Simulating Sample Paths of Linear Fractional Stable Motion[1]

October 8, 2003

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 a 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.**

Key Words–Simulation, linear fractional stable motion, linear process, heavy-tailed distributions, Fast Fourier transform, long-range dependence.

1. 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, non-stationary, 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

[1]W. B. Wu is with Department of Statistics, The University of Chicago, Chicago, IL 60637, USA (email: wbwu@galton.uchicago.edu).

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

D. Zhang is with Department of EECS, The University of Michigan, Ann Arbor, MI 48109, USA (email: danlu@umich.edu)

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.

Linear fractional stable motion, (LFSM, also known as fractional Levy motion) a process exhibiting such characteristics, is defined next. A random variable η is called *symmetric stable* with index $\alpha \in (0,2]$ (SαS) if $\eta =_d \sigma\varepsilon$, where $\sigma \geq 0$ is the scale coefficient and ε is a random variable with characteristic function $\mathbb{E}\exp(\imath\lambda\varepsilon) = \exp(-|\lambda|^\alpha)$ [31] ($\imath = \sqrt{-1}$ represents the imaginary unit). Write $\|\eta\|_\alpha = \sigma$ for the scale parameter σ. Let $\{M_\alpha(s),\ s \in \mathbb{R}\}$ denote a Levy α-stable motion; namely, $M_\alpha(\cdot)$ is a stochastic process that has stationary and independent SαS increments ($M_\alpha(s+h) - M_\alpha(s)$ is SαS) with scale parameter $\|M_\alpha(s+h) - M_\alpha(s)\|_\alpha = h^{1/\alpha}$, $h \geq 0$. Let $M_\alpha(ds)$ be the corresponding SαS random measure with control measure ds. When $\alpha = 2$, the rescaled version $\{\sqrt{2}M_\alpha(s),\ s \in \mathbb{R}\}$ corresponds to standard Brownian motion. Let $-1/\alpha < \beta < 1 - 1/\alpha$, $x_+ = \max(0,x)$ and $x_- = \max(0,-x)$. For $b_1, b_2 \in \mathbb{R}$, define the LFSM $\xi_{H,\alpha}^{b_1,b_2} = \{\xi_{H,\alpha}^{b_1,b_2}(t),\ t \in \mathbb{R}\}$ via the stochastic integral representation

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

The resulting process $\xi_{H,\alpha}^{b_1,b_2}$ has stationary increments and is self-similar in the sense that for $c > 0$, $\{\xi_{H,\alpha}^{b_1,b_2}(ct),\ t \in \mathbb{R}\}$ has the same distribution as $\{c^H \xi_{H,\alpha}^{b_1,b_2}(t),\ t \in \mathbb{R}\}$, 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 Levy 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 there have appeared several studies of queuing performance under a self-similar stable motion input [13, 14, 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 3.

Our approach allows one to generate a LFSM efficiently, since it has linear memory requirements and a competitive time complexity (for details see Section 2), similar to the efficient algorithm proposed in [36]. The paper is organized as follows: in Section 2, the theoretical development and the proposed algorithm are given, while the corresponding proofs are presented in Section 4. Finally, some concluding remarks are drawn in Section 3.2.

2. The Algorithm and a Limit Theorem

2.1. **Preliminaries.** Let $b_1 = 1$ and $b_2 = 0$ in (1) and write $\xi_{H,\alpha} = \xi^{1,0}_{H,\alpha}$. The general case with $b_1, b_2 \in \mathbb{R}$ is discussed in Section 2.5. Let $\{\varepsilon_i\}_{i\in\mathbb{Z}}$ be a sequence of independent and identically distributed (iid) standard (namely the scale $\|\varepsilon_i\|_\alpha = 1$) S$\alpha$S random variables, with $\alpha \in (0, 2]$. Define the sequence $\{a_n\}_{n\in\mathbb{Z}_+}$ 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])

$$X_n = a_1\varepsilon_{n-1} + a_2\varepsilon_{n-2} + a_3\varepsilon_{n-3} + \ldots = \sum_{i=1}^{\infty} a_i\varepsilon_{n-i}. \tag{2}$$

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 Kolmogorov's Three Series Theorem. Moreover, by the definition of SαS distributions, X_n has the same distribution as $(\sum_{i=1}^\infty |a_i|^\alpha)^{1/\alpha}\varepsilon_0$ and in addition we have that $\|X_n\|_\alpha = (\sum_{i=1}^\infty |a_i|^\alpha)^{1/\alpha}$. Let $S_n(t) = \sum_{j=1}^{\lceil nt\rceil} X_j = \sum_{i=0}^\infty X_i(\sum_{j=1-i}^{\lceil nt\rceil - i} a_j)$, $0 < t \leq 1$ 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, 4, 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 2.6 with $m = 2,000,000$ and $n = 10,000$, sample paths of LFSM for different values of the Hurst and Levy parameters (H and α) are shown in Figure 1. It is known that for smaller H, the sample paths are discontinuous (cf Chapter 10, Samorodnitsky and Taqqu (1994)) due to heavy tails. The heaviness of the tails of the marginal distribution induces very large bursts, especially for $\alpha < 1$.

