To appear in the journal "Fractals" in 2003 or 2004

Simulation methods for linear fractional stable motion and FARIMA using the Fast Fourier Transform *†‡

Stilian Stoev and Murad S. Taqqu
Boston University

July 9, 2003

Contents

Abstract

We present efficient methods for simulation, using the Fast Fourier Transform (FFT) algorithm, of two classes of processes with symmetric α-stable (SαS) distributions. Namely, *(i)* the *linear fractional stable motion* (LFSM) process and *(ii)* the *fractional autoregressive moving average* (FARIMA) time series with SαS *innovations*. These two types of *heavy-tailed* processes have infinite variances and *long-range dependence* and they can be used in modeling the traffic of modern computer telecommunication networks.

We generate paths of the LFSM process by using Riemann-sum approximations of its SαS stochastic integral representation and paths of the FARIMA time series by truncating their moving average representation. In both the LFSM and FARIMA cases, we compute the involved sums *efficiently* by using the Fast Fourier Transform algorithm and provide bounds and/or estimates of the approximation error.

[*] This research was partially supported by the NSF Grant DMS-0102410 at Boston University.

[†] *AMS Subject classification.* Primary 60G18, 60E07 ; secondary 62M10, 65C05.

[‡] *Keywords and phrases:* simulation, linear fractional stable motion, self-similarity, Hurst parameter, FARIMA time series, heavy tails, wavelets, Mallat's algorithm.

We discuss different choices of the discretization and truncation parameters involved in our algorithms and illustrate our method. We include MATLAB implementations of these simulation algorithms and indicate how the practitioner can use them.

1 Introduction

Modern computer telecommunication networks present numerous challenges for researchers from the computer science, engineering and applied mathematics communities. One of the fundamental problems in the interdisciplinary field of Computer Networks is to understand the nature of the information traffic on networks with various types of links and topologies.

Leland, Taqqu, Willinger and Wilson (1994) have shown that that the fluctuations in the Ethernet traffic of a Local Area Network do *not* obey the classical Poisson-type stochastic models. More precisely, by performing an extensive statistical analysis, these authors established that the Ethernet traffic is statistically *self-similar* (or *asymptotically self-similar*) and exhibits *long-range dependence.* Other authors have confirmed these observations in various other networking contexts, for example, Paxson and Floyd (1994) and Crovella and Bestavros (1996) (see also the collection of papers edited by Park and Willinger (2000), for further references).

The continuous-time stochastic process $X = \{X(t),\ t \in \mathbb{R}\}$ is said to be self-similar with self-similarity parameter $H > 0$, if for all $c > 0$,

$$\{X(ct),\ t \in \mathbb{R}\} =_d \{c^H X(t),\ t \in \mathbb{R}\}, \tag{1.1}$$

where $=_d$ denotes equality in finite-dimensional distributions. On the other hand, a stationary and finite variance time series $Y = \{Y(k),\ k \in \mathbb{Z}\}$ is said to be *long-range dependent* when its covariances $r(k) = \mathrm{Cov}(X(k), X(0))$ decay slowly, as $k \to \infty$, so that

$$\sum_{k=0}^{\infty} |r(k)| = \infty. \tag{1.2}$$

The *slow* decay of the covariances means that the values of the process X are strongly dependent over long periods of time.

The *fractional Brownian motion* process (fBm) $B_H = \{B_H(t),\ t \in \mathbb{R}\}$ is a well-known example of a Gaussian self-similar stochastic process, which has stationary increments. In fact, up to a multiplicative constant the fBm is the only Gaussian processes with these two properties. It has the following covariance structure

$$\mathrm{Cov}(B_H(s), B_H(t)) = \frac{\sigma^2}{2}\Big(|t|^{2H} + |s|^{2H} - |t-s|^{2H}\Big),\ t, s \in \mathbb{R},$$

where $\mathrm{Var}(B_H(1)) = \sigma^2 \in (0, \infty)$ and $H \in (0, 1)$ denotes the self-similarity parameter of B_H. The last relation implies that the increment process $Y(k) := B_H(k) - B_H(k-1),\ k \in \mathbb{Z}$ of fBm is long-range dependent when the self-similarity parameter H is in the range $(1/2, 1)$. For more details on the long-range dependence phenomenon and the fractional Brownian motion process see, for example, the collection of papers edited by Doukhan, Oppenheim and Taqqu (2003).

The fBm process has been very useful in the context of network traffic modeling not only because it reproduces the observed statistical behavior. In fact, the fBm process appears as the limiting process in a physical on-off type model, where traffic is generated by many users, sharing a single *fast* link (see, for example, Taqqu, Willinger and Sherman (1997)). In other networking scenarios however, when for example there are few users with very irregular pattern of behavior and unlimited access to link bandwidth one observes *large peaks* of traffic activity. These *peaks* or *bursts* in the traffic cannot be modeled by Gaussian or finite-variance processes. Heavy-tailed processes with possibly infinite variance distributions are more natural and appropriate models in such scenarios (see for example, Willinger, Paxson and Taqqu (1998), Mikosch, Resnick, Rootzen and Stegeman (2002) and also Levy and Taqqu (2000) and Pipiras, Taqqu and Levy (2000)).

Large traffic fluctuations have serious impact on system performance and they should be taken into account when designing and managing computer networks. An important problem in this context is to estimate the amount of excess bandwidth required to accommodate bursty traffic. One of the approaches in investigating performance and quality of service issues is to simulate stochastic processes which reproduce well the nature of the traffic and then use them as performance benchmarks.

Our goal, in this paper, is to provide practical tools for simulating α-stable ($0 < \alpha < 2$), infinite variance processes, which exhibit asymptotic self-similarity and/or long-range dependence. The definition of long-range dependence given in (1.2) is formally incorrect in the case of infinite variance processes, because the covariances do not exist. However, many authors have studied processes, which are strongly dependent in time and have infinite variances (see, for example, the collection of papers edited by Adler, Feldman and Taqqu (1998)). Here we will focus on two classical models: *(i)* the linear fractional stable motion (LFSM) and *(ii)* the fractional autoregressive moving average (FARIMA) time series with symmetric α-stable (SαS) innovations. The linear fractional stable motion is a self-similar process with stationary increments and SαS distributions. It can be viewed as an extension of the fractional Brownian motion to the stable infinite-variance case, because it has an analogous stochastic integral representation (see (2.1) below or Ch. 7 in Samorodnitsky and Taqqu (1994), for more details). The FARIMA time series with SαS innovations and positive values of the fractional differencing exponent are asymptotically self-similar and exhibit long-range dependence. The FARIMA time series models are also flexible, because they can represent a wide variety of short-term dependence structures (for more details and further references see Sections 2.2 and 3.2 below).

The ideas behind the methods of simulation, presented in this work, exploit well-known representations of the LFSM and FARIMA processes. Our algorithms, however, utilize the Fast Fourier Transform (FFT) algorithm and prove to be computationally efficient in practice.

The paper is organized as follows. In Sections 2.1 and 2.2, we describe methods for simulation of the LFSM process and the FARIMA time series with SαS innovations, respectively. We also give estimates on the errors of approximation involved in the simulation algorithms. Sections 3.1 and 3.2 are devoted to a numerical study of the methods presented in Sections 2.1 and 2.2, respectively. Our main focus there is on the practical choice of the discretization parameters involved in the methods of simulation. Section 4 contains MATLAB implementations of our algorithms and indications on how to use them.

2 Simulation Methods

2.1 The case of linear fractional stable motion (LFSM)

Consider the stochastic process $X_{H,\alpha} = \{X_{H,\alpha}(t),\ t \in \mathbb{R}\}$, which is defined by the following stochastic integral

$$X_{H,\alpha}(t) = C_{H,\alpha}^{-1} \int_{\mathbb{R}} \Big((t-s)_+^{H-1/\alpha} - (-s)_+^{H-1/\alpha} \Big) dL_\alpha(s), \tag{2.1}$$

where $0 < H < 1,\ \alpha \in (0,2)$ and where $L_\alpha = \{L_\alpha(s),\ s \in \mathbb{R}\}$ is a standard symmetric α-stable (SαS) Lévy motion. The stochastic integral in (2.1) is well-defined since for all $H \in (0,1)$ and $\alpha \in (0,2)$, $\int_{\mathbb{R}} |(t-s)_+^{H-1/\alpha} - (-s)_+^{H-1/\alpha}|^\alpha ds < \infty$.

The process $X_{H,\alpha}$ is called a *linear fractional stable motion* (LFSM, in short). It is self-similar with self-similarity parameter H, that is, it satisfies Relation (1.1). The process $X_{H,\alpha}$ has also stationary increments. The normalization constant $C_{H,\alpha}$ is such that $\|X_{H,\alpha}(1)\|_\alpha = 1$, more precisely

$$C_{H,\alpha} = \Big(\int_{\mathbb{R}} |(1-s)_+^{H-1/\alpha} - (-s)_+^{H-1/\alpha}|^\alpha ds \Big)^{1/\alpha}. \tag{2.2}$$

We denote by $\|\xi\|_\alpha$ the scale coefficient of a SαS random variable ξ, that is, the coefficient which appears in the characteristic function of ξ,

$$\varphi_\xi(\lambda) = \mathbb{E} e^{i\xi\lambda} = e^{-\|\xi\|_\alpha^\alpha |\lambda|^\alpha},\ \lambda \in \mathbb{R}.$$

For more details on the class of LFSM and other α-stable stochastic integrals see, for example, the book of Samorodnitsky and Taqqu (1994). The parameter $\alpha \in (0,2)$ controls the tails of the distribution of a SαS random variable ξ, that is,

$$\mathbb{P}(|\xi| \geq x) \sim \text{const}\, x^{-\alpha}, \quad \text{as } x \to \infty, \tag{2.3}$$

where $a_n \sim b_n,\ n \to \infty$ means $\lim_{n\to\infty} a_n/b_n = 1$. The greater the value of α, the lower the probability of extreme fluctuations of the SαS process $X_{H,\alpha}$.

In the Gaussian case $\alpha = 2$, L_α becomes Brownian motion and then the LFSM process $X_{H,\alpha}$ is the fractional Brownian motion process. Here, however, we suppose $0 < \alpha < 2$. In this case, (2.3) holds and consequently L_α has infinite variance, and when $\alpha \leq 1$, infinite mean as well. This implies in practice that L_α and hence $X_{H,\alpha}$ can, with high probability, take extremely high values. The simulation method in this section works also when $\alpha = 2$. In the Gaussian case, however, there exist more efficient and, in fact, exact methods (see, for example, Bardet *et al* (2003)). In the stable infinite variance case exact simulation techniques for the LFSM process $X_{H,\alpha}$ are not known.

We now present an approximate simulation method for paths $X_{H,\alpha}(k),\ k = 1,\ldots,N,\ N \in \mathbb{N}$ of the LFSM process $X_{H,\alpha}$, defined in (2.1). Observe that by self-similarity, we have

$$\Big(X_{H,\alpha}(1), X_{H,\alpha}(2), \ldots, X_{H,\alpha}(N) \Big) =_d N^H \Big(X_{H,\alpha}(1/N), X_{H,\alpha}(2/N), \ldots, X_{H,\alpha}(1) \Big).$$

The idea of the method is to use a Riemann sum approximation of the stochastic integral in the representation (2.1) (see e.g. p. 370 in Samorodnitsky and Taqqu (1994)). In practice, however, it is essential to compute these Riemann sums efficiently. Indeed, a direct implementation yields a prohibitively slow algorithm and it cannot be used in practice to generate long and/or accurate approximations in real time. Our method uses the fast Fourier transform algorithm (FFT, in short), which improves significantly the speed of computation as compared to a direct summation.

Let $Y(k) := X_{H,\alpha}(k) - X_{H,\alpha}(k-1),\ k \in \mathbb{Z}$ be the sequence of the increments of the LFSM process $X_{H,\alpha}$. By (2.1), we have that

$$Y(k) = C_{H,\alpha}^{-1} \int_{\mathbb{R}} g(k-s) dL_\alpha(s) = -C_{H,\alpha}^{-1} \int_{\mathbb{R}} g(s) dL_\alpha(k-s), \tag{2.4}$$

where $g(s) = (s)_+^{H-1/\alpha} - (s-1)_+^{H-1/\alpha}$. The stationary sequence $Y(k),\ k \in \mathbb{Z}$ is called *linear fractional stable noise.*

We will first generate an approximate path of the stationary linear fractional stable noise process $Y = \{Y(k),\ k \in \mathbb{Z}\}$ by replacing the integral in (2.4) with a Riemann sum. Then, by using the fact that $X_{H,\alpha}(k) = \sum_{j=1}^{k} Y(j),\ k = 1, \ldots, N$ we will obtain an approximate path of the LFSM process $X_{H,\alpha}$.

Introduce the parameters $m, M \in \mathbb{N}$ and let

$$Y_{m,M}(k) := C_{H,\alpha}^{-1}(m,M) \sum_{j=1}^{mM} \Big((j/m)_+^{H-1/\alpha} - (j/m-1)_+^{H-1/\alpha} \Big) Z_{\alpha,m}(mk-j), \tag{2.5}$$

where $Z_{\alpha,m}(j) := L_\alpha((j+1)/m) - L_\alpha(j/m),\ j \in \mathbb{Z}$ and where the normalization constant $C_{H,\alpha}(m,M)$ is such that $\|Y_{m,M}(1)\|_\alpha = 1$ (see (2.7) below for an exact expression). The random variables $Z_{\alpha,m}(j),\ j \in \mathbb{Z}$ are independent with the same SαS distribution and scale parameter $\|Z_{\alpha,m}(j)\|_\alpha = m^{-1/\alpha}$. The parameter m controls the mesh size and the parameter M the kernel function cut-off. For large values of the parameters m and M, the variables $Y_{m,M}(k)$ approximate well the variables $Y(k),\ k \in \mathbb{Z}$ in probability. A bound on the error of approximation $\|Y(k) - Y_{m,M}(k)\|_\alpha$ in terms of the parameters $m,\ M,\ H$ and α is given in Theorem 2.1 below.

