $(W'_{1,m,n}, \dots, W'_{m,m,n})$.

In summary, let $\{\varepsilon_i^\# \}_{i \in \mathcal{Z}}$, $\{\varepsilon_i\}_{i \in \mathcal{Z}}$, and $\{\varepsilon_i^* \}_{i \in \mathcal{Z}}$ be i.i.d. sequences, $c_1^* = c^* b_1$ and $c_2^* = c^* b_2$. Then the cumulative sum of

$$n^{-H} \left(b_2 c^* \varepsilon_1^\# + W_j + b_1 c^* \varepsilon_1^* \right), \quad 1 \leq j \leq n$$

converges in the sense of finite-dimensional distributions to the LFSM $\{\xi_{H,\alpha}^{b_1, b_2}(t), 0 \leq t \leq 1\}$ with error bound given by (17).

F. Matlab Code

In this subsection, we provide a Matlab program. The Matlab program `salphas(alpha,nbp)` generates N i.i.d. standard SoS random variables (cf Chambers *et al.* [8], [9]), while the program `LFSM(H,alpha,m,n)` returns $L = \lfloor m/n \rfloor$ identically (however, not independently) distributed sample paths of LFSM with number of grids n . On a Pentium III 1-GHz machine with 1.5 GB of memory, the running times for LFSM (0.75, 1.5, 4000000, 50000) and LFSM (0.75, 1.5, 20000000, 50000) are about 30 and 300 s, respectively. In the latter case $20000000/50000 = 400$ sample paths are generated by the proposed algorithm.

```
function r = salphas(alpha, nbp)
if alpha > 2 error ('alpha must be less
than or equal to 2!')
r = 0;
elseif alpha == 2
r = sqrt(2)* randn (nbp ,1);
elseif alpha == 1
r = tan (pi* (rand (nbp, 1)
-0.5* ones (nbp ,1)));
elseif alpha > 0
u = pi*(rand (nbp, 1) - 0.5*ones (nbp,1));
w = -log (rand (nbp,1));
r = ((cos ((1 - alpha)* u) ./w).^ (1/alpha - 1)
.* sin (alpha*u) ./ cos (u) .^ (1/alpha));
else error ('alpha must be positive!')
end r = real(r);
function X = LFSM (H,alpha,m,n)
A(1) = 1;
for j = 2:m
A (j) = j^ (H - 1/alpha) - (j - 1)^(H - 1/alpha);
end
L = floor(m/n);
c = (m - n)^(H - 1)* abs (H - 1/alpha)
*(alpha* (1 - H)) ^(-1/alpha);
r = salphas (alpha,m);
Y = real (ifft (fft (transpose (A))
.* fft(r)));
psi = stabrnd (alpha, 0,1,0,L,1);
X = zeros (n,L);
YY = reshape(Y,[n 1]);
YY = YY'*c*psi*ones(1,n);
X = cumsum(YY'); X = X/n^H;
```

III. COMPARISON WITH A STOCHASTIC INTEGRAL APPROXIMATION PROCEDURE

A. A Simple Approximation Algorithm

By discretizing the stochastic integral representation of the stationary increment process

$$\xi_{H,\alpha}(j) - \xi_{H,\alpha}(j-1), \quad 1 \leq j \leq T$$

Samorodnitsky and Taqqu [31] proposed the following two-step approximation procedure to simulate LFSM (cf [31, Ch. 7.11,

pp. 370–376]). Let $K_\beta(x) = x_+^\beta - (x-1)_+^\beta$. Then the fractional stable noise can be written as

$$\begin{aligned} Y_j &= \int_{\mathcal{R}} K_\beta(j+1-x)M_\alpha(dx) \\ &\approx \sum_{k \in \mathcal{Z}} K_\beta(j+1-k/l)[M_\alpha((k+1)/l) - M_\alpha(k/l)] \\ &= \sum_{u \in \mathcal{Z}} K_\beta(u/l)[M_\alpha(j+1-u/l+1/l) \\ &\quad - M_\alpha(j+1-u/l)] \end{aligned} \quad (27)$$

where $1/l$ is the mesh corresponding to the discretization of the integral. In the second step of the approximation, the latter sum is truncated to $1 \leq u \leq Ll$