[**Insert Figure 1 about here**]

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 below. 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.

2.2. **Asymptotic Results.** Recall that $M_\alpha(ds)$ denotes an SαS random measure with control measure ds. Then, for each $n \geq 1$ we have that

$$\varepsilon_{k,n} = n^{1/\alpha}[M_\alpha(\frac{k}{n}) - M_\alpha(\frac{k-1}{n})], \; k \in \mathbb{Z} \tag{3}$$

are iid standard SαS random variables. Define next the following variables

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

and

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

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

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

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

Notice that $\{\sum_{k=1}^K Y_{k,n}/n^H, \; 0 \leq K \leq n\} =_d \{\xi_{H,\alpha}(K/n), \; 0 \leq K \leq n\}$. 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 4.

2.3. **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 $m/n \to \infty$ as $n \to \infty$. The first step is to approximate the tail processes $\psi_{k,m,n} = \sum_{j=m+1-n}^{\infty} a_{k+j}\varepsilon_{-j}$, $1 \leq k \leq n$ 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 (12)) whose computation can be carried out in a sufficiently fast manner (cf Section 2.4).

LEMMA **1**. *Assume m, $n \to \infty$ and $m/n \to \infty$. Then, for the truncated process*

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

we have that

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

Proof of Lemma 1. Observe that

$$\|\sum_{k=1}^{K} \psi_{k,m,n} - K\psi_{1,m,n}\|_\alpha^\alpha = \sum_{j=m+1-n}^{\infty} |(K+j)^\beta - j^\beta - K[(1+j)^\beta - j^\beta]|^\alpha.$$

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

LEMMA **2**. *Define*

$$\begin{pmatrix} U_{1,m,n} \\ U_{2,m,n} \\ \dots \\ U_{n-1,m,n} \\ U_{n,m,n} \end{pmatrix} = \begin{pmatrix} a_{m-n+2} & a_{m-n+3} & \dots & a_{m-1} & a_m & 0 \\ a_{m-n+3} & a_{m-n+4} & \dots & a_m & 0 & 0 \\ \dots & \dots & \dots & 0 & 0 & 0 \\ a_m & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & \dots & 0 & 0 \end{pmatrix} \begin{pmatrix} \varepsilon_{n-1} \\ \varepsilon_{n-2} \\ \dots \\ \varepsilon_1 \\ \varepsilon_0 \end{pmatrix} \tag{9}$$

Then

$$\max_{1\le K\le n} \|\sum_{k=1}^{K} U_{k,m,n}\|_\alpha = \mathcal{O}(n^{1/\alpha+1} m^{\beta-1}). \tag{10}$$

Proof of Lemma 2. Since $\sum_{j=N}^{m} a_j = m^\beta - (N-1)^\beta$ for $m \geq N \geq 2$, (10) follows from

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

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

Lemma **3**. *Let $\{\varepsilon_n,\ n \in \mathbb{Z}\}$ be a sequence of iid standard SαS random variables; further let $\{c_n,\ n \in \mathbb{Z}\}$ and $\{d_n,\ n \in \mathbb{Z}\}$ be two sequences of real numbers such that $\sum_{n\in\mathbb{Z}}(|c_n|^\alpha + |d_n|^\alpha) < \infty$. Finally, define $\eta_1 = \sum_{n\in\mathbb{Z}} c_n\varepsilon_{-n}$ and $\eta_2 = \sum_{n\in\mathbb{Z}} d_n\varepsilon_{-n}$. We then have that*

$$\|\eta_1 + \eta_2\|_\alpha \leq \max(2, 2^{1/\alpha})(\|\eta_1\|_\alpha + \|\eta_2\|_\alpha). \tag{11}$$

Proof of Lemma 3. Using $|x+y|^\alpha \leq c_\alpha(|x|^\alpha + |y|^\alpha)$, where $c_\alpha = \max(2^{\alpha-1}, 1)$, we have that

$$\|\eta_1 + \eta_2\|_\alpha^\alpha = \sum_{n\in\mathbb{Z}} |c_n + d_n|^\alpha \leq c_\alpha \sum_{n\in\mathbb{Z}}(|c_n|^\alpha + |d_n|^\alpha) = c_\alpha(\|\eta_1\|_\alpha^\alpha + \|\eta_2\|_\alpha^\alpha),$$

which yields (11) since $\|\eta_1\|_\alpha^\alpha + \|\eta_2\|_\alpha^\alpha \leq 2(\|\eta_1\|_\alpha + \|\eta_2\|_\alpha)^\alpha$. ♢