We will now describe a method for *efficient* computation of the path $Y_{m,M}(k),\ k = 1, 2, \ldots, N$, which involves the FFT algorithm. Consider the moving average $W(n),\ n \in \mathbb{N}$,

$$W(n) := \sum_{j=1}^{mM} a_{H,m}(j) Z_\alpha(n-j), \tag{2.6}$$

where

$$a_{H,m}(j) := C_{H,\alpha}^{-1}(m,M) \Big((j/m)^{H-1/\alpha} - (j/m-1)_+^{H-1/\alpha} \Big) m^{-1/\alpha},\ j \in \mathbb{N},$$

and where $Z_\alpha(j),\ j \in \mathbb{Z}$ is a sequence of independent identically distributed (iid) standard SαS random variables, that is $\|Z_\alpha(1)\|_\alpha = 1$. Since $\{Z_{\alpha,m}(j),\ j \in \mathbb{Z}\} =_d \{m^{-1/\alpha} Z_\alpha(j),\ j \in \mathbb{Z}\}$, Relations (2.5) and (2.6) imply

$$\{Y_{m,M}(k),\ k = 1, \ldots, N\} =_d \{W(mk),\ k = 1, \ldots, N\},$$

and

$$C_{H,\alpha}(m,M) = m^{-1} \Big(\sum_{j=1}^{mM} \Big| (j/m)^{H-1/\alpha} - (j/m-1)_+^{H-1/\alpha} \Big|^\alpha \Big)^{1/\alpha}. \tag{2.7}$$

We shall then focus on computing the moving average sequence $W(n),\ n = 1,\dots,mN$. Let $\widetilde{Z}_\alpha(j),\ j \in \mathbb{Z}$ be $m(M+N)$-periodic with

$$\widetilde{Z}_\alpha(j) := Z_\alpha(j), \quad \text{for } j = 1,2,\dots,m(M+N),$$

and let

$$\widetilde{a}_{H,m}(j) := \begin{cases} a_{H,m}(j) & , \text{ for } j = 1,\dots,mM \\ 0 & , \text{ for } j = mM+1,\dots,m(M+N) \end{cases} .$$

Then

$$\Big\{W(n)\Big\}_{n=1}^{mN} =_d \Big\{ \sum_{j=1}^{m(M+N)} \widetilde{a}_{H,m}(j)\widetilde{Z}_\alpha(n-j)\Big\}_{n=1}^{mN}, \tag{2.8}$$

because for all $n = 1,2,\dots,mN$, the sum in (2.6) involves only variables $Z_\alpha(j)$, with indices j in the range $-mM \le j \le mN-1$. Relation (2.8) suggests that the variables $W(n),\ n = 1,2,\dots,mN$ can be generated by using a (circular) convolution of the two $m(M+N)$-periodic sequences $\widetilde{a}_{H,m}$ and $\widetilde{Z}_\alpha$, where $\widetilde{a}_{H,m}$ denotes the $m(M+N)$-periodic extension of the sequence $\widetilde{a}_{H,m}(j),\ j = 1,2,\dots,m(M+N)$.

The circular convolution of the $m(M+N)$-periodic sequences $\widetilde{a}_{H,m}$ and $\widetilde{Z}_\alpha$ can be computed by using their Discrete Fourier Transforms (DFT). For convenience, we recall some standard definitions and results concerning the DFT of periodic sequences.

Let $a = (a(j),\ j \in \mathbb{Z})$ be a R-periodic sequence, that is, $a(j+R) = a(j)$, for all $j \in \mathbb{Z}$. The DFT, $\mathcal{D}_R(a) = (\mathcal{D}_R(a)(k),\ k \in \mathbb{Z})$, of the sequence a is defined as follows

$$\widehat{a}(k) = \mathcal{D}_R(a)(k) = \sum_{j=0}^{R-1} e^{2\pi ijk/R} a(j),\ k \in \mathbb{Z}.$$

Thus the sequence $\widehat{a}$ is R-periodic and it satisfies the *inversion formula*

$$a(j) = \mathcal{D}_R^{-1}(\widehat{a})(j) := \frac{1}{R}\sum_{k=0}^{R-1} e^{-2\pi ijk/R}\widehat{a}(k),\ j \in \mathbb{Z}.$$

Furthermore, for any two R-periodic sequences a and b, we have the following *convolution theorem*

$$\mathcal{D}_R(a)(k)\mathcal{D}_R(b)(k) = \mathcal{D}_R(a*b)(k),\ k \in \mathbb{Z},$$

where

$$(a*b)(n) := \sum_{j=0}^{R-1} a(n-j)b(j),\ n \in \mathbb{Z}.$$

If R is an integer power of two, then the DFT of the R-periodic sequence a can be computed *efficiently* by using the FFT (radix-2) algorithm.

The above discussion leads to the following algorithm.

Algorithm

1. Pick *large enough* integers m and M, so that $m(M+N)$ is an integer power of two.

2. By using the radix-2 FFT algorithm, compute the DFT

$$\widehat{a}(k) := \mathcal{D}_{m(M+N)}(\widetilde{a}_{H,m})(k),\ k = 0, \dots, m(M+N) - 1$$

of the $m(M+N)-$periodic sequence $\widetilde{a}_{H,m}$.

3. Generate $m(M+N)$ iid standard SαS variates $Z(j),\ j = 1, \dots, m(M+N)$ and, by using the radix-2 FFT algorithm, compute

$$\widehat{Z}(k) := \mathcal{D}_{m(M+N)}(Z)(k),\ k = 0, \dots, m(M+N) - 1.$$

4. Using the radix-2 FFT algorithm, compute the inverse DFT of the sequence $\widehat{a}(k)\widehat{Z}(k),\ k = 0, \dots, m(N+M) - 1$, and keep only

$$W(n+1) := \mathcal{D}^{-1}_{m(M+N)}(\widehat{a}\widehat{Z})(n), \quad \text{where } n = 0, \dots, mN - 1,$$

and where $(\widehat{a}\widehat{Z})(k) = \widehat{a}(k)\widehat{Z}(k),\ k = 0, 1, \dots, m(M+N) - 1$.

5. Set

$$Y_{m,M}(k) = W(mk),\ k = 1, \dots, N \tag{2.9}$$

and let

$$X(n) := \sum_{k=1}^{n} Y_{m,M}(k),\ n = 1, \dots, N. \tag{2.10}$$

$Y_{m,M}(k),\ k = 1, \dots, N$ is the approximation of the linear fractional stable noise process Y and the sequence $X(n),\ n = 1, \dots, N$ is the desired approximate path of the linear fractional stable motion process $X_{H,\alpha}$.

In Section 3.1, we present numerical illustrations of the simulation algorithm. MATLAB implementation of this algorithm can be found in Section 4.

Figure 1 displays simulated paths of LFSM obtained by using this code. The path behavior of the LFSM process $X_{H,\alpha}$ depends on the structure of the kernel function $f_{H,\alpha}(t,s) = (t - s)_+^{H-1/\alpha} - (-s)_+^{H-1/\alpha},\ t, s \in \mathbb{R}$ in (2.1). In Figure 1, five of the six graphs correspond to the case $H - 1/\alpha < 0$. When $H - 1/\alpha < 0$ the function $f_{H,\alpha}(t,s),\ s \in \mathbb{R}$ has singularities at $s = 0$ and $s = t$. These singularities magnify the stable noise process $dL_\alpha(s),\ s \in \mathbb{R}$ and cause the large spikes in the paths of the LFSM process. These paths are very irregular and in fact discontinuous with probability one (see Ch. 10 in Samorodnitsky and Taqqu (1994), for example). Their dependence structure, however, resembles that of a *negatively correlated* process and thus we shall refer to this case as to the *negative dependence* scenario.

In the case when $H - 1/\alpha > 0$, (since $0 < H < 1$ it necessarily follows that $\alpha > 1$) the kernel $f_{H,\alpha}(t,s),\ s \in \mathbb{R}$ is *bounded* and positive, for all $t > 0$. Thus the jumps in the paths of $X_{H,\alpha}$, due to the fluctuations of the noise process $dL_\alpha(s),\ s \in \mathbb{R}$, are not as magnified as in the case $H - 1/\alpha < 0$. In this case especially for large values of H, the kernel function $f_{H,\alpha}(t,s),\ s \in \mathbb{R}$ decays *slowly* as $s \to -\infty$. This implies that the past fluctuations in the process $dL_\alpha(s),\ s \in \mathbb{R}$ influence significantly the present values of the process $X_{H,\alpha}$. This case is referred to as the *long-range dependence* scenario (see Figure 1). The paths of the process $X_{H,\alpha}$ can be made *continuous* with probability one (see Ch. 10 in Samorodnitsky and Taqqu (1994), for example).

The following result, gives a upper bound on the error of approximation involved in the above algorithm, when applied to linear fractional stable noise.

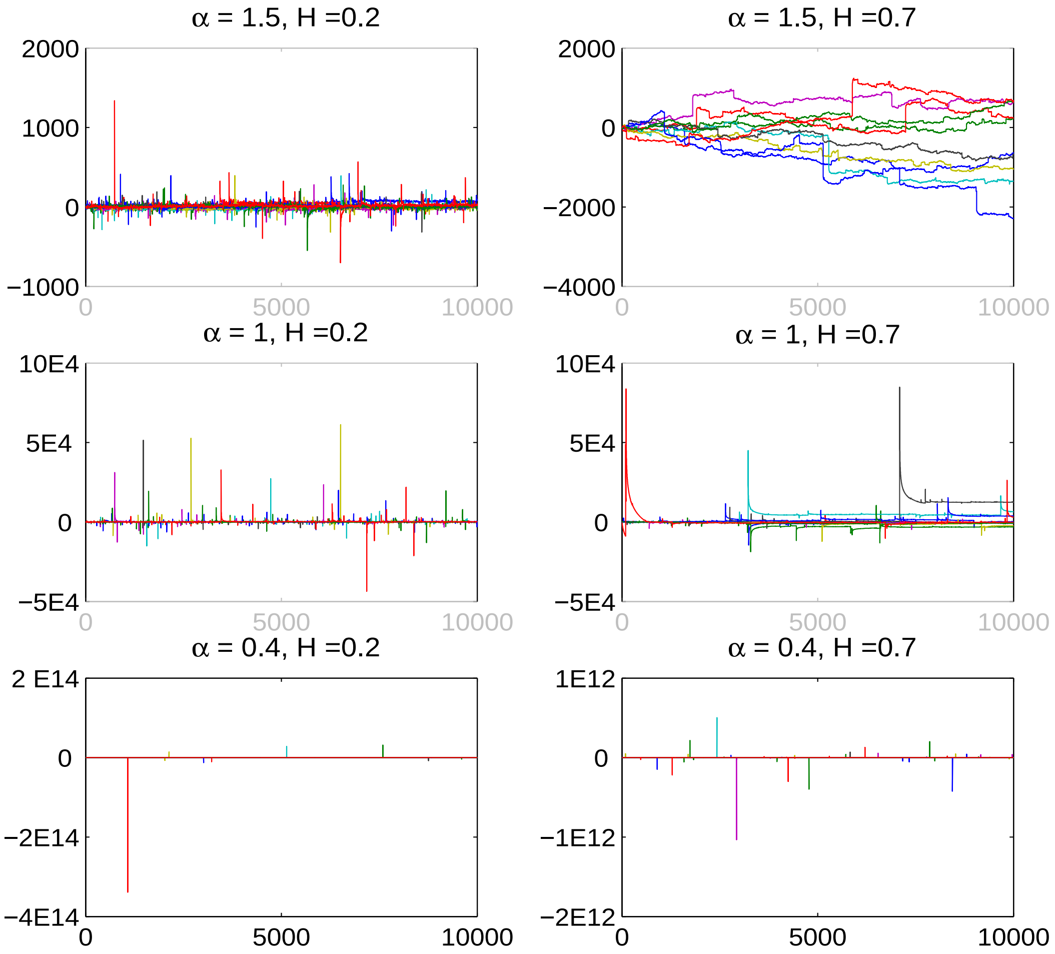

Figure 1: Each graph displays ten independent simulated paths of LFSM with $N+M = 16\,384 = 2^{14}$, and discretization parameters $m = 256 = 2^8$, $M = 600$ (see Section 2.1), where the symbol $a\text{E}b$ denotes the number $a \times 10^b$. The vertical scale is different from one graph to the other. Observe the difference between "high" and "low" values of α, that is, "heavy" and "very heavy" tails, respectively. (For "very low" values of α there are a number of fluctuations whose size is much greater than in rest of the path.) Also note the difference between the *negative dependence* $(H < 1/\alpha)$ and *long-range dependence* $(H > 1/\alpha)$ cases. All graphs but the one on the top right involve negative dependence scenarios. Because of the nature of the kernel of LFSM, in the cases of negative dependence, the jumps are quickly balanced by jumps with opposite signs, which results in visual spikes on the graphs. In the case of long-range dependence ($\alpha = 1.5,\ H = 0.7$), the paths are more dependent on the past and no significant spikes are observed.