$$\check{Y}_j = \sum_{u=1}^{Ll} K_\beta(u/l) \times [M_\alpha(j+1-u/l+1/l) - M_\alpha(j+1-u/l)] \quad (28)$$

where L is the cutoff in the limits of integration, which defines the memory (cf [31, eq. (7.11.1)]). Notice that

$$\eta_{j-u/l} = l^\alpha [M(j+1-u/l+1/l) - M(j+1-u/l)]$$

$1 \leq j \leq T$, $1 \leq u \leq Ll$ are $l(T+L-1)$ i.i.d. $S_\alpha(1,0,0)$ random variables. One can then generate $l(T+L-1)$ i.i.d. $S_\alpha(1,0,0)$ random variables and use the partial sum $\sum_{j=1}^{\lfloor Tt \rfloor} \check{Y}_j/T^H$, $0 \leq t \leq 1$ to obtain an approximated LFSM path; see the description of the algorithm in [31] for additional details.

The proposed procedure in this paper differs from Samorodnitsky and Taqqu's two-step approximation approach in several important aspects. The first step in both approaches is based on discretizing the stochastic integral representation of the process; see our Theorem 1, where an error bound is given for approximating stochastic integrals $Y_{k,n}$ by linear processes $X_{k,n}$. The roles of m and n in our approximation procedure correspond to Ll and l , respectively, in Samorodnitsky and Taqqu's algorithm. In the second step of the Samorodnitsky–Taqqu approximation (28), the two tails $\sum_{u \leq 0} K_\beta(u/l)\varepsilon_{j-u/l}$ and $\sum_{u > Ll} K_\beta(u/l)\varepsilon_{j-u/l}$ are discarded. As discussed in our Remark 1, the accuracy of our procedure is substantially improved by approximating $\psi_{k,m,n}$ by $\psi_{1,m,n}$.

For the computation of \check{Y}_j , noticing its convolution structure between the kernel $K_\beta(u/l)$ and $\varepsilon_{j-u/l}$, one can also use the FFT algorithm. Recently, Stoev and Taqqu (2003) pursued this idea and proposed an analogous algorithm to the one presented in this paper, by padding a significant number of zeroes to the vector of values of the kernel $K_\beta(u/l)$ so that the FFT algorithm becomes applicable. The Stoev–Taqqu algorithm has computational complexity $\mathcal{O}[(Ll) \log(Ll)]$ and the order of the approximation error is given by $\mathcal{O}[l^{-H \min(1,\alpha)} + L^{-(1-H) \min(1,\alpha)}]$; see [30, Corollary 2.1]. Let $p_1 = H \min(1,\alpha)$ and $p_2 = (1-H) \min(1,\alpha)$. In order for Stoev–Taqqu algorithm to reach a pre-assigned accuracy level $\delta > 0$ in the sense of (17) and (19), one requires

$$l^{-p_1} + L^{-p_2} \leq \delta. \quad (29)$$

TABLE I
THE VALUES OF m , J , AND $\rho = (m \log m)/(J \log J)$ ARE GIVEN BY (20) AND (30), FOR THE LÉVY PARAMETER $\alpha = 0.5, 1.5$.
THE APPROXIMATION ERROR $\delta = 0.1, 0.01$ AND THE HURST INDEX $H = 0.1, 0.3, 0.5, 0.7, 0.9$

		$\alpha = .5$			$\alpha = 1.5$		
δ	H	m	J	ρ	m	J	ρ
.1		2.7×10^{11}	2.2×10^{25}	5.3×10^{-15}	4.8×10^{11}	4.7×10^{12}	9.3×10^{-2}
.1	.3	4.3×10^4	1.1×10^{12}	1.5×10^{-8}	7.8×10^4	1.0×10^6	6.0×10^{-2}
	.5	2.0×10^3	2.5×10^{10}	2.5×10^{-8}	4.1×10^3	1.6×10^5	1.7×10^{-2}
	.7	6.5×10^2	1.1×10^{12}	1.3×10^{-10}	1.5×10^3	1.0×10^6	7.7×10^{-4}
	.9	4.2×10^2	2.2×10^{25}	1.8×10^{-24}	1.3×10^3	4.7×10^{12}	7.1×10^{-11}
.1		9.1×10^{21}	3.8×10^{47}	1.1×10^{-26}	2.1×10^{22}	6.1×10^{23}	3.2×10^{-2}
.01	.3	3.6×10^8	3.7×10^{21}	3.8×10^{-14}	9.1×10^8	6.1×10^{10}	1.2×10^{-2}
	.5	9.6×10^5	2.5×10^{18}	1.2×10^{-13}	2.9×10^6	1.6×10^9	1.3×10^{-3}
	.7	1.0×10^5	3.7×10^{21}	6.3×10^{-18}	4.5×10^5	6.1×10^{10}	3.8×10^{-6}
	.9	4.4×10^4	3.8×10^{47}	1.1×10^{-44}	3.5×10^5	6.1×10^{23}	1.3×10^{-19}