Theorem **2**. *Let*

(12)

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

Then

$$\max_{1\leq K\leq n} \| \sum_{k=1}^{K} (W_{k,m,n} + \psi_{1,m,n} - X_k) \|_\alpha = \mathcal{O}(n^{1/\alpha+1}m^{\beta-1} + n^2 m^{\beta-2+1/\alpha}). \tag{13}$$

Proof of Theorem 2. 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 (13) follows from Lemmas 1, 2 and 3. ♢

Remark **1**. The inclusion of the term $\psi_{1,m,n}$ in (8) of Lemma 1 significantly improves the accuracy of the approximation. The intuition is clear since as $n/m \to 0$, $a_{k+j}/a_{1+j} \to 1$ *uniformly* over $k = 1, \ldots, n$ as $j \geq m + 1 - n \to \infty$ 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

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

The latter has the same order of magnitude as

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

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

$$n \|\psi_{1,m,n}\|_\alpha \leq 2^{1/\alpha} (\| \sum_{k=1}^{n} \psi_{k,m,n} - n\psi_{1,m,n} \|_\alpha + \| \sum_{k=1}^{n} \psi_{k,m,n} \|_\alpha)$$

due to (11). Clearly, $n^2 m^{\beta-2+1/\alpha} = o(nm^{\beta-1+1/\alpha})$ since $n/m \to 0$. Thus, the error bound is substantially improved by subtracting $\psi_{1,m,n}$ in (8).

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 Levy 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*

$$\sup_{0 \leq u \leq 1} \|n^{-H} S_n^*(u) - \xi_{H,\alpha}(u)\|_\alpha = \mathcal{O}[n^{-H} + (n/m)^{\min(2-H,\ 1-H+1/\alpha)}]. \tag{14}$$

Proof of Proposition 1. Observe that $\xi_{H,\alpha}(K/n) = n^{-H} \sum_{k=1}^{K} Y_{k,n}$. Define

$$\eta_1 = \sum_{k=1}^{K} (W_{k,m,n} + \psi_{1,m,n} - X_{k,n}) \text{ and } \eta_2 = \sum_{k=1}^{K} (X_{k,n} - Y_{k,n}).$$

Then by using relationship (11), and Theorems 1 and 2 we get

$$\max_{1 \leq K \leq n} \|n^{-H} S_n^*(K/n) - \xi_{H,\alpha}(K/n)\|_\alpha = \mathcal{O}[n^{-H} + (n/m)^{\min(2-H,\ 1-H+1/\alpha)}]$$

since $H = \beta + 1/\alpha$ and $n/m \to 0$. By the stationarity of the increments of the LFSM processes, we have that $\{\xi_{H,\alpha}(u) - \xi_{H,\alpha}(\lfloor nu \rfloor/n),\ k/n \leq u \leq (k+1)/n\} =_d \{\xi_{H,\alpha}(v),\ 0 \leq$

$v \leq 1/n\}$ for all $k = 0, 1, \ldots, n-1$. Therefore,

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

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

Remark **2**. In Lemma 2, we introduce additional dependence artifacts to ensure the circular representation (12). In our `FFT`-based algorithm (cf Section 2.4), such a circular structure is needed for a fast implementation. The additional dependence yields the error term $(n/m)^{1-H+1/\alpha}$ in (14). 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 $m = o[n^{(1+1/\alpha)/(1-H+1/\alpha)}]$. ♢

Samorodnitsky and Taqqu (1994, Proposition 3.5.1) showed that the stochastic integrals $\int_{\mathbb{R}} f_n(s) M_\alpha(ds)$ converge in probability to $\int_{\mathbb{R}} f(s) M_\alpha(ds)$ if and only if $\| \int_{\mathbb{R}} [f_n(s) - f(s)] M_\alpha(ds)\|_\alpha^\alpha = \int_{\mathbb{R}} |f_n(s) - f(s)|^\alpha ds \to 0$, 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 (14) 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

$$n^{-H} + (n/m)^{\min(2-H,\ 1-H+1/\alpha)} \leq \delta. \tag{15}$$

Let $p = p(\alpha, H) = \min(2 - H,\ 1 - H + 1/\alpha)$ and $L = m/n$ denote the over-sampling rate. The expression in equation (15) 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 over-sampling 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 $L \to \infty$ as $n \to \delta^{-1/H}$, whereas $n = (\delta - L^{-p})^{-1/H} \to \infty$ as $L \to \delta^{-1/p}$. Thus, these expressions capture the trade-off between L and n that is necessary for making the procedure computationally tractable. We elaborate next on this trade-off.

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

$$m(\delta; \alpha, H) = \{\delta^{-1}[(H/p)^{-H/(H+p)} + (H/p)^{p/(H+p)}]\}^{1/H+1/p}. \tag{16}$$

Thus, 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 obtain the following values for $m = 252779$ 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 2.4).