Theorem 2.1 *Let Y be the linear fractional stable noise process and $Y_{m,M}$ its approximation, defined by* (2.4) *and* (2.5)*, respectively. For all $m, M \in \mathbb{N}$, $M \geq 2$ and $k \in \mathbb{Z}$, we have*

$$\|Y(k) - Y_{m,M}(k)\|_\alpha \leq J_{H,\alpha}^{1/\alpha}(m, M). \tag{2.11}$$

The bound $J_{H,\alpha}$ is as follows. For $1 \leq \alpha \leq 2$,

$$J_{H,\alpha}(m, M) = \frac{2^\alpha}{C_{H,\alpha}^\alpha(m, M)} I_{H,\alpha}(m, M), \tag{2.12}$$

and for $0 < \alpha < 1$,

$$J_{H,\alpha}(m, M) = \frac{1}{C_{H,\alpha}^\alpha(m, M)} I_{H,\alpha}(m, M) + \frac{(C_{H,\alpha} + C_{H,\alpha}(m, M))^{(1-\alpha)\alpha}}{\alpha^\alpha C_{H,\alpha}^\alpha(m, M)} I_{H,\alpha}^\alpha(m, M), \tag{2.13}$$

where

$$I_{H,\alpha}(m, M) := \frac{C_{\alpha,1}(H)}{m^{\alpha H}} + \frac{C_{\alpha,2}(H)}{(M-1)^{\alpha(1-H)}}. \tag{2.14}$$

The constants $C_{\alpha,i}(H)$, $i = 1, 2$ do not depend on k (see (2.22) *and* (2.23)*, below, for their exact expressions).*

Proof: Consider the SαS random variables

$$\xi := C_{H,\alpha}(m, M) Y_{m,M}(k) = \sum_{j=1}^{mM} \Big((j/m)_+^{H-1/\alpha} - (j/m - 1)_+^{H-1/\alpha} \Big) Z_{\alpha,m}(mk - j), \tag{2.15}$$

where $Z_{\alpha,m}(j) = L_\alpha((j+1)/m) - L_\alpha(j/m)$, $j \in \mathbb{Z}$ and

$$\eta := C_{H,\alpha} Y(k) = -\int_{\mathbb{R}} \Big((s)_+^{H-1/\alpha} - (s-1)_+^{H-1/\alpha} \Big) dL_\alpha(k - s). \tag{2.16}$$

We will first prove that

$$\|\eta - \xi\|_\alpha^\alpha \leq I_{H,\alpha}(m, M), \tag{2.17}$$

where $I_{H,\alpha}(m, M)$ is defined in (2.14). Since $Z_{\alpha,m}(mk - j) = -\int_{(j-1)/m}^{j/m} dL_\alpha(k - s)$, we have that

$$\begin{aligned} \eta - \xi \;\; = \;\; & -\sum_{j=1}^{mM} \int_{(j-1)/m}^{j/m} \Big(s^{H-1/\alpha} - (j/m)^{H-1/\alpha} - ((s-1)_+^{H-1/\alpha} - (j/m - 1)_+^{H-1/\alpha}) \Big) dL_\alpha(k - s) \\ & - \int_M^\infty \Big(s^{H-1/\alpha} - (s-1)^{H-1/\alpha} \Big) dL_\alpha(k - s). \end{aligned}$$

The independence and stationarity of the increments of the Lévy motion $L_\alpha = \{L_\alpha(s),\ s \in \mathbb{R}\}$ implies that

$$\begin{aligned} \|\eta - \xi\|_\alpha^\alpha \;\; = \;\; & \sum_{j=1}^{mM} \int_{(j-1)/m}^{j/m} \Big| s^{H-1/\alpha} - (j/m)^{H-1/\alpha} - ((s-1)_+^{H-1/\alpha} - (j/m - 1)_+^{H-1/\alpha}) \Big|^\alpha ds \\ & + \int_M^\infty \Big| s^{H-1/\alpha} - (s-1)^{H-1/\alpha} \Big|^\alpha ds =: I + J. \end{aligned} \tag{2.18}$$

Consider first the second term J in (2.18). By the mean value theorem applied to the function $x^{H-1/\alpha}$ over the interval $x \in [s-1, s]$, we obtain

$$J \leq |H - 1/\alpha|^{\alpha} \int_{M}^{\infty} (s-1)^{\alpha(H-1)-1} ds = \frac{|H - 1/\alpha|^{\alpha}}{\alpha(1-H)} (M-1)^{-\alpha(1-H)}, \tag{2.19}$$

since $(H - 1/\alpha) - 1 < 0$, $M \geq 2$ and $\alpha(H-1) < 0$.

Now we focus on the second term I in (2.18). By using the inequality $(x+y)^{\alpha} \leq 2x^{\alpha} + 2y^{\alpha}$, valid for all $x, y \geq 0$ and $\alpha \in (0, 2]$, we get that

$$\begin{aligned} I &\leq 4\int_{0}^{1/m} s^{(H-1/\alpha)\alpha} ds + 4\int_{0}^{1/m} (1/m)^{(H-1/\alpha)\alpha} ds + 2\sum_{j=2}^{mM} \int_{(j-1)/m}^{j/m} |s^{H-1/\alpha} - (j/m)^{H-1/\alpha}|^{\alpha} ds \\ &\quad +2\sum_{j=m+2}^{mM} \int_{(j-1)/m}^{j/m} |(s-1)^{H-1/\alpha} - (j/m-1)^{H-1/\alpha}|^{\alpha} ds \\ &\leq \frac{4(1+\alpha H)}{\alpha H} m^{-\alpha H} + 4\sum_{j=2}^{mM} \int_{(j-1)/m}^{j/m} |s^{H-1/\alpha} - (j/m)^{H-1/\alpha}|^{\alpha} ds. \end{aligned} \tag{2.20}$$

The monotonicity of the function $s^{H-1/\alpha}$, for all $s > 0$ implies that, for all $s \in [(j-1)/m, j/m]$, $j \geq 2$,

$$\begin{aligned} |s^{H-1/\alpha} - (j/m)^{H-1/\alpha}|^{\alpha} &\leq |((j-1)/m)^{H-1/\alpha} - (j/m)^{H-1/\alpha}|^{\alpha} \\ &\leq |H - 1/\alpha|^{\alpha} m^{-\alpha} ((j-1)/m)^{\alpha(H-1)-1}, \end{aligned} \tag{2.21}$$

where the last inequality follows by the mean value theorem, applied to the function $x^{H-1/\alpha}$ on the interval $x \in [(j-1)/m, j/m]$.

By using the inequalities in (2.20) and (2.21), we get that

$$\begin{aligned} I &\leq \frac{4(1+\alpha H)}{\alpha H} m^{-\alpha H} + 4|H - 1/\alpha|^{\alpha} m^{-\alpha-1} \sum_{j=1}^{\infty} (j/m)^{\alpha(H-1)-1} \\ &= \frac{4(1+\alpha H)}{\alpha H} m^{-\alpha H} + 4|H - 1/\alpha|^{\alpha} m^{-\alpha H} \sum_{j=1}^{\infty} j^{\alpha(H-1)-1} \\ &\leq \Big(\frac{4(1+\alpha H)}{\alpha H} + 4|H - 1/\alpha|^{\alpha}(1 + \frac{1}{\alpha(1-H)})\Big) m^{-\alpha H} =: C_{\alpha,1}(H) m^{-\alpha H}. \end{aligned} \tag{2.22}$$

The last inequality follows by the fact that $\sum_{j=1}^{\infty} j^{\alpha(H-1)-1} \leq 1 + \int_{1}^{\infty} x^{\alpha(H-1)-1} dx = 1 + 1/\alpha(1-H)$.

By combining the inequalities in (2.19) and (2.22), we obtain the bound in (2.17), where the constant $C_{\alpha,2}(H)$ is given by

$$C_{\alpha,2}(H) := \frac{|H - 1/\alpha|^{\alpha}}{\alpha(1-H)}. \tag{2.23}$$

We shall now use (2.17) to establish (2.11). If follows from (2.5) and (2.15) that $\|\xi\|_{\alpha} = C_{H,\alpha}(m, M)$ and from (2.2), (2.4) and (2.16) that $\|\eta\|_{\alpha} = C_{H,\alpha}$. Therefore (2.15) and (2.16) imply that

$$\|Y(k) - Y_{m,M}(k)\|_\alpha^\alpha = \Big\| \frac{\eta}{\|\eta\|_\alpha} - \frac{\xi}{\|\xi\|_\alpha} \Big\|_\alpha = \Big\| \frac{1}{\|\xi\|_\alpha}(\eta - \xi) + \frac{(\|\xi\|_\alpha - \|\eta\|_\alpha)}{\|\xi\|_\alpha \|\eta\|_\alpha}\eta \Big\|_\alpha^\alpha. \tag{2.24}$$

Suppose first that $1 \le \alpha \le 2$. By using the triangle inequality for the norm $\|\cdot\|_\alpha$ and Relation (2.24), we obtain

$$\|Y(k) - Y_{m,M}(k)\|_\alpha^\alpha \le \Big(\frac{\|\eta - \xi\|_\alpha}{\|\xi\|_\alpha} + \frac{|\|\eta\|_\alpha - \|\xi\|_\alpha|}{\|\xi\|_\alpha} \Big)^\alpha \le \frac{2^\alpha}{\|\xi\|_\alpha^\alpha} \|\eta - \xi\|_\alpha^\alpha \le \frac{2^\alpha}{\|\xi\|_\alpha^\alpha} I_{H,\alpha}(m, M)$$

in view of (2.17). This is (2.11).

Now assume that $0 < \alpha < 1$. By (2.24) and the triangle inequality for the metric $\|\cdot\|_\alpha^\alpha$, we have

$$\|Y(k) - Y_{m,M}(k)\|_\alpha^\alpha \le \frac{1}{\|\xi\|_\alpha^\alpha} \Big(\|\eta - \xi\|_\alpha^\alpha + \Big| \|\eta\|_\alpha - \|\xi\|_\alpha \Big|^\alpha \Big). \tag{2.25}$$

Lemma 2.1, below, implies that

$$\Big| \|\eta\|_\alpha - \|\xi\|_\alpha \Big|^\alpha \le \alpha^{-\alpha} \|\eta - \xi\|_\alpha^{\alpha^2} \Big(\|\eta\|_\alpha + \|\xi\|_\alpha \Big)^{(1-\alpha)\alpha}.$$

By Relations (2.17) and (2.25), the last inequality implies the bound in (2.11), since $\|\xi\|_\alpha = C_{H,\alpha}(m, M)$ and $\|\eta\|_\alpha = C_{H,\alpha}$. This completes the proof of the theorem. □

The following lemma was used in the proof of Theorem 2.1, above.

Lemma 2.1 *Let ξ and η be two non-trivial, jointly $S\alpha S$ random varibles, where $0 < \alpha < 1$. Then, we have that*

$$\Big| \|\eta\|_\alpha - \|\xi\|_\alpha \Big| \le \alpha^{-1} \|\eta - \xi\|_\alpha^\alpha \Big(\|\eta\|_\alpha + \|\xi\|_\alpha \Big)^{1-\alpha}, \tag{2.26}$$

where $\|\xi\|_\alpha$ denotes the scale coefficient of the random variable ξ.

PROOF: For all $0 < x_1 \le x_2$, we have the following inequality

$$x_2 - x_1 \le \alpha^{-1}(x_2^\alpha - x_1^\alpha) x_2^{1-\alpha}. \tag{2.27}$$

Indeed, consider the function $\varphi(x_1) := \alpha^{-1}(x_2^\alpha - x_1^\alpha)x_2^{1-\alpha} - x_2 + x_1$, which is equal to the difference between the right- and left-hand sides of (2.27). For all $x_1 \in (0, x_2]$, the function $\varphi(x_1)$ is non-increasing, since $\alpha - 1 < 0$ and $\varphi'(x_1) = -(x_1/x_2)^{\alpha-1} + 1 \le 0$. This implies that $\varphi(x_1) \ge 0 = \varphi(x_2)$ and hence the inequality (2.27) holds for all $x_1 \in (0, x_2]$.

Without loss of generality let $x_2 := \|\eta\|_\alpha \ge x_1 := \|\xi\|_\alpha$. By (2.27), we then have that

$$\Big| \|\eta\|_\alpha - \|\xi\|_\alpha \Big| \quad \le \quad \alpha^{-1} \Big| \|\eta\|_\alpha^\alpha - \|\xi\|_\alpha^\alpha \Big| \|\eta\|_\alpha^{1-\alpha} \le \alpha^{-1}(\|\eta - \xi\|_\alpha^\alpha) \|\eta\|_\alpha^{1-\alpha},$$

where the last inequality follows by the triangle inequality for the metric $\|\cdot\|_\alpha^\alpha$, $0 < \alpha < 1$. The last relation implies (2.26). □

By using the triangle inequality for the metric $\|\cdot\|_\alpha^{1\wedge\alpha}$, one can also obtain a bound on the error of approximation $\|X_{H,\alpha}(n) - X(n)\|_\alpha^{1\wedge\alpha} \le \sum_{k=1}^n \|Y(k) - Y_{m,M}(k)\|_\alpha^{1\wedge\alpha}$, $n = 1, 2, \ldots, N$ (here $a \wedge b$ stands for $\min\{a, b\}$ and $a \vee b$ for $\max\{a, b\}$). Since the bounds in Theorem 2.1 do not depend on k, one gets

Corollary 2.1 *Let $X_{H,\alpha}$ be LFSM defined in* (2.1) *and $X(n)$ be its approximation defined in* (2.10) *and* (2.5)*. Let $J_{H,\alpha}(m,M)$ be the bound defined in* (2.12)-(2.14)*. For all $n \in \mathbb{N}$, we have*

$$\|X_{H,\alpha}(n) - X(n)\|_\alpha \leq n^{1\vee 1/\alpha} J_{H,\alpha}^{1/\alpha}(m,M), \tag{2.28}$$

and, as $m, M \to \infty$,

$$\|X_{H,\alpha}(n) - X(n)\|_\alpha = \mathcal{O}(1/m^{H(1\wedge\alpha)} + 1/M^{(1-H)(1\wedge\alpha)}). \tag{2.29}$$

The result of Theorem 2.1 implies that

Corollary 2.2 *As m, $M \to \infty$, the approximations $\{Y_{m,M}(k),\ k \in \mathbb{Z}\}$ converge in finite-dimensional distributions to the linear fractional stable noise process $\{Y(k),\ k \in \mathbb{Z}\}$.*

Proof: Let $k_1, \ldots, k_n \in \mathbb{Z}$ and $\lambda_1, \ldots, \lambda_n \in \mathbb{R}$. Let also Y and $Y_{m,M}$ be defined on the *same* probability space by Relations (2.4) and (2.5), then by using the triangle inequality for the metric $\|\cdot\|_\alpha^{1\wedge\alpha}$, we obtain

$$\Big\|\sum_{j=1}^n \lambda_j Y(k_j) - \sum_{j=1}^n \lambda_j Y_{m,M}(k_j)\Big\|_\alpha^{1\wedge\alpha} \leq \sum_{j=1}^n |\lambda_j|^{1\wedge\alpha} \|Y(k_j) - Y_{m,M}(k_j)\|_\alpha^{1\wedge\alpha} \longrightarrow 0, \tag{2.30}$$

as $m, M \to \infty$ by (2.11)-(2.14). This implies the desired convergence in finite-dimensional distributions. Indeed, for a sequence $\xi_n,\ n \in \mathbb{N}$ of SαS random variables, the convergence of the scale coefficients $\|\xi_n\|_\alpha \to 0$, implies $\xi_n \to_d 0$, as $n \to \infty$, where $\to_d$ means convergence in distribution and, since Relation (2.30) holds for all $\lambda_1, \ldots, \lambda_n \in \mathbb{R}$, one has convergence of the finite-dimensional distributions. □

Remarks

1. Independent identically distributed SαS pseudo-random variables $Z_\alpha(j),\ j = 1, \ldots, n$, can be generated by using the *classical* algorithm of Chambers, Mallows and Stuck (1976, 1987) (see also Section 4).