Let

$$C(p_1, p_2) = (p_1/p_2)^{-p_1/(p_1+p_2)} + (p_1/p_2)^{p_2/(p_1+p_2)}$$

and $J = Ll$. Similarly to the derivation of (20), we have the inequality

$$l^{-p_1} + (J/l)^{-p_2} \geq C(p_1, p_2)J^{-p_1 p_2/(p_1+p_2)}$$

which due to (29) implies that

$$J \geq [\delta^{-1}C(p_1, p_2)]^{1/p_1+1/p_2} \quad (30)$$

and the corresponding computational complexity is given by

$$\mathcal{O}(J \log J) = \mathcal{O}[(\delta^{-1})^{1/p_1+1/p_2} \log \delta^{-1}], \quad \text{as } \delta \rightarrow 0.$$

For the same level of accuracy $\delta > 0$, Proposition 1 and (20) show that the proposed algorithm has computational complexity

$$\mathcal{O}(m \log m) = \mathcal{O}[(\delta^{-1})^{1/H+1/p(\alpha, H)} \log \delta^{-1}].$$

Since $1/H + 1/p(\alpha, H) < 1/p_1 + 1/p_2$, our algorithm is computationally less costly. Table I gives the values of m , J , and $\rho = (m \log m)/(J \log J)$ for a few selected values of the Lévy parameter $\alpha = 0.5, 1.5$, the approximation error $\delta = 0.1, 0.01$, and the Hurst index $H = 0.1, 0.3, 0.5, 0.7, 0.9$. The ratio ρ reflects the reduction in computational complexity. As shown by the entries of Table I, the gains of our algorithm become more prominent for larger values of H and/or α is small.

In summary, the inclusion of the correction term $\psi_{1,m,n}$ enables the efficient computation, and Proposition 1 provides a theoretical justification for the obtained accuracy.

Remark 4: In a recent paper, Stoev *et al.* [29] present an excellent treatment of the estimation problem of the self-similarity parameter H for sample paths of LFSM. In particular, two estimators are proposed: the first one is based on the finite impulse response transformation (FIRT) (H_{FIRT}) and the second one on a wavelet transformation (WT) (H_{FW}). In their simulation study, the Samorodnitsky–Taqqu algorithm is used and the bias, standard deviation, and asymptotic distributions of H_{FIRT}

and H_{WT} are discussed. It is interesting to note that based on the sample paths generated by our algorithm, the estimator of H based on the FIRT (or WT) method has the same distribution as H_{FIRT} (or H_{WT}) if the number of zero moments $Q \geq 2$ (cf [29]). Actually, approximating $\psi_{k,m,n}$ by $\psi_{1,m,n}$ results in a linear trend in the partial sum process, which vanishes under the FIRT and the WT. Hence, our algorithm improves in accuracy in the sense of (17), but does not offer improvements in estimating H . \square

B. Discussion and Concluding Remarks

An efficient method is proposed for generating sample paths of LFSM. One of the main points of our study is that it provides a detailed estimation of the error bound obtained by the proposed finite approximation of the infinite term linear process. This derivation provides the user with practical guidelines on how to select the appropriate embedding dimension m when one is interested in generating a sample path of size n with a given degree of accuracy δ . The proposed method allows one to generate multiple identically distributed traces, that can be used in parallel simulation scenarios of queueing systems, thus leading to additional computational savings.

Several additional research threads are currently being pursued, including the use of similar ideas in the generation of higher dimensional processes which are of interest to geophysicists and environmental scientists.