2.4. **The Algorithm.** Observe that $\{W_{k,m,n}, 1 \le k \le n\}$ and $\psi_{1,m,n}$ defined in (12) and (7) respectively are independent. Thus, by simulating $\{W_{k,m,n}, 1 \le k \le n\}$ 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

$$(\sum_{j=m+1-n}^{\infty} |a_{1+j}|^{\alpha})^{1/\alpha} = \frac{|\beta|}{[\alpha(1-H)]^{1/\alpha}} (m-n)^{H-1}[1 + \mathcal{O}(m^{-1})] \tag{17}$$

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

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

$$A_m = \begin{pmatrix} a_1 & a_2 & \ldots & a_{m-1} & a_m \\ a_2 & a_3 & \ldots & a_m & a_1 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ a_m & a_1 & \ldots & a_{m-2} & a_{m-1} \end{pmatrix}. \tag{18}$$

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

$$\begin{pmatrix} W_{1,m,n} \\ W_{2,m,n} \\ \ldots \\ W_{m-1,m,n} \\ W_{m,m,n} \end{pmatrix} = A_m \mathbf{e}_m^{\mathbf{T}}, \tag{19}$$

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

$$v_j = \sum_{t=1}^{m} u_t \exp[-\imath(t-1)\omega_{j-1}], \quad 1 \le j \le m. \tag{20}$$

The Inverse Discrete Fourier Transform $\mathbf{u}_m = IDFT(\mathbf{v}_m)$ satisfies

$$u_j = \frac{1}{m} \sum_{t=1}^{m} v_t \exp[\imath(t-1)\omega_{j-1}], \quad 1 \le j \le m. \tag{21}$$

Using the celebrated fast Fourier transforms (FFT) algorithm, one can obtain $DFT(\mathbf{u}_m)$ and $IDFT(\mathbf{v}_m)$ with time complexity $\mathcal{O}(m \log m)$ steps and $\mathcal{O}(m)$ memory complexity. Let $\mathbf{b}_m = DFT(\mathbf{a}_m) = (b_1, \ldots, b_m)$, $\mathbf{f}_m = DFT(e_1, e_m, e_{m-1}, \ldots, e_2) = (f_1, \ldots, f_m)$ and $\mathbf{p}_m = (b_1 f_1, \ldots, b_m f_m)$. Then due to the fact that the $m \times m$ matrix A is circulant [11], we have

$$A_m \mathbf{e}_m^{\mathbf{T}} = IDFT(\mathbf{p}_m). \tag{22}$$

Actually, by (21), the ℓth ($1 \le \ell \le m$) element in $IDFT(\mathbf{p}_m)$ is

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

where $q \bmod m \in \{0, 1, \ldots, m-1\}$ is the remainder of q when divided by m. Notice that $(k-1)\omega_{\ell-1} = (\ell-1)\omega_{k-1}$ and $\sum_{k=1}^{m} \exp[-\imath(t-1)\omega_{k-1} - \imath(t'-1)\omega_{k-1} + \imath(\ell-1)\omega_{k-1}]$ is nonzero if and only if $-(t-1) - (t'-1) + (\ell-1) \bmod m = 0$, in which case the sum is m. Hence, the ℓth 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 $1 \le t \le m$ and $1 \le t' \le m$ 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 $\mathbf{b}_m = DFT(\mathbf{a}_m) = (b_1, \ldots, b_m)$
Step 2: Generate a m-dimensional random vector $\mathbf{e}_m$ whose elements are iid SαS random variables employing the algorithm of Chambers et al. [8]
Step 3: Use the `FFT` to compute the DFT of $\mathbf{a}_m$. Let $\mathbf{f}_m = DFT(\mathbf{e}_m) = (f_1, \ldots, f_m)$
Step 4: Let $\mathbf{p}_m = (b_1 f_1, \ldots, b_m f_m)$
Step 5: Use the `FFT` to compute the IDFT of $\mathbf{p}_m$. Let $(W_1, \ldots, W_m) = IDFT(\mathbf{p}_m)$
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 SαS random variable independent of $(W_1, \ldots, W_m)$.

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 (14). 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

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

are identically distributed by appropriate permutations of ε_i, where ε_ℓ^* are iid and are independent of $\{\varepsilon_i\}_{i \in \mathbb{Z}}$. 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 $n \to \infty$,

$$\begin{aligned}\sup_{\lambda_1,\lambda_2\in\mathbb{R}} &|\mathbb{E}\exp[\imath(\lambda_1(W_1+c^*\varepsilon_1)+\lambda_2(W_n+c^*\varepsilon_2))] \\ &-\mathbb{E}\exp[\imath(\lambda_1(W_1+c^*\varepsilon_1))]\times\mathbb{E}\exp[\imath(\lambda_2(W_n+c^*\varepsilon_2))]| \\ \le \sup_{\lambda_1,\lambda_2} &|\mathbb{E}\exp[\imath(\lambda_1 W_1+\lambda_2 W_n)]-\mathbb{E}\exp(\imath\lambda_1 W_1)\times\mathbb{E}\exp(\imath\lambda_2 W_n)| \to 0.\end{aligned}$$