2. Relation (2.29) in Corollary 2.1 provides asymptotic bounds for $\|X_{H,\alpha}(n) - X(n)\|_\alpha$ as $m, M \to \infty$. When the discretization parameters m and M are fixed, however, the error of approximation $\|X_{H,\alpha}(n) - X(n)\|_\alpha$ may increase, as n grows, because $X(n)$ is the cumulative sum of the approximations $Y_{m,M}(1), \ldots, Y_{m,M}(n)$ to the linear fractional stable noise sequence $Y(1), \ldots, Y(n)$. It is thus important to choose *appropriately large* values of m and M, when one wants to generate *long* sample paths of the LFSM process $X_{H,\alpha}$.

   We discuss the choice of the discretization parameters m and M in Section 3.1.

3. When $m(M+N) = 2^p,\ p \in \mathbb{N}$, the complexity of the above algorithm is that of the FFT, namely, of the order $\mathcal{O}(m(N+M)\log_2(m(M+N)))$. On the other hand a direct computation of the sums in (2.5) would have complexity of the order $\mathcal{O}(mMN)$. The FFT-based algorithm improves significantly the speed of computation of the paths of LFSM, in comparison with the direct approach.

4. When $m(M+N)$ is not an integer power of two, then one can use other types of FFT algorithms (e.g. the so-called mixed-radix FFTs) (see, for example, Ch. 12 in Press *et al* (1992)).

2.2 The case of fractional autoregressive moving average (FARIMA)

Fractional autoregressive moving average (FARIMA, in short) time series with SαS, $\alpha \in (1,2)$ innovations were studied in Kokoszka and Taqqu (1995) as an infinite variance counterpart to the FARIMA model introduced by Granger and Joyeux (1980). These models, denoted FARIMA(p,d,q), $p,q \in \mathbb{N}$, extend the usual ARIMA(p,d,q) models, replacing the integer differencing exponent d with an arbitrary fractional real number. In this section we assume that

$$0 < d < 1 - 1/\alpha, \quad \text{and} \quad 1 < \alpha < 2. \tag{2.31}$$

A fractional ARIMA time series $Y = \{Y(k),\ k \in \mathbb{Z}\}$ with SαS innovations is defined as the stationary solution to the back-shift operator equation

$$\Phi_p(B)Y(k) = \Theta_q(B)(I-B)^{-d}Z_\alpha(k),\ k \in \mathbb{Z}, \tag{2.32}$$

where the innovations $Z_\alpha(k)$s are iid standard SαS random variables, $BY(k) := Y(k-1)$ and $\Phi_p(z),\ \Theta_q(z)$ are real polynomials of degrees $p,\ q$, respectively,

$$\Phi_p(z) = 1 - \phi_1 z - \phi_2 z^2 - \cdots - \phi_p z^p, \ \text{ and } \ \Theta_q(z) = 1 - \theta_1 z - \theta_2 z^2 - \cdots - \theta_q z^q, \tag{2.33}$$

with roots outside the unit disk $\{z \in \mathbb{C} : |z| \le 1\}$. These assumptions and condition (2.31) guarantee the existence and uniqueness of the solution Y to (2.32), for more details see e.g. Kokoszka and Taqqu (1995).

Corollary 3.1 in Kokoszka and Taqqu (1995) implies that

$$Y(k) = \sum_{j=0}^{\infty} c(j)Z_\alpha(k-j) = \sum_{j=-\infty}^{k} c(k-j)Z_\alpha(j), \quad k \in \mathbb{Z}, \tag{2.34}$$

where the coefficients $c(j)$ satisfy the relation

$$\Big|\frac{c(j)}{j^{d-1}} - \frac{\Theta_q(1)}{\Phi_p(1)\Gamma(d)}\Big| = O(j^{-1}), \qquad \text{as } j \to \infty. \tag{2.35}$$

The polynomials Φ_p and Θ_q can be used to model a variety of short term dependence structures of the heavy-tailed time series $Y(k),\ k \in \mathbb{Z}$ (see Figure 2). On the other hand, because of relation (2.35), the parameter d governs the long-term behavior of the FARIMA model. In fact, since $0 < d < 1 - 1/\alpha$ and $c(j) \sim j^{d-1}$, as $j \to \infty$, the moving average coefficients $c(j)$ decay very slowly, so that $\sum_j |c(j)| = \infty$. This indicates that the process Y is strongly dependent in time and the parameter d can be interpreted as the long-range dependence exponent of Y. For more details and applications see Kokoszka and Taqqu (1996).

We will now discuss a fast method for simulation of FARIMA time series, which involves the FFT algorithm.

- *The FARIMA*$(0,d,0)$ *case.* The moving average coefficients $c(j) := b(j),\ j \in \mathbb{Z}$ are given by

$$b(0) = 1, \ \ b(j) = \frac{\Gamma(j+d)}{\Gamma(d)\Gamma(j+1)}, \ \ j = 1,2,\ldots \tag{2.36}$$

(see e.g. Relations (2.3) and (2.4) in Kokoszka and Taqqu (1995)). We approximate the path $Y(n),\ n = 0, 1, \dots, N-1,\ N \in \mathbb{N}$ by the truncated moving average

$$Y(n) \approx Y_M(n) := \sum_{j=0}^{M-1} c(j) Z_\alpha(n-j),\ n = 0, 1, \dots, N-1. \tag{2.37}$$

Using Relation (2.35), one can estimate the error of approximation in (2.37); for more details see Theorem 2.2 below.

We now briefly describe the simulation procedure. As in the case of LFSM, let

$$\widetilde{b}(j) = \begin{cases} b(j) & ,\ \text{for } j = 0, \dots, M-1, \\ 0 & ,\ \text{for } j = M, \dots, M+N-1, \end{cases} \tag{2.38}$$

and set $\widetilde{b}(j + k(M+N)) := \widetilde{b}(j),\ j = 0, \dots, M+N-1,\ k \in \mathbb{Z}$, so that the sequence $\widetilde{b} = (\widetilde{b}(j),\ j \in \mathbb{Z})$ is $(M+N)$-periodic. Define also the $(M+N)$-periodic sequence

$$\widetilde{Z}(j + k(M+N)) := Z_\alpha(j),\ j = 0, 1, \dots, M+N-1,\ k \in \mathbb{Z}, \tag{2.39}$$

where $Z_\alpha(j),\ j = 0, 1, \dots, M+N-1$ are iid standard SαS random variables. Then by (2.37), we have

$$\{Y_M(n),\ n = 0, \dots, N-1\} =_d \Big\{ \sum_{j=0}^{M+N-1} \widetilde{b}(j) \widetilde{Z}(n-j),\ n = 0, \dots, N-1 \Big\}.$$

We can now use the FFT algorithm to compute the last circular convolutions (as in the case of LFSM in Section 2.1). That is, we take the inverse DFT of the product of the DFTs of the two $(M+N)$-periodic sequences $\widetilde{b}$ and $\widetilde{Z}$. More precisely, we have the following algorithm.

Algorithm

1. Choose a truncation parameter $M \in \mathbb{N}$ and a sample size $N \in \mathbb{N}$, so that $(M+N)$ is an integer power of two.

2. By using the Relations (2.36) and (2.38), calculate the set of coefficients $\widetilde{b}(j),\ j = 0, 1, \dots, M+N-1$. Using the radix-2 FFT algorithm, compute the DFT $\widehat{b}(k) := \mathcal{D}_{M+N}(\widetilde{b})(k),\ k = 0, 1, \dots, M+N-1$ of the $(M+N)$-periodic sequence $\widetilde{b}$.

3. Generate $(M+N)$ standard iid SαS pseudo-random variables $\widetilde{Z}(j),\ j = 0, 1, \dots, M+N-1$. By using the radix-2 FFT, compute the DFT $\widehat{Z}(k) := \mathcal{D}_{M+N}(\widetilde{Z})(k),\ k = 0, 1, \dots, M+N-1$ of the $(M+N)$-periodic sequence $\widetilde{Z}$, defined in (2.39).

4. By using the radix-2 FFT algorithm, compute the inverse DFT of the product $(\widehat{b}\widehat{Z})$ and let
$$Y_M(j) := \mathcal{D}^{-1}_{M+N}(\widehat{b}\widehat{Z})(j),\ j = 0, 1, \dots, N-1,$$
where $(\widehat{b}\widehat{Z})(k) := \widehat{b}(k)\widehat{Z}(k),\ k = 0, 1, \dots, M+N-1$.

   Take the sequence $Y_M(j),\ j = 0, 1, \dots, N-1$ as the *simulated* path of the FARIMA time series Y.

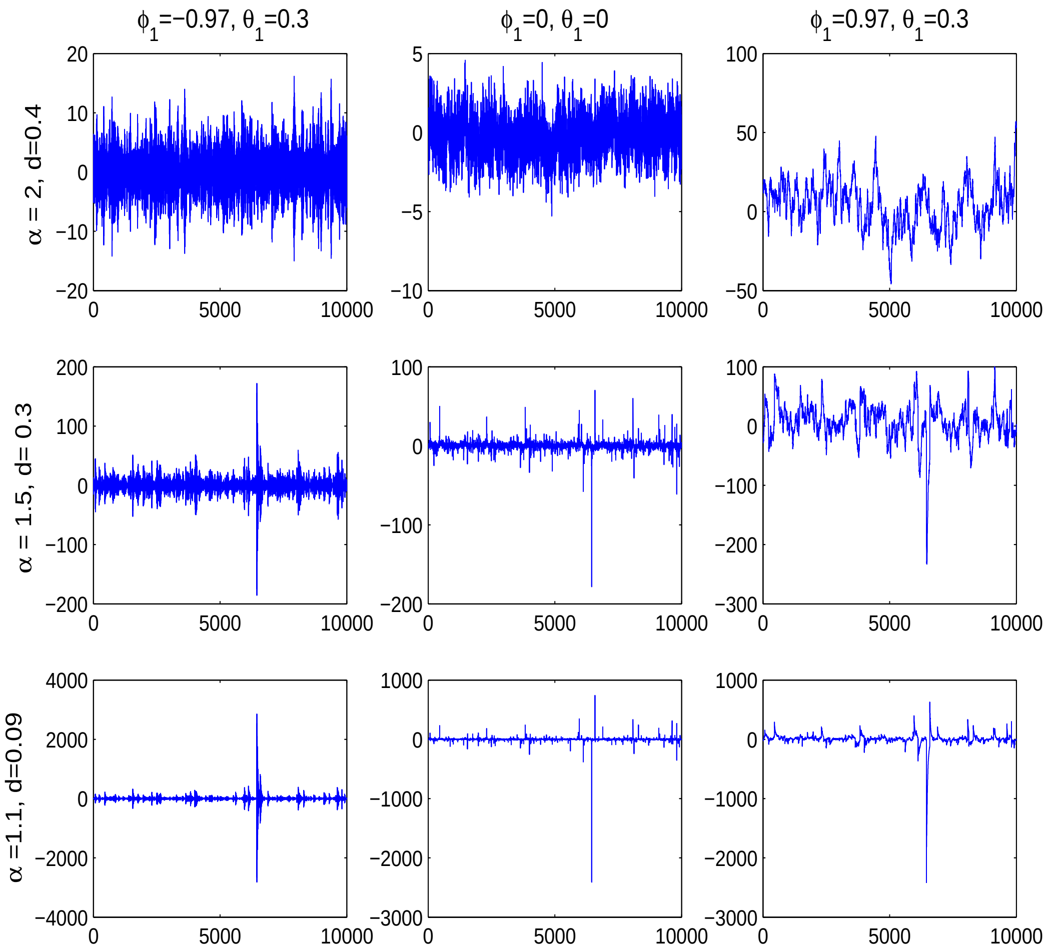

Figure 2: Each graph displays one simulated path of FARIMA$(1, d, 1)$ with SαS innovations, $N = 10,000$ and $N+M = 2^{20}$ (see Section 2.2). All nine paths were computed by using the same seed for the random number generator. Thus paths with equal values of the index of stability α, involve the same samples of iid SαS innovations. The vertical scale is different from one graph to the other. The paths in the first column correspond to the so-called *weak short-range dependence* case and those in the last column to the *strong short-range dependence* case (see, for example, Section 7 in Stoev and Taqqu (2003*a*)).

Section 3.2 contains a brief numerical illustration of this algorithm. Its MATLAB implementation can be found in Section 4.

• *The FARIMA(p, d, q) case.* Let $b(j),\ j = 0, 1, \ldots$ denote the moving average coefficients of a FARIMA$(0, d, 0)$ time series given in (2.36). Then for the coefficients $c(m),\ m \in \mathbb{Z}$ in (2.34), we have the formulas

$$c(m) = b(m) + \sum_{j=1}^{p} \phi_j c(m-j) - \sum_{k=1}^{q} \theta_k b(m-k),\ m = 0, 1, \ldots, \tag{2.40}$$

where $c(m) = 0$, for $m < 0$ and where ϕ_j and θ_k are the coefficients of $\Phi_p(z)$ and $\Theta_q(z)$ in (2.33), above (see e.g. Relation (2.8) in Kokoszka and Taqqu (1995)). By using (2.40) and the expression for $b(j)$ in (2.36), we compute the coefficients $c(j),\ j = 0, 1, \ldots, M-1$. We then use the truncated moving average in (2.37) to obtain an approximate path $Y_M(k),\ k = 0, 1, \ldots, N-1$ of a FARIMA(p, d, q) time series Y.

The path $Y_M(j),\ j = 0, 1, \ldots, N-1$ can be computed by using the *same* algorithm as in the FARIMA$(0, d, 0)$ case. One must only change step 2, which now involves the coefficients $c(j)$, defined in (2.40), instead of $b(j)$, defined in (2.36).

The following theorem provides the first order of the approximation error $\|Y_M(k) - Y(k)\|_\alpha$, as $M \to \infty$.

Theorem 2.2 *Let $\alpha \in (1, 2),\ d \in (0, 1 - 1/\alpha),\ H = d + 1/\alpha$ and $M \in \mathbb{N}$. Let also Y be the FARIMA time series defined in* (2.34) *and Y_M be its approximation, defined in* (2.37). *For all $k \in \mathbb{Z}$, as $M \to \infty$, we have*

$$\|Y(k) - Y_M(k)\|_\alpha - \frac{C(H, \alpha)}{M^{1-H}} = \mathcal{O}\Big(\frac{1}{M^{(1-H)+1}}\Big), \tag{2.41}$$

where $C(H, \alpha) = |\Theta_q(1)|/(|\Phi_p(1)|\Gamma(d)(1-H)^{1/\alpha}\alpha^{1/\alpha})$.