IV. PROOF OF THEOREM 1

Let $a_i = 0$ for $i \leq 0$ and

$$f(s) = f_{K/n}(s) = (K/n - s)_+^\beta - (-s)_+^\beta.$$

Then

$$\sum_{k=1}^K Y_{k,n} = n^H \int_{\mathcal{R}} f(s) M_\alpha(ds)$$

and

$$\sum_{k=1}^K X_{k,n} = \sum_{i \in \mathcal{Z}} \left(\sum_{k=1}^K a_{k-i} \right) \varepsilon_{i,n}.$$

Let

$$g_i(s) = f(s) - n^{-\beta} \sum_{k=1}^K a_{k-i}.$$

Since $\varepsilon_{k,n} = n^{1/\alpha} \int_{(k-1)/n}^{k/n} M_\alpha(ds)$

$$\sum_{k=1}^K (Y_{k,n} - X_{k,n}) = n^H \sum_{i \in \mathcal{Z}} \int_{(i-1)/n}^{i/n} g_i(s) M_\alpha(ds).$$

Observe that $g_i(s) \equiv 0$ when $i \geq K + 1$. To prove (6), it thus suffices to establish

$$\begin{aligned} \sum_{i \in \mathcal{Z}} \int_{(i-1)/n}^{i/n} |g_i(s)|^\alpha ds &= \sum_{i \leq K} \int_{(i-1)/n}^{i/n} |g_i(s)|^\alpha ds \\ &= \mathcal{O}(n^{-\alpha H}). \end{aligned} \quad (31)$$

If $i = 0$, then for $-1/n \leq s < 0$ and $1 \leq K \leq n$

$$|(K/n - s)_+^\beta - (K/n)^\beta| = \mathcal{O}[|s|(K/n)^{\beta-1}] = \mathcal{O}[|s|(1/n)^{\beta-1}]$$

and

$$\int_{-1/n}^0 |(-s)_+^\beta|^\alpha ds = \mathcal{O}(n^{-\alpha H}).$$

Using $|\phi_1 + \phi_2|^\alpha \leq c_\alpha (|\phi_1|^\alpha + |\phi_2|^\alpha)$, where $c_\alpha = \max(2^{\alpha-1}, 1)$, we have

$$\begin{aligned} \int_{-1/n}^0 |g_0(s)|^\alpha ds &= \int_{-1/n}^0 |f(s) - (K/n)^\beta|^\alpha ds \\ &\leq c_\alpha \int_{-1/n}^0 \left[|(K/n - s)_+^\beta - (K/n)^\beta|^\alpha \right. \\ &\quad \left. + |(-s)_+^\beta|^\alpha \right] ds \\ &= \mathcal{O}(n^{-\alpha H}). \end{aligned}$$

If $i = K$, then $\sum_{k=1}^K a_{k-i} = 0$ and when $(K-1)/n \leq s < K/n$, $g_i(s) = (K/n - s)_+^\beta$. So

$$\int_{(K-1)/n}^{K/n} |g_i(s)|^\alpha ds = \mathcal{O}(n^{-\alpha H}).$$

For $i \leq -1$, notice that

$$\sum_{k=1}^K a_{k-i} = (K-i)^\beta - (-i)^\beta$$

and when $(i-1)/n \leq s < i/n$, $|(-s)^\beta - (-i/n)^\beta| = \mathcal{O}[(-i/n)^{\beta-1}|s - i/n|]$. Thus,

$$\begin{aligned} \sum_{i \leq -1} \int_{(i-1)/n}^{i/n} |g_i(s)|^\alpha ds &\leq \sum_{i \leq -1} c_\alpha \int_{(i-1)/n}^{i/n} \left[|(-s)^\beta - (-i/n)^\beta|^\alpha \right. \\ &\quad \left. + |(K/n - s)^\beta - ((K-i)/n)^\beta|^\alpha \right] ds \\ &\leq 2c_\alpha \sum_{i \leq -1} \int_{(i-1)/n}^{i/n} |(-s)^\beta - (-i/n)^\beta|^\alpha ds \\ &= 2c_\alpha \sum_{i \leq -1} \int_{(i-1)/n}^{i/n} \mathcal{O}[(-i/n)^{\beta-1}|s - i/n|]^\alpha ds \\ &= \mathcal{O}(n^{-\alpha H}). \end{aligned}$$