2.5. **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 $b_1, b_2 \in \mathbb{R}$. 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), (4) and (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 $Y_{k,n}' = n^H \int_{\mathbb{R}} b_1[(\frac{k}{n}-s)_+^\beta - (\frac{k-1}{n}-s)_+^\beta] + b_2[(\frac{k}{n}-s)_-^\beta - (\frac{k-1}{n}-s)_-^\beta] M_\alpha(ds)$. A similar version of Theorem 1 implies that $\max_{1\le K\le n} \| \sum_{i=1}^K (X_{k,n}' - Y_{k,n}')\|_\alpha = \mathcal{O}(1)$. As in Lemma 1, we 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 $\max_{1\le K\le n} \| \sum_{k=1}^K \psi_{k,m,n}' - K\psi_{1,m,n}'\| = \mathcal{O}(n^2 m^{\beta-2+1/\alpha})$ and $\max_{1\le K\le n} \| \sum_{k=1}^K \phi_{k,m,n}' - K\phi_{1,m,n}'\| = \mathcal{O}(n^2 m^{\beta-2+1/\alpha})$. As in Theorem 2, the "main part" $X_k - \psi_{k,m,n}' - \phi_{k,m,n}' = \sum_{j=n-m}^{m-n} a_{k+j}' \varepsilon_{-j,n}$, $1 \le k \le m$, 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 (19), we can write the vector $(W_{1,m,n}', \ldots, W_{m,m,n}')^{\mathbf{T}} = A_m' \mathbf{e}_{2m+1}^{\mathrm{T}}$, where A_m' is a circulant matrix generated by the vector $\mathbf{a}' = (a_{-m}', a_{-m+1}', \ldots, a_m')$ and $\mathbf{e}_{2m+1}$ is a $2m+1$-dimensional random vector with iid standard SαS random variables as elements. Thus, Algorithm 1 can similarly be applied to generate the random vector $(W_{1,m,n}', \ldots, W_{m,m,n}')$.

In summary, let $\{\varepsilon_i^{\#}\}_{i\in\mathbb{Z}}$, $\{\varepsilon_i\}_{i\in\mathbb{Z}}$ and $\{\varepsilon_i^*\}_{i\in\mathbb{Z}}$ be iid sequences, $c_1^* = c^* b_1$ and $c_2^* = c^* b_2$. Then the cumulative sum of $n^{-H}(b_2 c^* \varepsilon_1^{\#} + W_j + b_1 c^* \varepsilon_1^*)$, $1 \le j \le n$ converges in the sense of finite dimensional distributions to the LFSM $\{\xi_{H,\alpha}^{b_1,b_2}(t),\ 0 \le t \le 1\}$ with error bound given by (14).