Proof: Consider the random variable $\xi_M = C\sum_{j=M}^{\infty} j^{d-1} Z_\alpha(k-j)$, where $C = \Theta_q(1)/(\Phi_p(1)\Gamma(d))$. Since the random variables $Z_\alpha(j),\ j \in \mathbb{Z}$ are standard SαS and independent, we have that

$$\|\xi_M\|_\alpha^\alpha = |C|^\alpha \sum_{j=M}^{\infty} j^{(d-1)\alpha}. \tag{2.42}$$

For all $j \geq 2$, $\int_j^{j+1} x^{(d-1)\alpha} dx \leq j^{(d-1)\alpha} \leq \int_{j-1}^{j} x^{(d-1)\alpha} dx$, since $(d-1)\alpha < 0$. Thus,

$$\int_M^\infty x^{(d-1)\alpha} dx \leq \sum_{j=M}^{\infty} j^{(d-1)\alpha} \leq \int_{M-1}^{\infty} x^{(d-1)\alpha} dx$$

and therefore, by (2.42), we get

$$\frac{|C|^\alpha}{\alpha(1-H)} \frac{1}{M^{\alpha(1-H)}} \leq \|\xi_M\|_\alpha^\alpha \leq \frac{|C|^\alpha}{\alpha(1-H)} \frac{1}{(M-1)^{\alpha(1-H)}}, \tag{2.43}$$

because $\int_M^\infty x^{(d-1)\alpha}dx = (\alpha(1-H))^{-1}M^{-\alpha(1-H)}$. Observe that the mean value theorem applied to the function $-x^{-(1-H)}$ on the interval $x \in [M-1, M]$ yields

$$-\frac{1}{M^{1-H}} + \frac{1}{(M-1)^{1-H}} \le (M-(M-1))\frac{1-H}{(M-1)^{(1-H)+1}} = \mathcal{O}\Big(\frac{1}{M^{(1-H)+1}}\Big), \tag{2.44}$$

as $M \to \infty$. Hence, by raising the expression in (2.43) to the power $1/\alpha$, and using (2.44), we obtain

$$\|\xi_M\|_\alpha - \frac{|C|}{(\alpha(1-H))^{1/\alpha}}\frac{1}{M^{1-H}} = \mathcal{O}\Big(\frac{1}{M^{(1-H)+1}}\Big), \tag{2.45}$$

as $M \to \infty$.

We now estimate the error of approximation $\|Y(k) - Y_M(k)\|_\alpha$. By using Relations (2.34) and (2.37), the fact that $Z_\alpha(j)$s, $j \in \mathbb{Z}$ are iid standard SαS and the triangle inequality for the norm $\|\cdot\|_\alpha$, we obtain

$$\begin{aligned}
&\Big|\|Y(k) - Y_M(k)\|_\alpha - \|\xi_M\|_\alpha\Big| \le \|Y(k) - Y_M(k) - \xi_M\|_\alpha \\
&= \Big\|\sum_{j=M}^\infty c(j)Z_\alpha(k-j) - C\sum_{j=M}^\infty j^{d-1}Z_\alpha(k-j)\Big\|_\alpha = \Big(\sum_{j=M}^\infty |c(j) - Cj^{d-1}|^\alpha\Big)^{1/\alpha} \\
&= \Big(\sum_{j=M}^\infty \Big|\frac{c(j)}{j^{d-1}} - C\Big|^\alpha j^{(d-1)\alpha}\Big)^{1/\alpha} = \mathcal{O}\Big(\frac{1}{M}\Big)\Big(\sum_{j=M}^\infty j^{(d-1)\alpha}\Big)^{1/\alpha} = \mathcal{O}\Big(\frac{1}{M^{(1-H)+1}}\Big), \qquad (2.46)
\end{aligned}$$

as $M \to \infty$. In the last expression we used Relation (2.35) and the fact that $\sum_{j=M}^\infty j^{(d-1)\alpha} = \mathcal{O}(M^{(d-1)\alpha+1}) = \mathcal{O}(M^{-\alpha(1-H)})$, as $M \to \infty$.

Since $C(H,\alpha) = |C|/(\alpha(1-H))^{1/\alpha}$, by using the triangle inequality, (2.45) and (2.46), we get that the left-hand side of (2.41) is bounded above by

$$\Big|\|Y(k) - Y_M(k)\|_\alpha - \|\xi_M\|_\alpha\Big| + \Big|\|\xi_M\|_\alpha - \frac{C(H,\alpha)}{M^{1-H}}\Big| = \mathcal{O}(\frac{1}{M^{(1-H)+1}}),$$

which completes the proof of the theorem. □

Remarks

1. If $M+N = 2^p$, $p \in \mathbb{N}$, then the complexity of the above simulation algorithm is of the order $\mathcal{O}((M+N)\log_2(M+N))$. Observe that the coefficients $b(j)$ and $c(j)$, $j = 0, 1, \ldots, M-1$ can be computed by using $\mathcal{O}(M)$ operations.

2. In view of Relation (2.41) in Theorem 2.2, the expression $C(H,\alpha)/M^{1-H}$ provides a *very good* estimate for the error of approximation $\|Y(k) - Y_M(k)\|_\alpha$ in the simulation algorithm.

3. The main advantage of the FFT-based simulation method over a direct computation of the moving averages in (2.37) is that it allows us to work with *very* large values of N and M, using small amount of time and computer memory.

3 Implementation of the Algorithms

3.1 The case of linear fractional stable motion (LFSM)

In this section we illustrate briefly the simulation algorithm for the LFSM process $X_{H,\alpha}$ given in Section 2.1. Our main goal here is to show how different choices of the discretization parameters m and M affect the quality of the simulated paths.

We generate paths of the LFSM process $X_{H,\alpha}$ by using the algorithm in Section 2.1. We choose different values of the discretization parameters m and M and the parameters H and α. For each choice of m, M, H and α, we generate $n = 256$ independent paths $X(1), \dots, X(N)$ of length $N = 10,000$. Then we evaluate the quality of the simulated paths by estimating the self-similarity parameter H. We use *wavelet transform* estimators $\widehat{H}_{WT}$ and *finite impulse response* estimators $\widehat{H}_{FIRT}$. We will briefly recall the definitions and some basic properties of these estimators (see Stoev, Pipiras and Taqqu (2002), for more details).

- *The wavelet and FIRT estimators $\widehat{H}_{WT}$ and $\widehat{H}_{FIRT}$* involve the following arrays of coefficients

$$D_\kappa = \{D_\kappa(n,k),\ n \in \mathbb{N},\ k \in \mathbb{Z}\},$$

where κ stands either for the label "*WT*" or the label "*FIRT*". In the case of the estimator $\widehat{H}_{FIRT}$, the array D_{FIRT} is made out of *finite impulse response coefficients* $D_{FIRT}(n,k)$ of the path $X(k),\ k = 1, \dots, N$, defined as

$$D_{FIRT}(n,k) = \sum_{i=0}^{p} u_i X(n(i+k)),\ n \geq 1,\ k \in \mathbb{Z}, \tag{3.1}$$

where $u = (u_0, u_1, \dots, u_p)$ is a discrete linear filter with $Q \geq 1,\ Q \in \mathbb{N}$ zero moments. That is,

$$\sum_{i=0}^{p} i^q u_i = 0,\ q = 0, 1, \dots, Q-1, \quad \text{and} \quad \sum_{i=0}^{p} i^Q u_i \neq 0.$$

In the case of the wavelet estimator $\widehat{H}_{WT}$, the array D_{WT} is made out of *wavelet coefficients* $D_{WT}(n,k)$, computed from the path $X(k),\ k = 1, \dots, N$. Namely,

$$D_{WT}(n,k) = \int_{\mathbb{R}} \widetilde{X}(t) n^{-1} \psi(n^{-1}t - k) dt,\ n \geq 1,\ k \in \mathbb{Z}. \tag{3.2}$$

where ψ is a *mother wavelet* function with compact support and $Q \geq 1$ zero moments, that is,

$$\int_{\mathbb{R}} t^q \psi(t) dt = 0,\ q = 0, 1, \dots, Q-1, \quad \text{and} \quad \int_{\mathbb{R}} t^Q \psi(t) dt \neq 0.$$

The process $\widetilde{X}$ is the continuous-time interpolation of the path $X(k),\ k = 1, \dots, N$ via the *scaling function* φ associated with the wavelet ψ, that is,

$$\widetilde{X}(t) := \sum_{k=1}^{N} X(k) \varphi(t-k).$$

We choose a wavelet ψ which generates an orthonormal basis of $L^2(\mathbb{R})$ through a *multiresolution analysis* (for more details see for example Daubechies (1992)).

The arguments n and k of the coefficients $D_\kappa(n,k),\ \kappa \in \{WT, FIRT\}$ are referred to as a *scale* and *location*. When n is large, the coefficients $D_\kappa(n,k),\ k \in \mathbb{Z}$ capture "coarse" details of the time series, and when n is small the $D_\kappa(n,k)$s capture "fine" (or high-frequency) details of the process. Observe that on a scale $n \in \mathbb{N},\ n < N$ there are only about $[N/n]$ non-zero coefficients $D_\kappa(n,k)$, because the wavelet ψ and the scaling function φ have compact supports and because the filter u is finite. In practice, one often uses dyadic scales $n = 2^m,\ m \in \mathbb{N}$ and computes the wavelet coefficients $D_{WT}(n,k)$s via the Mallat's fast discrete wavelet transform algorithm (see, for example Mallat (1998)).

The estimators $\widehat{H}_\kappa,\ \kappa \in \{WT, FIRT\}$ are obtained by performing a log-linear regression on the coefficients and measuring the slope. Formally,

$$\widehat{H}_\kappa = \sum_{j=j_1}^{j_2} w_j \Big(\frac{1}{N_j} \sum_{k=1}^{N_j} \log_2 |D_\kappa(2^j, k)| \Big), \tag{3.3}$$

where $j_1, j_2, N_j \in \mathbb{N},\ j = j_1, \ldots, j_2$, and where the weights $w_j,\ j = j_1, \ldots, j_2$ are such that

$$\sum_{j=j_1}^{j_2} w_j = 0 \quad \text{and} \quad \sum_{j=j_1}^{j_2} j w_j = 1$$

(for more details see Stoev, Pipiras and Taqqu (2002)). Relation (3.3) involves parameters $j_1,\ j_2$ and $N_j,\ j = j_1, \ldots, j_2$.

- *Properties of the estimators $\widehat{H}_{WT}$ and $\widehat{H}_{FIRT}$*

  1. The estimator $\widehat{H}_{FIRT}$ is *unbiased* for all $0 < \alpha < 2$ (Proposition 2.3 in Stoev, Pipiras and Taqqu (2002)). When $1 < \alpha < 2$, under certain technical conditions, the estimators $\widehat{H}_{WT}$ and $\widehat{H}_{FIRT}$ are *consistent* and *asymptotically normal*, as $N \to \infty$ and $j_1,\ j_2 \to \infty$, (Theorems 3.1 and 3.2, Stoev, Pipiras and Taqqu (2002)).

  2. In practice, when $1 < \alpha < 2$, as shown in Stoev, Pipiras and Taqqu (2002), although both estimators $\widehat{H}_{WT}$ and $\widehat{H}_{FIRT}$ are *very good* and essentially unbiased, $\widehat{H}_{WT}$ is slightly better. In fact, the wavelet estimator $\widehat{H}_{WT}$ seems to be more robust and with slightly lower sample variance than the estimator $\widehat{H}_{FIRT}$.

  3. In the case $0 < \alpha \leq 1$, the wavelet estimator $\widehat{H}_{WT}$ has greater bias than the estimator $\widehat{H}_{FIRT}$. In fact, the wavelet coefficients $D_{WT}(n,k)$ are not well-defined unless $H - 1/\alpha > -1$ and the estimators $\widehat{H}_{WT}$ do not work well for $\alpha < 1/2$. On the other hand, the estimators $\widehat{H}_{FIRT}$ are unbiased and still perform well for all $0 < \alpha \leq 1$. Thus in the sequel we will use the estimators $\widehat{H}_{WT}$ in the case $1 < \alpha < 2$, and the estimators $\widehat{H}_{FIRT}$ in the case $0 < \alpha \leq 1$.

Figures 3 and 4 contain approximate 95% confidence intervals for the bias $\mathbb{E}(\widehat{H}_\kappa) - H$ of the estimators $\widehat{H}_\kappa,\ \kappa \in \{WT, FIRT\}$. These confidence intervals are obtained by using the normal approximation, based on samples of $n = 256$ independent replications of the estimators. More precisely, for each set of values of the parameters m, M, H and α, we generated independently n paths of LFSM, computed the estimators $\widehat{H}_\kappa$ and then we obtained the following confidence intervals

$$\Big[\overline{H}_\kappa - H - z_{0.025}\frac{S_{WT}}{\sqrt{n}}, \overline{H}_\kappa - H + z_{0.025}\frac{S_{WT}}{\sqrt{n}}\Big],\ \kappa \in \{WT, FIRT\}. \tag{3.4}$$

In the above expression $\overline{H}_\kappa$ and S_κ denote the sample mean and the sample standard deviation of the n estimates $\widehat{H}_\kappa$ and $z_{0.025} \approx 1.96$ denotes the 0.975th quantile of the standard normal distribution.

The estimators $\widehat{H}_{WT}$ were computed by using Daubechies wavelets with 3 zero moments and all available dyadic scales with the exception of the first 3, that is, $j_1 = 4$ and $j_2 = 12$. We choose $j_1 = 4$ (instead of $j_1 = 1$) to get rid of the potential effects due to the initialization of the Mallat's fast discrete wavelet transform algorithm. To take care of the filter edge effects we choose $j_2 = 12 < [\log_2(N)] = 13$, that is $2^{12} = 4,096 < N = 10,000$. For the FIRT estimators $\widehat{H}_{FIRT}$, we used the filter $u = (1, -3, 3, -1)$, which also has 3 zero moments and the same dyadic scales as in the case of the wavelet estimator $\widehat{H}_{WT}$. The number of coefficients N_j at a scale 2^j without edge effects in (3.3) is of the order $N/2^j$. More precisely, Relations (3.1) and (3.2) imply that $N_j = [N/2^j] - p$ in the case of FIRT; and $N_j = [N/2^j - K - K/2^j] - [K + K/2^j]$ in the case of WT, where $K > 0$ is such that $\text{supp}\{\varphi\} \cup \text{supp}\{\psi\} \subset [-K, K]$, and where $[x]$ denotes the integer part of the number $x \in \mathbb{R}$.