Now it remains to verify (31) when $1 \leq i \leq K-1$. In this case, $\sum_{k=1}^K a_{k-i} = (K-i)^\beta$ and hence,

$$|g_i(s)| = \mathcal{O}[|(K-i)/n|^{\beta-1}|s - i/n|].$$

Therefore,

$$\begin{aligned} \sum_{1 \leq i \leq K-1} \int_{(i-1)/n}^{i/n} |g_i(s)|^\alpha ds &= \sum_{1 \leq i \leq K-1} \mathcal{O}[|(K-i)/n|^{\alpha(\beta-1)}] \int_{(i-1)/n}^{i/n} |s - i/n|^\alpha ds \\ &= \mathcal{O}(n^{-\alpha H}). \end{aligned} \quad \square$$

ACKNOWLEDGMENT

The authors would like to thank the two referees for their very helpful comments and suggestions that led to a significant improvement of the paper. The current version of the Matlab algorithm was provided by one referee.

REFERENCES

- [1] P. Abry and F. Sellan, "The wavelet based synthesis for fractional Brownian motion," *Appl. Comput. Harmonic Anal.*, vol. 3, pp. 377–383, 1996.
- [2] P. Abry and V. Veitch, "Wavelet analysis of long-range dependent traffic," *IEEE Trans. Inform. Theory*, vol. 44, pp. 2–15, Jan. 1998.
- [3] F. Avrami and M. S. Taqqu, "Weak convergence of moving averages with infinite variance," in *Dependence in Probability and Statistics: A Survey of Recent Results*, E. Eberlein and M. S. Taqqu, Eds. Boston, MA, Birkhauser, 1986, pp. 399–416.
- [4] —, "Weak convergence of sums of moving averages in the α -stable domain of attraction," *Ann. Probab.*, vol. 20, pp. 483–503, 1992.
- [5] A. Astrauskas, "Limit theorems for sums of linearly generated random variables," *Lithuanian Math. J.*, vol. 23, pp. 123–134, 1983.
- [6] J. Beran, *Statistics for Long Memory Processes*. London, U.K.: Chapman & Hall, 1994.
- [7] J. F. Coeurjolly, "Simulation and identification of the fractional Brownian motion: A bibliographical and comparative study," *J. Statist. Software*, vol. 5, no. 7, 2000.
- [8] J. M. Chambers, C. L. Mallows, and B. W. Stuck, "A method for simulating stable random variables," *J. Amer. Statist. Assoc.*, vol. 71, pp. 340–344, 1976.
- [9] —, "A method for simulating stable random variables," *J. Amer. Statist. Assoc.*, vol. 82, p. 704, 1987.
- [10] Y. A. Davydov, "The invariance principle for stationary processes," *Theory Probab. Applic.*, vol. 15, pp. 487–498, 1970.
- [11] C. R. Dietrich and G. N. Newsam, "Fast and exact simulation of stationary Gaussian processes through circulant embedding of the covariance matrix," *SIAM J. Scient. Comput.*, vol. 18, pp. 1088–1107, 1997.
- [12] P. Flandrin, "Wavelet analysis and synthesis of fractional Brownian motion," *IEEE Trans. Inform. Theory*, vol. 38, pp. 910–917, Mar. 1992.
- [13] J. R. Gallardo, D. Makrakis, and L. Orozco-Bardosa, "Use of α -stable self-similar stochastic processes for modeling traffic in broadband networks," *Perform. Eval.*, vol. 40, pp. 71–98, 2000.
- [14] S. Giordano, S. Porcarelli, and G. Procissi, "Testing alpha-stable processes in modeling broadband teletraffic," in *Proc. IEEE Int. Conf. Communications '00*, New Orleans, LA, June 2000, pp. 18–22.
- [15] F. C. Harmantzis, C. Hatzinakos, and I. Katzela, "Tail probabilities for the multiplexing of fractional α -stable broadband traffic," in *Proc. IEEE Int. Conf. Communications '01*, Helsinki, Finland, June 2001, pp. 2665–2669.
- [16] J. Istas and G. Lang, "Variations quadratiques et estimation de l'exposant de holder local d'un processus Gaussien," *Ann. de l'Institut Henri Poincare*, vol. 33, pp. 407–436, 1997.
- [17] A. Karasaridis and D. Hatzinakos, "Network traffic modeling using α -stable self-similar processes," *IEEE Trans. Commun.*, vol. 49, pp. 1203–1214, July 2001.
- [18] J. T. Kent and A. Wood, "Estimating the fractal dimension of a locally self-similar Gaussian process using increments," *J. Roy. Statist. Soc., B*, vol. 59, pp. 679–700, 1997.