2.6. **Matlab Code.** In this section we provide a `Matlab` program that generates L versions of LFSM. The `Matlab` program `salphas(alpha,nbp)` generates N iid standard SαS 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 1G Hz machine with 1.5GB of memory, the running times for `LFSM(.75,1.5,4000000,50000)` and `LFSM(.75,1.5,20000000,50000)` are about 30 and 300 seconds 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;
```

3. Comparison with a Stochastic Integral Approximation Procedure

3.1. **A Simple Approximation Algorithm.** By discretizing the stochastic integral representation of the stationary increment process $\xi_{H,\alpha}(j)-\xi_{H,\alpha}(j-1)$, $1 \le j \le T$, Samorodnitsky and Taqqu [31] proposed the following two-step approximation procedure to simulate LFSM (cf Chapter 7.11, pp 370-376). Let $K_\beta(x) = x_+^\beta - (x-1)_+^\beta$. Then the fractional stable noise can be written as

$$\begin{aligned} Y_j &= \int_{\mathbb{R}} K_\beta(j+1-x)M_\alpha(dx) \approx \sum_{k\in\mathbb{Z}} K_\beta(j+1-k/l)[M_\alpha((k+1)/l) - M_\alpha(k/l)] \\ &= \sum_{u\in\mathbb{Z}} K_\beta(u/l)[M_\alpha(j+1-u/l+1/l) - M_\alpha(j+1-u/l)], \end{aligned} \tag{23}$$

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 $1 \le u \le Ll$:

$$\tilde{Y}_j = \sum_{u=1}^{Ll} K_\beta(u/l)[M_\alpha(j+1-u/l+1/l) - M_\alpha(j+1-u/l)], \tag{24}$$

where L is the cutoff in the limits of integration, which defines the memory (cf Equation (7.11.1) in [31]). Notice that $\eta_{j-u/l} = l^\alpha[M(j+1-u/l+1/l) - M(j+1-u/l)]$, $1 \le j \le T$, $1 \le u \le Ll$ are $l(T+L-1)$ iid $S_\alpha(1,0,0)$ random variables. One can then generate $l(T+L-1)$ iid $S_\alpha(1,0,0)$ random variables and use the partial sum $\sum_{j=1}^{\lfloor Tt \rfloor} \tilde{Y}_j/T^H$, $0 \le t \le 1$ 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 (24), the two tails $\sum_{u\le 0} K_\beta(u/l)\varepsilon_{j-u/l}$ 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 $\tilde{Y}_j$, 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 Corollary 2.1 therein. 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 (14) and (15), one requires

$$l^{-p_1} + L^{-p_2} \leq \delta. \tag{25}$$

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 (16), we have the inequality $l^{-p_1} + (J/l)^{-p_2} \geq C(p_1,p_2)J^{-p_1p_2/(p_1+p_2)}$, which due to (25) implies that

$$J \geq [\delta^{-1}C(p_1,p_2)]^{1/p_1+1/p_2} \tag{26}$$

and the corresponding computational complexity is given by $\mathcal{O}(J\log J) = \mathcal{O}[(\delta^{-1})^{1/p_1+1/p_2}\log\delta^{-1}]$ as $\delta \to 0$. For the same level of accuracy $\delta > 0$, Proposition 1 and (16) 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 1 gives the values of m, J and $\rho = (m\log m)/(J\log J)$ for a few selected values of the Levy parameter $\alpha = .5,\ 1.5$, the approximation error $\delta = .1,\ .01$ and the Hurst index $H = .1,\ .3,\ .5,\ .7,\ .9$. The ratio ρ reflects the reduction in computational complexity. As shown by the entries of Table 1, the gains of our algorithm become more prominent for larger values of H and/or α is small.

[Insert Table 1 about here]

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 (14), but does not offer improvements in estimating H.

3.2. **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.

4. Proofs

Proof of Theorem 1. Let $f(s) = f_{K/n}(s) = (K/n - s)_+^{\beta} - (-s)_+^{\beta}$ and $a_i = 0$ for $i \leq 0$. Then $\sum_{k=1}^{K} Y_{k,n} = n^H \int_{\mathbb{R}} f(s) M_\alpha(ds)$ and $\sum_{k=1}^{K} X_{k,n} = \sum_{i\in\mathbb{Z}} (\sum_{k=1}^{K} a_{k-i})\varepsilon_{i,n}$. Let $g_i(s) = f(s) - n^{-\beta}\sum_{k=1}^{K} a_{k-i}$. Since $\varepsilon_{k,n} = n^{1/\alpha}\int_{(k-1)/n}^{k/n} M_\alpha(ds)$,