- We will now make some general observations, based on the results on Figures 3 and 4.

1. Consider Figure 3 where $\alpha = 1.5$. Observe that when $m = 64$ and even when $m = 16$, the simulated paths of LFSM are *very good*, for all choices of the parameter M and for most values of the self-similarity parameter H.

   The best choice of the discretization parameters, in this case, seems to be $m = 64$ and $M = 600$, for most values of H. As we will see below the value of m should be adapted to that of M, particularly in the case of negative dependence $H < 1/\alpha$.

2. In the long-range dependence cases, that is, $H > 1/\alpha \approx 0.67$, the simulated paths seem to improve slightly when greater values for the cut-off parameter M are chosen (see Figure 3), since the corresponding confidence intervals for the bias become more centered around zero.

3. Surprisingly, the choice $m = 256$ (on Figure 3, where $\alpha = 1.5$) *impairs* the precision of the paths for most parameter values of H and M, especially in the negative-dependence cases, that is, $0 < H < 1/\alpha \approx 0.67$. Observe, however, that when $m = 256$, for all values of H, the paths seem to improve when larger values of M are chosen and that in all long-range dependence cases ($H > 1/\alpha \approx 0.67$) the simulated paths are admissible when $M = 600$ or $M = 6000$.

4. Consider now Figure 4 where $\alpha = 0.5$ and hence the noise can take very large values. It seems here that our choices of the parameters m and M are inappropriate, for many values of H, since the corresponding confidence intervals do not cover zero. In contrast with Figure 3, the choices $m = 16$ and $m = 64$ do *not* work. On the other hand the choice $m = 256$ is admissible for most values of H when $M = 60,\ 600$ and 6000.

5. In extreme situations, for example $H = 0.1$, $\alpha = 0.5$ (see Figure 4) no choice of the parameters m and M seems to be appropriate. Observe, however, that when $\alpha = 0.5$, the quality of the paths improves, for all values of H and M when larger values of the mesh-parameter m are chosen.

6. The quality of the simulated paths seems not to improve significantly when "very large" values of the parameter M are chosen and when the parameter m is held *fixed*.

Remarks

1. Since the complexity of the FFT-based simulation algorithm is $\mathcal{O}(m(M+N)\log_2(m(M+N)))$ when $m(M+N)$ is an integer power of 2, we chose only dyadic values for the mesh-parameter m and for the sum $N+M$.

2. The time it takes to compute one path of the LFSM process of length N on a dedicated PC Pentium II (Deschutes) with CPU frequency 447.701 MHz, $770,084$ kilobytes of RAM and Red Hat Linux OS version 7.1 is *approximately* 77.63 seconds when $m = 2^8 = 256$, $M = 6000$ and $N + M = 2^{14}$, that is $N = 10,384$.

3.2 The case of fractional autoregressive moving average (FARIMA)

The FARIMA time series $Y = \{Y(k),\ k \in \mathbb{Z}\}$, with parameter $d > 0$, defined in (2.32), can be used to model processes, which exhibit large fluctuations (heavy-tailed distributions) and also have strong dependence in time (long-range dependence) (see Sections 1 and 2.2 for more details). We present in this section a numerical study of the simulation algorithm for these time series, described in Section 2.2. As in Section 3.1 above, our main goal is to demonstrate how different choices of the truncation parameter M affect the precision of the simulated paths.

Recall that the simulated paths $Y_M = \{Y_M(k),\ k = 1,\dots,N\}$ of FARIMA defined in (2.37), are in fact moving averages of iid SαS ($1 < \alpha < 2$) random variables. Thus, for all $M \in \mathbb{N}$ the generated approximate paths of FARIMA will have SαS marginal distributions, that is, heavy-tails (see Figure 2).

We first consider the effect of block-averaging. To do so observe that the FARIMA time series $Y = \{Y(k),\ k \in \mathbb{Z}\}$ is *asymptotically self-similar* with Hurst parameter $H = d + 1/\alpha$. That is, we have that, as $N \to \infty$,

$$\Big\{N^{-H}\sum_{j=1}^{[Nt]} Y(j),\ t \geq 0\Big\} \overset{f.d.d.}{\longrightarrow} \{CX_{H,\alpha}(t),\ t \geq 0\}, \tag{3.5}$$

where $X_{H,\alpha}(t),\ t \in \mathbb{R}$ is the LFSM process defined in (2.1) and where $C \neq 0$ is a constant (see Astrauskas (1983) or Proposition 5.1 in Stoev and Taqqu (2003*a*), for example). The interpretation of Relation (3.5) is two-fold. Firstly, since $H < 1$, (3.5) implies that $N^{-1}\sum_{j=1}^{N} Y(j) \to^{P} 0 = \mathbb{E}Y(j)$, that is the weak law of large numbers holds for the time series Y. This fact also follows from the general result that the time series Y is *ergodic* (see Cambanis, Hardin and Weron (1984)). Secondly, one can get the rate of convergence of the block-wise averages

$$Y(k;B) = \frac{1}{B}\sum_{j=1}^{B} Y(j+kB),\ k \in \mathbb{Z}$$

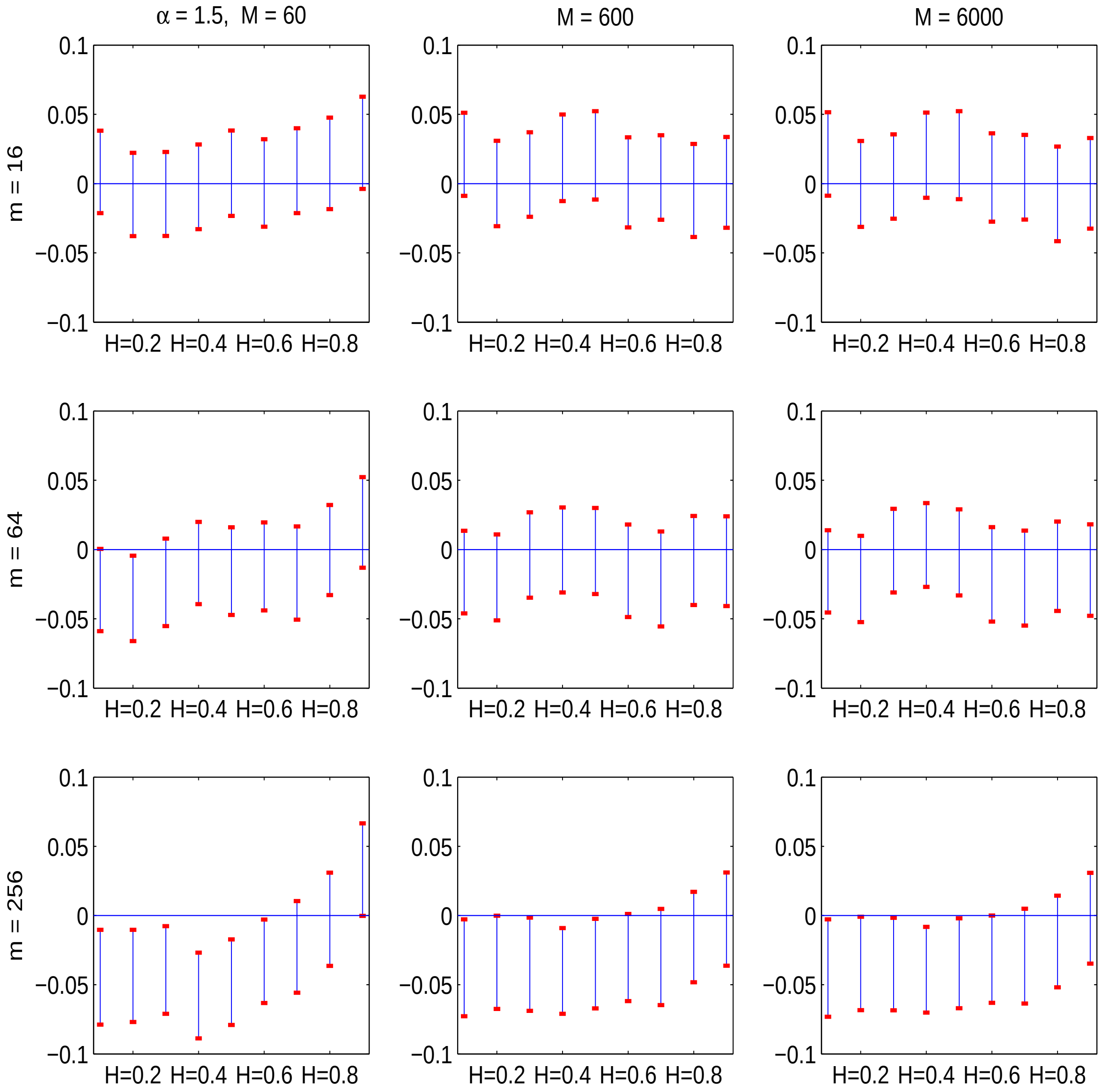

Figure 3: This figure contains approximate 95%-confidence intervals for the bias $\mathbb{E}(\widehat{H}_{WT}) - H$ of the wavelet estimators $\widehat{H}_{WT}$. These confidence intervals were computed by using samples of $n = 256$ independent replications of the estimators, obtained from 256 independently simulated paths of the LFSM process with $\alpha = 1.5$. The values of H are $H = 0.1,\ 0.2, \ldots, 0.9$ (note that $H - 1/\alpha \approx H - 0.67$, so that $H = 0.7,\ 0.8$ and 0.9 correspond to long-range dependence). The values of the discretization parameters m and M, used in the simulation algorithm, are indicated in the margins of the figure. Plots in the same row involve equal values for the parameter m and for those in the same column we used the same values for the parameter M, respectively. Observe that the scales on all nine plots are equal.

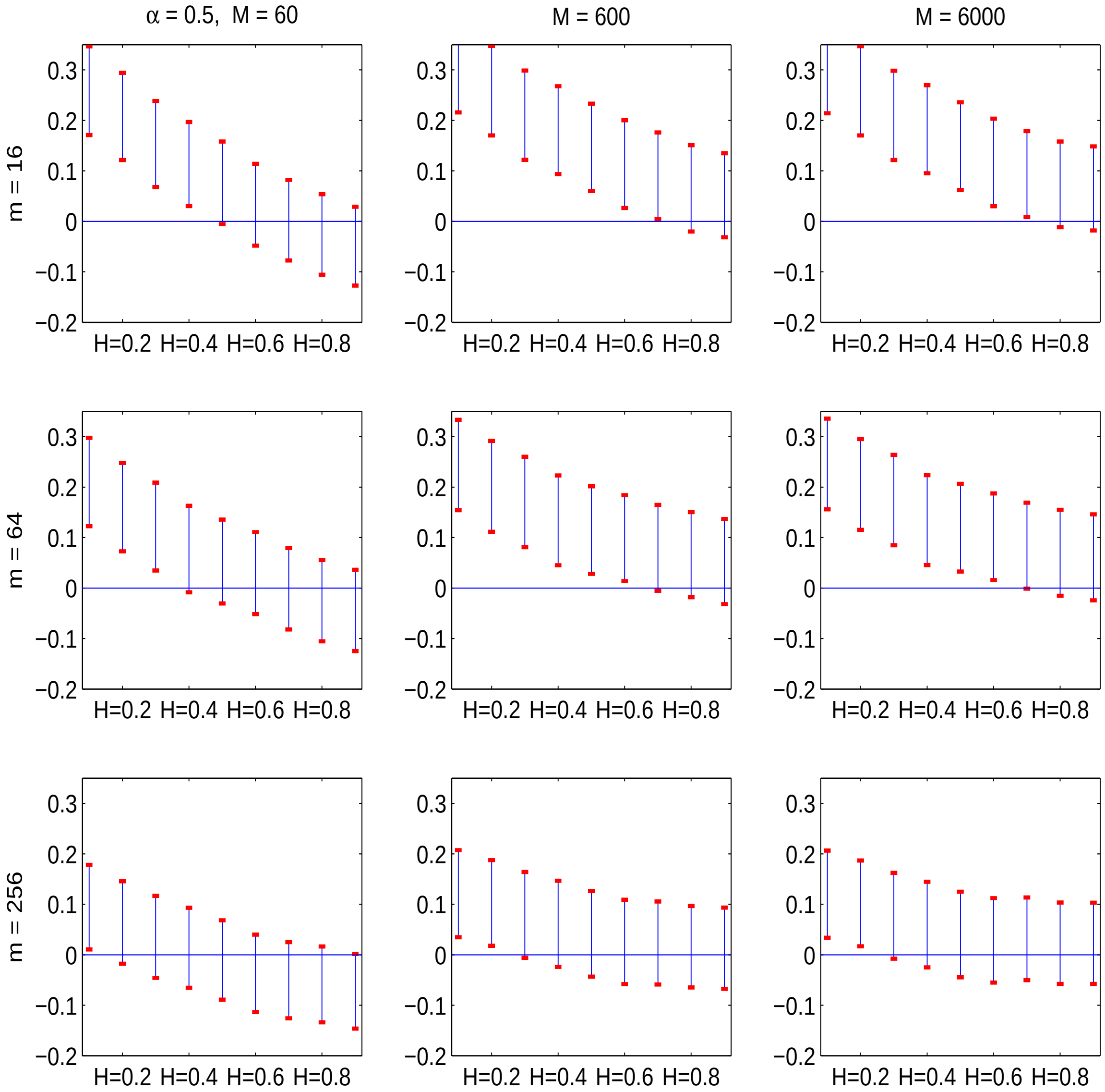

Figure 4: This is obtained as Figure 3. It contains approximate 95%-confidence intervals for the bias $\mathbb{E}(\widehat{H}_{FIRT}) - H$ of the FIRT estimators $\widehat{H}_{FIRT}$. These confidence intervals were computed by using samples of $n = 256$ independent replications of the estimators, obtained from 256 independently simulated paths of the LFSM process. Observe that now the index of stability $\alpha = 0.5$ and that the scales on all nine plots are equal. Since $\alpha < 1$, all cases correspond to negative dependence $H < 1/\alpha$.

to the mean $\mathbb{E}Y(j) = 0$. Indeed, (3.5) implies

$$B^{1-H}\Big(Y(k;B) - 0\Big) \xrightarrow{d} \xi, \quad \text{as } B \to \infty, \tag{3.6}$$

where ξ is a non-trivial SαS random variable. If $Y(j)$ were independent we would have

$$B^{1-1/\alpha}\Big(Y(k;B) - 0\Big) \xrightarrow{d} \xi, \quad \text{as } B \to \infty.$$

Since $H = d + 1/\alpha > 1/\alpha$, Relation (3.6) implies that the averages $Y(k;B)$ converge, as $B \to \infty$, *more slowly* to their mean $\mathbb{E}Y(j) = 0$ than averages of independent and identically distributed SαS random variables. This fact reveals the *long-range dependence* of the FARIMA time series Y. The Hurst parameter H is an indicator of the degree of long-term dependence of Y.