- [19] A. N. Kolmogorov, "Winersche Spiralen und einige andere Interessante Kurven in Hilbertschen Raum," *S. R. Academic Sci., U.S.S.R.*, vol. 26, pp. 115–118, 1940.
- [20] T. Konstantopoulos and S. J. Lin, "Macroscopic models for long-range dependent network traffic," *Queueing Syst.. Theory and Applic.*, vol. 28, pp. 215–243, 1998.
- [21] W. Leland, M. S. Taqqu, W. Willinger, and D. Wilson, "On the self-similar nature of Ethernet traffic," *IEEE/ACM Trans. Networking*, vol. 2, pp. 1–15, Feb. 1994.
- [22] B. Mandelbrot and W. J. Van Ness, "Fractional Brownian motions, fractional noises and applications," *SIAM Rev.*, vol. 10, pp. 422–437, 1969.
- [23] T. Mikosch, S. Resnick, H. Rootzén, and A. Stegeman, "Is network traffic approximated by stable Lévy motion or fractional Brownian motion?," *Ann. Appl. Probab.*, vol. 12, pp. 23–68, 2002.
- [24] I. Norros, "On the use of fractional Brownian motion in the theory of connectionless networks," *IEEE J. Select. Areas Commun.*, vol. 13, pp. 953–962, Aug. 1995.
- [25] I. Norros, P. Mannersalo, and J. L. Wang, "Simulation of fractional Brownian motion with conditionalized midpoint displacement," *Adv. Perform. Anal.*, vol. 2, pp. 77–101, 1999.
- [26] K. Park and W. Willinger, Eds., *Self-Similar Network Traffic and Performance Evaluation*. New York: Wiley, 2000.
- [27] V. Paxson and S. Floyd, "Wide-area traffic: The failure of Poisson modeling," *IEEE/ACM Trans. Networking*, vol. 3, pp. 226–244, June 1995.
- [28] V. Paxson, "Fast, approximate synthesis of fractional Gaussian noise for generating self-similar network traffic," *Computer Commun. Rev.*, vol. 27, pp. 5–18, 1997.
- [29] S. Stoev, V. Pipiras, and M. S. Taqqu, "Estimation of the self-similarity parameter in linear fractional stable motion," *Signal Processing*, vol. 82, pp. 1873–1901, 2002.
- [30] S. Stoev and M. S. Taqqu, "Simulation methods for linear fractional stable motion and FARIMA using the fast Fourier transform," *Fractals*, 2004, to be published.
- [31] G. Samorodnitsky and M. S. Taqqu, *Stable Non-Gaussian Random Processes*. London, U.K.: Chapman & Hall, 1994.
- [32] M. S. Taqqu, V. Teverovsky, and W. Willinger, "Is network traffic self-similar or multifractal?," *Fractals*, vol. 5, pp. 63–73, 1997.
- [33] M. S. Taqqu, W. Willinger, and R. Sherman, "Proof of a fundamental result in self-similar traffic modeling," *Computer Commun. Rev.*, vol. 27, pp. 5–23, 1997.
- [34] M. Yuksel, B. Sikdar, K. S. Vastola, and B. Szymanski, "Workload generation for ns simulations of wide area networks and the internet," in *Proc. Communication Networks and Distributed Systems Modeling and Simulation Conf.*, San Diego, CA, 2000, pp. 93–98.
- [35] W. Willinger, M. S. Taqqu, W. Leland, and D. Wilson, "Self-similarity in high-speed packet traffic: Analysis and modeling of ethernet traffic measurements," *Statist. Sci.*, vol. 10, pp. 67–85, 1995.
- [36] A. Wood and G. Chan, "Simulation of stationary Gaussian processes in $[0, 1]^d$," *J. Comput. Graph. Statist.*, vol. 3, pp. 409–432, 1994.