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

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

$$\sum_{i\in\mathbb{Z}} \int_{(i-1)/n}^{i/n} |g_i(s)|^\alpha ds = \sum_{i\leq K} \int_{(i-1)/n}^{i/n} |g_i(s)|^\alpha ds = \mathcal{O}(n^{-\alpha H}). \tag{27}$$

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

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

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

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

Now it remains to verify (27) 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

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

$\diamondsuit$

Acknowledgments

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 is provided by one referee.

References

[1] Abry, P. and Sellan, F. (1996), The wavelet based synthesis for fractional Brownian motion, *Applied and Computational Harmonic Analysis*, 3, 377-383

[2] Abry, P. and Veitch, V. (1998), Wavelet analysis of long-range dependent traffic, *IEEE Transaction on Information Theory*, 44, 2-15

[3] Avram, F. and Taqqu, M.S. (1986), Weak convergence of moving averages with infinite variance, in *Dependence in Probability and Statistics: A Survey of Recent Results*, Eberlein and Taqqu (eds.), 399-416, Boston: Birkhauser

[4] Avram, F. and Taqqu, M.S. (1992), Weak Convergence of Sums of Moving Averages in the α-Stable Domain of Attraction, *Annals of Probability*, 20, 483-503

[5] Astrauskas, A. (1983), Limit theorems for sums of linearly generated random variables, *Lithuanian Mathematics Journal*, 23, 123-134

[6] Beran, J. (1994), *Statistics for Long Memory Processes*, London: Chapman & Hall

[7] Coeurjolly, J.F. (2000), Simulation and identification of the fractional Brownian motion: a bibliographical and comparative study, *Journal of Statistical Software*, 5, (http://www.jstatsoft.org)

[8] Chambers, J.M., Mallows, C.L. and Stuck, B. W. (1976), A method for simulating stable random variables, *Journal of the American Statistical Association*, 71, 340-344

[9] Chambers, J.M., Mallows, C.L. and Stuck, B. W. (1987), Correction to: "A method for simulating stable random variables", *Journal of the American Statistical Association*, 82, 704

[10] Davydov, Y.A. (1970), The invariance principle for stationary processes, *Theory of Probability and its Applications*, 15, 487-498

[11] Dietrich, C.R. and Newsam, G.N. (1997), Fast and exact simulation of stationary Gaussian processes through circulant embedding of the covariance matrix, *SIAM Journal on Scientific Computing*, 18, 1088-1107

[12] Flandrin, P. (1992), Wavelet analysis and synthesis of fractional Brownian motion, *IEEE Transaction on Information Theory*, 38, 910-917

[13] Gallardo, J.R., Makrakis, D., Orozco-Bardosa, L. (2000), Use of α-stable self-similar stochastic processes for modeling traffic in broadband networks, *Performance Evaluation*, 40, 71-98

[14] Giordano, S., Porcarelli, S., Procissi, G. (2000), Testing alpha-stable processes in modeling broadband teletraffic, *Proceedings of the ICC'00*, 18-22, New Orleans

[15] Harmantzis, F.C., Hatzinakos, C., Katzela, I. (2001), Tail probabilities for the multiplexing of fractional α-stable broadband traffic, *Proceedings of the ICC'01*, Helsinki

[16] Istas, J. and Lang, G. (1997), Variations quadratiques et estimation de l'exposant de Holder local d'un processus Gaussien, *Annales de l' Institut Henri Poincare*, 33, 407-436

[17] Karasaridis, A., Hatzinakos, D. (2001), Network traffic modeling using α-stable self-similar processes, *IEEE Transactions on Communications*, 49, 1203-1214

[18] Kent, J.T., Wood, A. (1997), Estimating the fractal dimension of a locally self-similar Gaussian process using increments, *Journal of the Royal Statistical Society*, B, 59, 679-700