We will show now that if we choose M too small, the approximations converge relatively fast to the mean $\mathbb{E}Y(j) = 0$, an indication that the approximated sequence is weakly dependent. We will contrast this behavior with what happens when M is chosen very large.

In Figures 5 and 6, we simulate a FARIMA$(0, d, 0)$ sequence with $H = d + 1/\alpha = 0.9 > 1/\alpha$, that is with long-range dependence. We approximate the sequence with $M = 1000$ and $M = 2^{20}$. Figure 5 clearly indicates that a value of $M = 1000$ is insufficient when $\alpha = 1.5$, since the block-wise averaged time series converge rapidly to the mean 0 and hence are rather weakly dependent. (This should be contrasted with the case $M = 2^{20}$ where the convergence to the mean occurs at a slower rate.)

Figure 6 deals with the heavier-tailed situation $\alpha = 1.2$. In this case, the previous method breaks down because the resulting graphs are similar and hence do not allow one to conclude that the choice $M = 1000$ is bad. The heavy-tailed nature of the simulated time series implies that in practice the convergence of averages of iid SαS random variables is relatively slow. Thus one needs to use a different method in order to distinguish between weakly and strongly dependent heavy-tailed time series. In the following analysis we use wavelet estimators for the Hurst parameter $H = d + 1/\alpha$ of the FARIMA time series to determine whether the value of the truncation parameter M is sufficiently large.

Consider the aggregated process $X(n) = \sum_{j=1}^{n} Y(j),\ n \in \mathbb{N}$. Relation (3.5) suggests that one can estimate the Hurst parameter $H = d + 1/\alpha$ of the FARIMA time series by using the wavelet estimators for the self-similarity parameter of the LFSM process $X_{H,\alpha}$. Indeed, let $\widehat{H}_{WT}$ be the log-type wavelet estimator defined in (3.3). Under certain technical conditions, the estimator $\widehat{H}_{WT}$ is consistent (Proposition 4.1 in Stoev and Taqqu (2003*b*)) and asymptotically normal (Theorem 5.1 in Stoev and Taqqu (2003*a*)), as $N \to \infty$ and as $n_j \to \infty,\ j = 1, \ldots, m$. In practice, one uses the Mallat's fast discrete wavelet transform algorithm and discards the first few smallest dyadic scales, which are impaired by the potential presence of non-trivial polynomials $\Phi(\cdot)$ and $\Theta(\cdot)$ in (2.32) of the FARIMA time series (for more details and a simulation study see Stoev and Taqqu (2003*a*)).

In Figures 7 and 8, we present approximate 95% confidence intervals for the parameter H computed from samples of $n = 256$ independent estimates $\widehat{H}_{WT}$ (see (3.4)). The estimators $\widehat{H}_{WT}$ were computed from n independently generated paths $Y_M(1), \ldots, Y_M(N),\ N = 2^{20}$ of FARIMA$(0, d, 0)$ and FARIMA$(1, d, 1)$ time series. We used the Daubechies wavelet with 3 zero moments and dyadic scales $2^{j_1}, 2^{j_1+1}, \ldots, 2^{j_2}$, where $j_2 \le [\log_2(N)] = 20$ is the maximal available

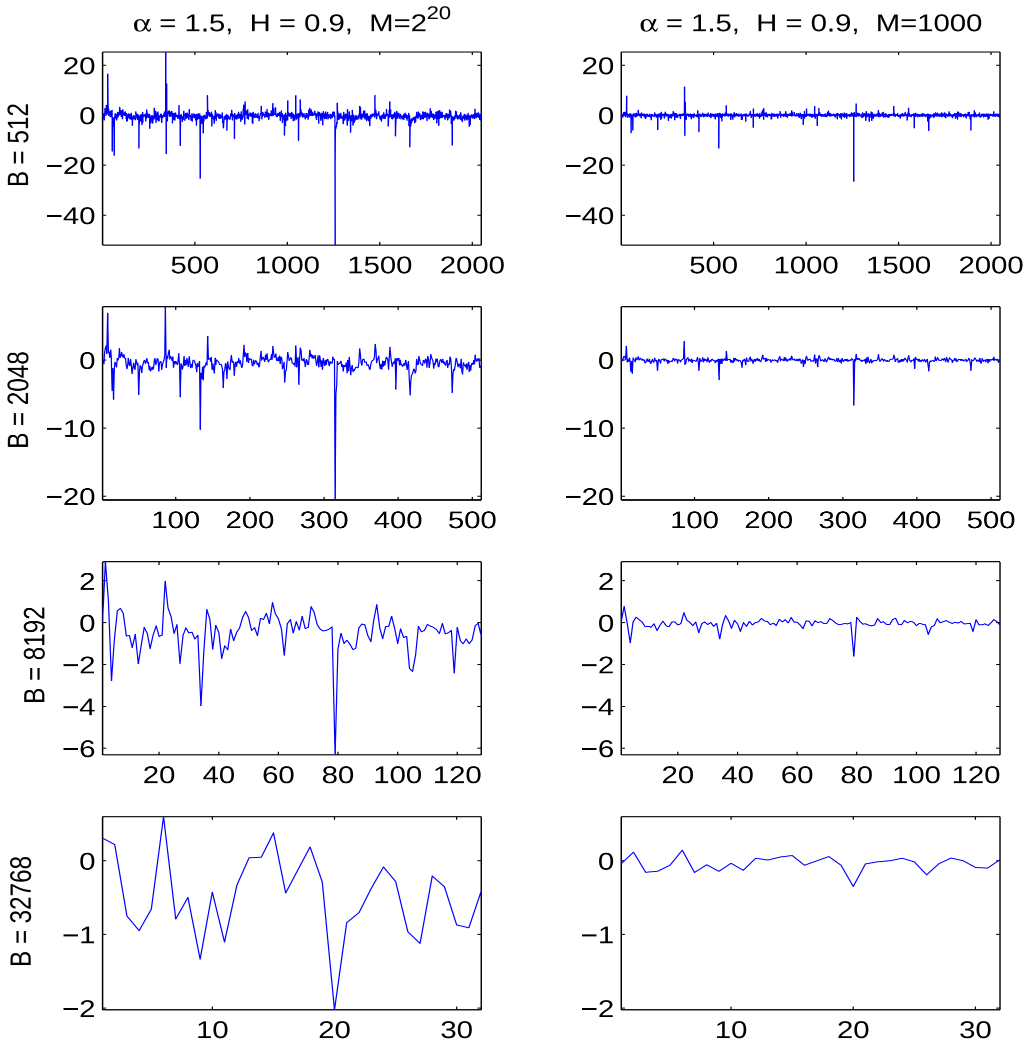

Figure 5: This figure contains graphs of block-wise averages $Y_M(k;B) := B^{-1}\sum_{j=1}^{B} Y_M(j+kB)$, $k = 0, 1, \ldots, [N/B]-1$ for single paths of simulated FARIMA$(0,d,0)$, $d = H - 1/\alpha$, time series of length $N = 2^{20} = 1,048,576$. The first column of plots corresponds to a path with truncation parameter $M = 2^{20}$, whereas the plots in the second column correspond to a path with $M = 1000$. Observe that the size of the blocks B is equal for plots in the same row and increases from the top towards the bottom rows. Note also that the vertical scales for plots in the same row are equal. We obtained the two paths by using the same sequence of iid SαS noise. We generated $M+N = 2^{21}$ such variables. In the case when $M = 1000$, we set $N = 2^{21}-M$ and truncated the time series to a length of 2^{20} and in the case $M = 2^{20}$ we set $N = 2^{20}$. Observe that the plots in the right column even-out more rapidly than the plots in the left column as the block-size B grows. This indicates that the path generated by using $M = 1000$ is rather weakly dependent and is not as good approximation to a FARIMA time series as the path with $M = 2^{20}$.

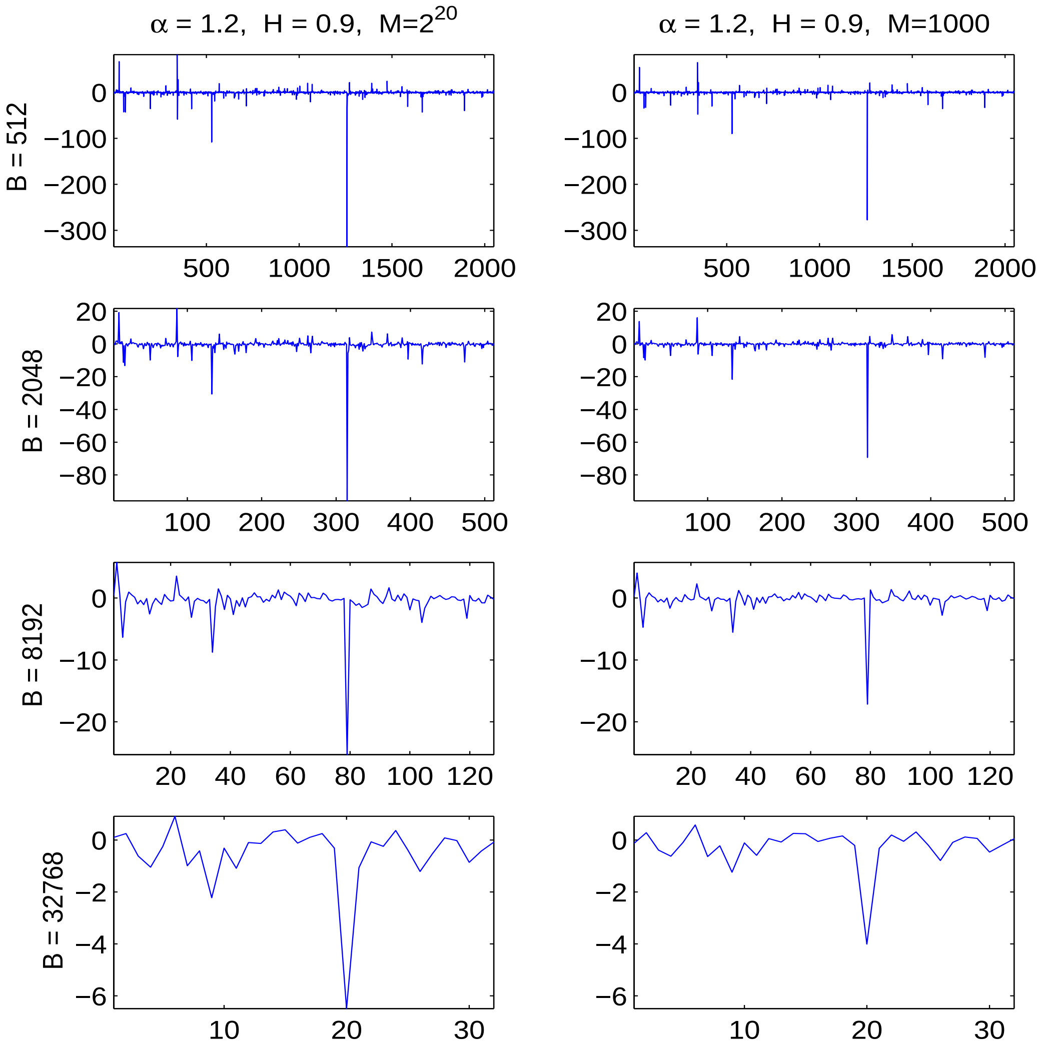

Figure 6: This figure is obtained as Figure 5, that is, the plots on the left-hand side display block-wise averages of simulated FARIMA$(0, d, 0)$, $d = H - 1/\alpha$, time series with $M = 2^{20} = 1,048,576$ and length $N = 2^{20}$. The plots on the right-hand side correspond to $M = 1000$. Here, however, the index of stability α is equal to 1.2. Thus the time series have heavier tails than the time series on Figure 6. (Note also the difference between the vertical scales on these two figures.) All four paths involved in Figures 5 and 6 were obtained by using the the same seed for the random number generator. Here in contrast with Figure 5 the left- and right-hand side plots are rather similar and one cannot conclude that the value $M = 1000$ is insufficient to obtain a good approximation to a FARIMA time series (see also Figure 7 and related comments).

dyadic scale in the data sets. The horizontal axes correspond to different choices of the smallest dyadic scale j_1.

Observe that as j_1 grows the estimators become less biased but their sample variances increase. This effect can be explained as follows. Firstly, by using larger scales one reduces the influence of the short-term dependence structure of the FARIMA time series on the estimators of the Hurst parameter (see also Figure 2 in Stoev and Taqqu (2003*a*) and the discussion therein). Secondly, the larger the scales the fewer wavelet coefficients are available. Thus estimators based on larger scales have greater sample variance than estimators based on lower scales.

Now let us compare the simulated FARIMA time series in the cases when the truncation parameter M is equal to 2^{20} and 1000, respectively. In the cases when $M = 2^{20}$, we see on Figures 7 and 8 that the estimators $\widehat{H}_{WT}$ tend to "stabilize" around the true value of the Hurst parameter H when $j_1 \geq 7$. On the other hand, in the cases when $M = 1000$, the bias of the estimators continues to fluctuate as j_1 grows. Moreover, when $M = 1000$, the estimators $\widehat{H}_{WT}$ tend to under-estimate the true value of H, for $j_1 \geq 8$. This observation can be explained by the fact that the simulated FARIMA time series Y_M, with $M = 1000$ are more weakly dependent than the time series generated with $M = 2^{20}$ and thus the former have effectively *lower* Hurst parameter.

Recall that the results displayed in Figure 6 were inconclusive. In contrast, Figure 7 clearly indicates that a value of $M = 1000$ is *not* sufficient for generating precise approximations to FARIMA time series of length $N = 2^{20}$. Furthermore, Figures 7 (where $\alpha = 1.2$) and 8 (where $\alpha = 1.5$) also suggest that the choice $M = N = 2^{20}$ provides relatively good approximations the FARIMA time series. Additional details about these figures are given in the remarks below.