[19] Kolmogorov, A. N. (1940), Winersche spiralen und einige andere interessante kurven in Hilbertschen raum, S.R. Academic Science, USSR, 26, 115-118

[20] Konstantopoulos, T., Lin, S.J. (1998), Macroscopic models for long-range dependent network traffic, *Queueing Systems. Theory and Applications*, 28, 215-243

[21] Leland, W., Taqqu, M., Willinger, W. and Wilson, D. (1994), On the self-similar nature of Ethernet traffic, *ACM/IEEE Transactions on Networking*, 2, 1-15

[22] Mandelbrot, B., Van Ness, W.J. (1969), Fractional Brownian motions, fractional noises and applications, *SIAM Review*, 10, 422-437

[23] Mikosch, T., Resnick, S., Rootzén, H. and Stegeman, A. (2002), 'Is network traffic approximated by stable Levy motion or fractional Brownian motion?', *Annals of Applied Probability*, 12, 23-68

[24] Norros, I. (1995), On the use of fractional Brownian motion in the theory of connectionless networks, *IEEE Journal on Selected Areas in Communications*, 13, 953-962

[25] Norros, I., Mannersalo, P. and Wang, J.L. (1999), Simulation of fractional Brownian motion with conditionalized midpoint displacement, *Advances in Performance Analysis*, 2, 77-101

[26] Park, K., Willinger, W. (eds) (2000), *Self-Similar Network Traffic and Performance Evaluation*, Wiley

[27] Paxson, V., Floyd, S. (1995), Wide-area traffic: the failure of Poisson modeling, *IEEE/ACM Transactions on Networking*, 3, 226-244

[28] Paxson, V. (1997), Fast, approximate synthesis of fractional Gaussian noise for generating self-similar network traffic, *Computer Communication Review*, 27, 5-18

[29] Stoev, S., Pipiras, V. and Taqqu, M. (2002), Estimation of the self-similarity parameter in linear fractional stable motion, *Signal Processing* 82, 1873-1901

[30] Stoev, S. and Taqqu, M. (2003), Simulation methods for linear fractional stable motion and FARIMA using the Fast Fourier Transform, *Fractals*, To appear.

[31] Samorodnitsky, G. and Taqqu, M. (1994), *Stable Non-Gaussian Random Processes*, London: Chapman & Hall

[32] Taqqu, M., Teverovsky, V. and Willinger, W. (1997), Is network traffic self-similar of multifractal?, *Fractals*, 5, 63-73

[33] Taqqu, M., Willinger, W. and Sherman, R. (1997), Proof of a fundamental result in self-similar traffic modeling, *Computer Communication Review*, 27, 5-23

[34] Yuksel, M, Sikdar, B, Vastola, K.S. and Szymanski, B. (2000), Workload generation for ns simulations of wide area networks and the Internet, *Proceedings of Communication Networks and Distributed Systems Modeling and Simulation Conference*, 93-98, San Diego

[35] Willinger, W., Taqqu, M, Leland, W. and Wilson, D. (1995), Self-similarity in high-speed packet traffic: analysis and modeling of Ethernet traffic measurements, *Statistical Science*, 10, 67-85

[36] Wood, A. and Chan, G. (1994), Simulation of stationary Gaussian processes in $[0,1]^d$, *Journal of Computational and Graphical Statistics*, 3, 409-432

		$\alpha = .5$			$\alpha = 1.5$		
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}

TABLE 1. The values of m, J and $\rho = (m \log m)/(J \log J)$ are given by (16) and (26), for the Levy parameter $\alpha = .5,\ 1.5$, the approximation error $\delta = .1,\ .01$ and the Hurst index $H = .1,\ .3,\ .5,\ .7,\ .9$.

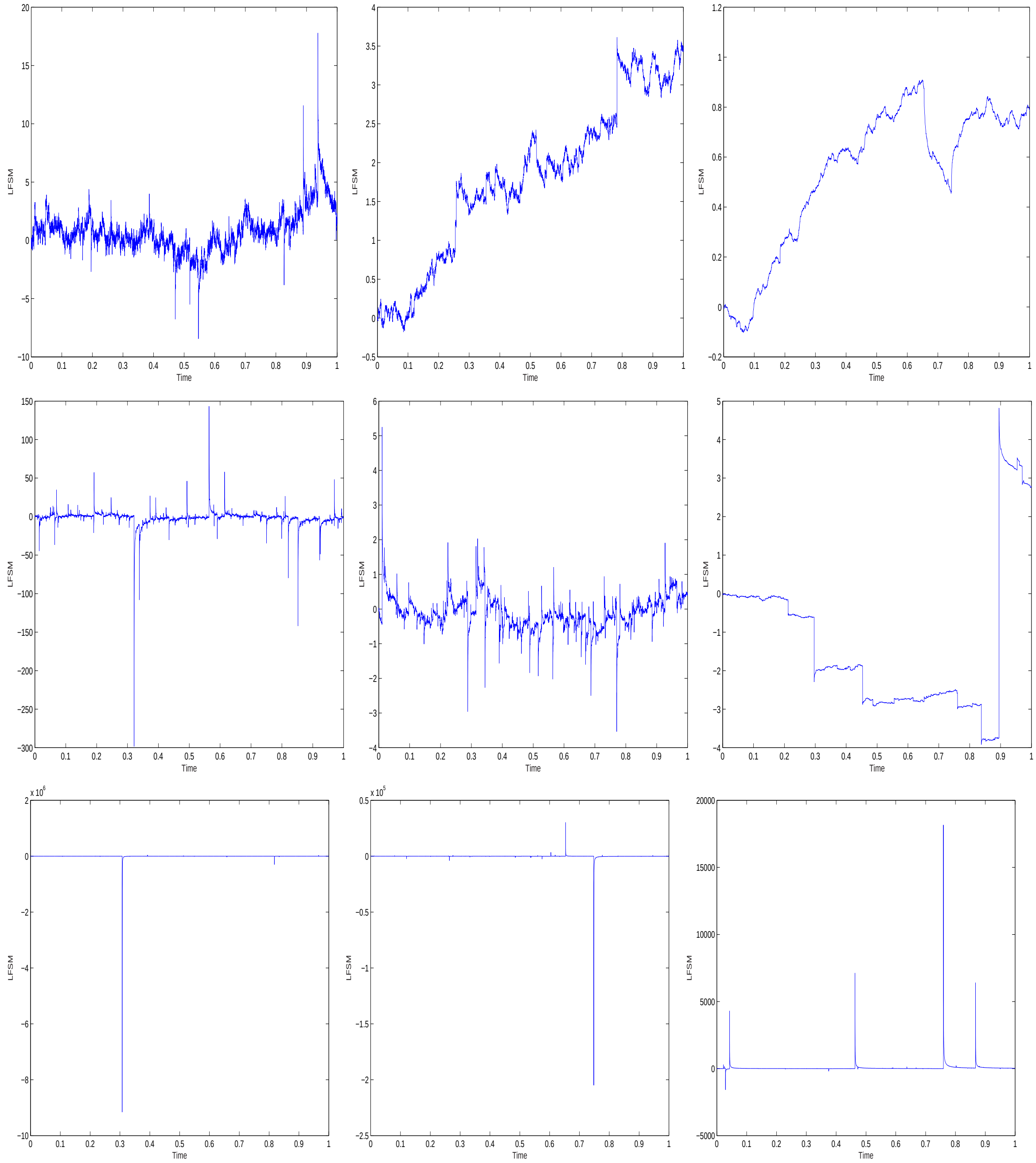

FIGURE 1. Top, middle and bottom panels: realizations of linear fractional stable motions for $\alpha = 1.8$, $\alpha = 1.2$ and $\alpha = 0.6$. In all cases, the left panel corresponds to $H = .2$, the middle panel to $H = .5$ and the right panel to $H = .8$. The $x-$axis represents time ($t = k/n,\ k = 0, 1, 2, \cdots, n$), while on the $y-$axis the values of the LFSM process are given.