Remarks

1. Because the parameters $\phi = 0.9$ and $\theta = 0.1$ of FARIMA$(1, d, 1)$ affect the lower scales, a significant bias appears at low values of j_1. This phenomenon is analyzed in detail in Stoev and Taqqu (2003*a*).

2. The weak dependence of the time series with $M = 1000$ becomes apparent as the lowest estimation scale 2^{j_1} grows. In fact the estimators of H seem to stabilize around the value $1/\alpha$, since the confidence intervals on Figures 7 and 8 eventually center around $1/1.2 - 0.9 \approx -0.0667$ and $1/1.5 - 0.9 \approx -0.2333$, respectively (when $j_1 \geq 13$).

3. Consider Figures 7 and 8 in the cases when $M = 1000$. For relatively low values of j_1 the estimated values of the Hurst parameter H appear to be very close to the true values of $H = 0.9$. More precisely, when j_1 is in the range $4 \leq j_1 \leq 8$ in the cases of FARIMA$(0, d, 0)$ and in the range $7 \leq j_1 \leq 8$ in the cases of FARIMA$(1, d, 1)$ with $\phi = 0.9$ and $\theta = 0.1$, the corresponding confidence intervals for the bias are very close to zero.

   This can be explained as follows. Let us focus on FARIMA$(0, d, 0)$, where we do not have to deal with the bias arising from the parameters ϕ and θ. Firstly, the dyadic scales 2^j of order from 2^4 to 2^8 are relatively large and the corresponding wavelet coefficients $D_{WT}(2^j, k)$ of the FARIMA time series scale very well, that is, they behave approximately like wavelet coefficients of an LFSM process. (Recall that the FARIMA time series considered here are asymptoticaly self-similar – see (3.5) above and Stoev and Taqqu (2003*a*) for more details.) Secondly, the truncation parameter $M = 1000 \approx 2^{10}$ is greater than the dyadic

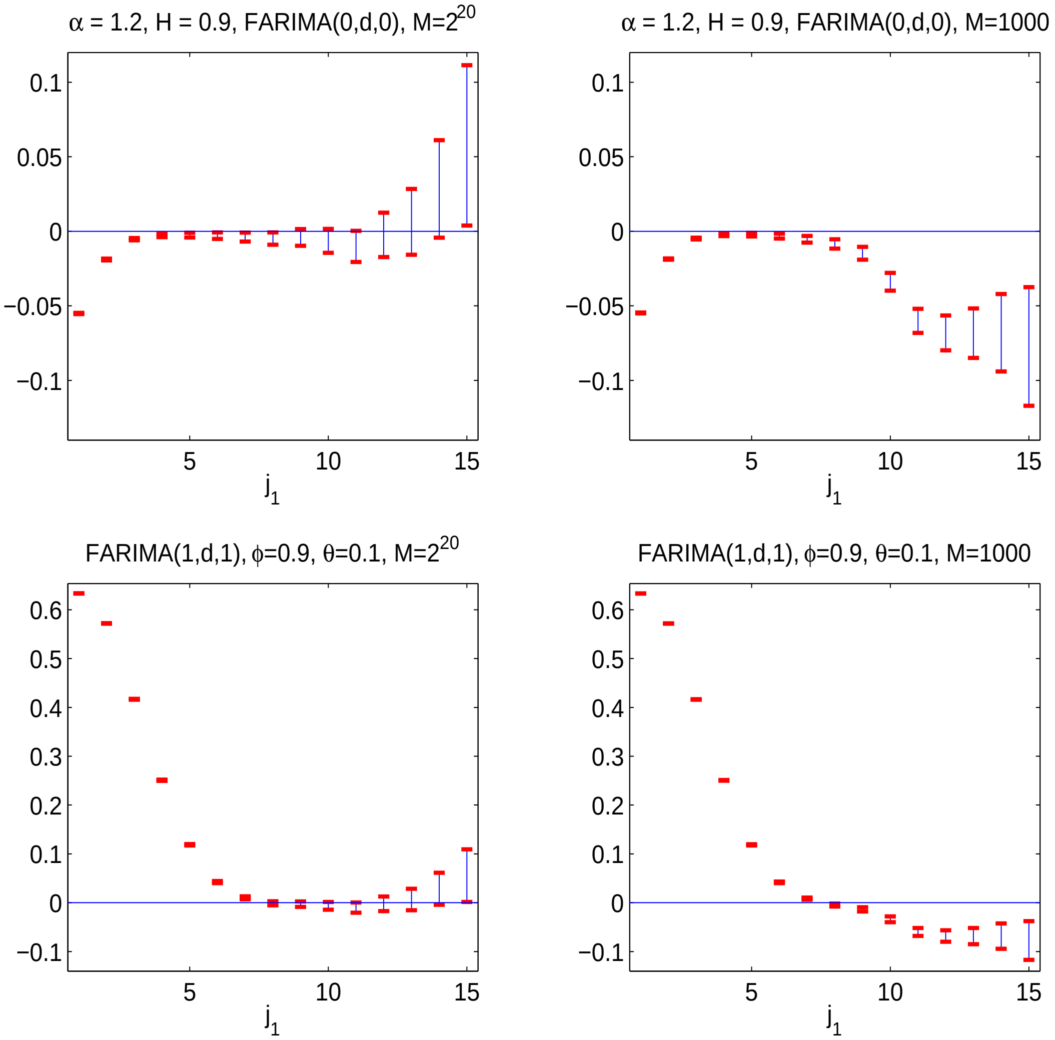

Figure 7: We display here approximate 95% confidence intervals for the bias $\mathbb{E}(\widehat{H}_{WT}) - H$ of the log-type wavelet estimator $\widehat{H}_{WT}$ (see (3.3)). We computed these confidence intervals by using normal approximation, as in (3.4), from samples of $n = 256$ estimates $\widehat{H}_{WT}$, obtained from n independently simulated paths of FARIMA time series. For the plots in the first column we used simulated FARIMA time series of length $N = 2^{20}$ and with truncation parameter $M = 2^{20}$. For the plots in the second column we used parameters $M = 1000$, $N = 2^{21} - M$ and then we truncated the time series to a length of 2^{20}. The vertical scales of the plots in the same rows are equal. The horizontal axis of the plots is j_1. The wavelet estimators $\widehat{H}_{WT}(j_1)$ are based on the dyadic scales $2^{j_1}, 2^{j_1+1}, \dots, 2^{j_2}$, where $j_2 \leq 20$ is the maximal available octave in the data. When $M = 1000$, the ranges $4 \leq j_1 \leq 8$ and $7 \leq j_1 \leq 8$ work well in the cases of FARIMA$(0, d, 0)$ and FARIMA$(1, d, 1)$, respectively, but when $M = 2^{20}$, j_1 can take higher values.

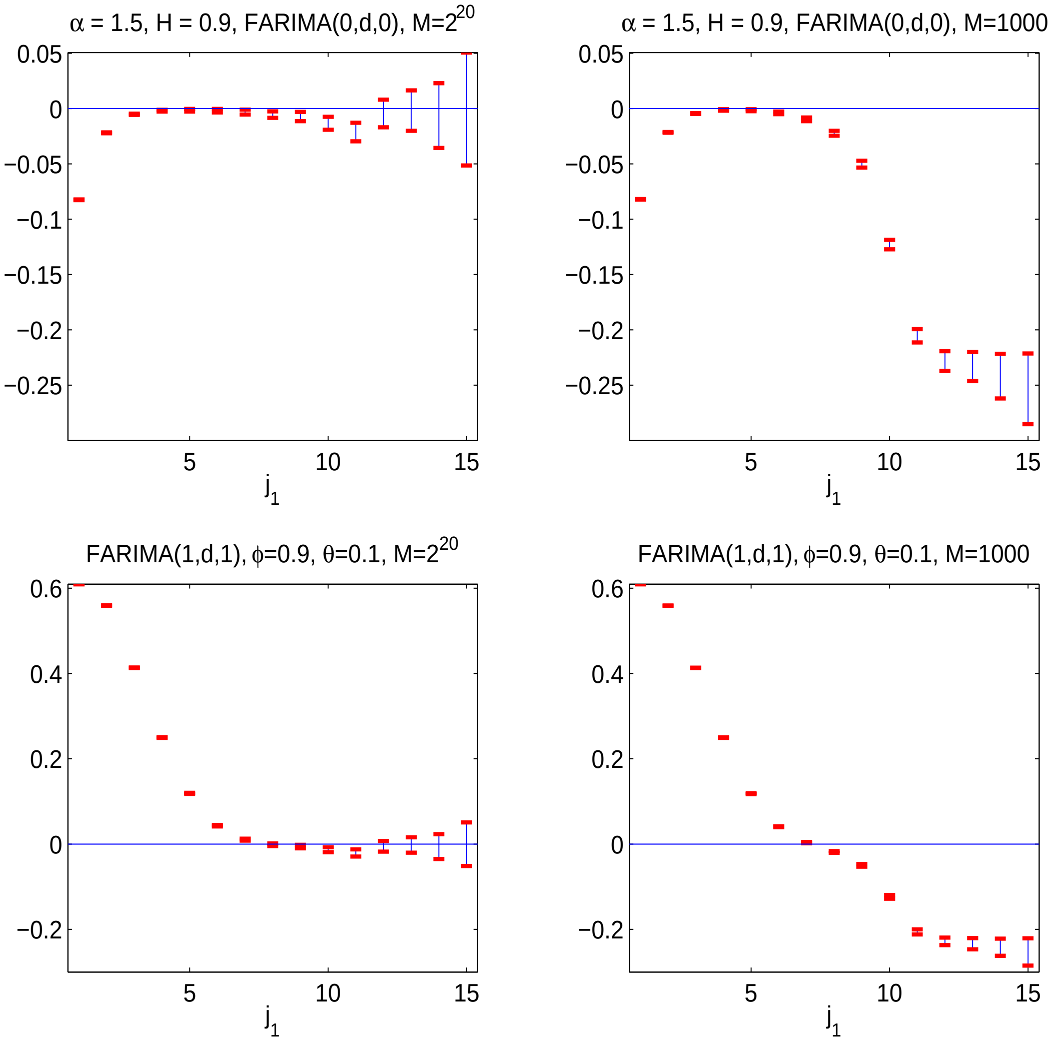

Figure 8: This figure is obtained as Figure 7, now however the simulated FARIMA time series involve SαS innovations with $\alpha = 1.5$. Observe that in this case the widths of the confidence intervals are comparable to the corresponding confidence intervals in Figure 7.

scales 2^4 to 2^8 and hence the approximation is good on these scales. Therefore the resulting estimators work well in the range $4 \leq j_1 \leq 8$. In the case of FARIMA$(1, d, 1)$ with $\phi = 0.9$ and $\theta = 0.1$, the relevant range of scales is smaller because, as indicated above, bias is present at the lower range of the scale.

4. When $M = 2^{20}$, the estimators are good when $j_1 \geq 4$ in the case of FARIMA$(0, d, 0)$ and $j_1 \geq 7$ in the case of FARIMA$(1, d, 1)$, a fact, consistent with the explanation above. Nevertheless, very large values of j_1 should be avoided, because of the corresponding large variances. In practice, when estimating the Hurst parameter H, one should use the smallest possible, but adequate j_1.

5. The time it takes to simulate one path of a FARIMA$(0, d, 0)$ time series with parameters $M = N = 2^{20} = 1,048,576$ on a dedicated PC Pentium II (Deschutes) with CPU frequency 447.701 MHz, $770,084$ kilobytes of RAM and Red Hat Linux OS version 7.1 is *approximately* 36.59 seconds. This is in line with what we observed for LFSM (Remark 2 in Section 3.1), where one had twice as many operations ($m(M + N) = 2^{22}$) which took twice as long to perform.

3.3 Conclusion

We summarise here the results from Sections 3.1 and 3.2.

In the case of LFSM, the choice of the discretization parameters m and M, and especially, that of m is rather *delicate*. This choice depends on the values of the parameters H and α. When $0 < \alpha \leq 1$, one should use significantly large values of m, especially in the cases for small values of H. On the other hand, when $1 < \alpha \leq 2$, the choice of large values of m seems to be inappropriate when the parameter M is small, especially in the cases of negative dependence ($0 < H < 1/\alpha$). The long-range dependence cases ($1/\alpha < H < 1$) seem to be more robust with respect to the choice of the mesh-parameter m.

When simulating FARIMA time series, one should use *large values* for the truncation parameter M in order to represent well the long-term dependence behavior of the time series, particularly when the Hurst parameter $H = d + 1/\alpha \in (1/\alpha, 1)$ is close to 1. Choices of M greater or comparable to the length N of the simulated time series work well in most cases.

The FFT-based algorithms presented in Sections 2.1 and 2.2 are computationally efficient. For example, the time required to generate an acceptable LFSM sequence of length $N \approx 10,000$ is about 1 minute and to simulate a FARIMA$(0, d, 0)$ sequence of length $N \approx 10^6$ is about 30 seconds. One can use these algorithms successfully in practice to generate significantly long paths, which represent well the dependence structures of the LFSM or FARIMA processes.

4 MATLAB code and illustration of how to use it

We provide here the MATLAB code for the simulation algorithms described in Section 2 and illustrate how to use it. The function `fftlfsn(H,alpha,m,M,C,N,n)` is used to generate linear fractional stable noise (with scaling constant `C`). For example, the commands

```
H = 0.2; alpha =1.5; m = 256; M = 6000; N = 2^14 - M;
y = fftlfsn(H,alpha,m,M,1,N,1);
x = cumsum(y);
```

generate a simulated path y of length N of linear fractional stable noise and a path x of LFSM.

The function **fftfarima(alpha,d,n,Ph,Th,M,N)** is used to generate FARIMA (where the vectors Ph and Th contain the coefficients of the polynomials Φ and Θ, characterizing the ARMA part of the time series). For example, the commands

```
alpha = 1.2; d = 0.1; Ph = []; Th = []; M = 2^20; N = M;
y0 = fftfarima(alpha,d,1,Ph,Th,M,N);
Ph = [0.9]; Th = [0.2];
y1 = fftfarima(alpha,d,1,Ph,Th,M,N);
```

generate a sequence y0 of FARIMA$(0, d, 0)$ with $d = 0.1$ (ie $H = d + 1/\alpha \approx 0.93$) and a sequence y1 of FARIMA$(1, d, 1)$ with $\Phi(z) = 1 - 0.9z$, $\Theta(z) = 1 - 0.1z$ and $d = 0.1$.

Each MATLAB function is preceded by comments, describing its parameters. For brevity, we do not include the code for the functions **rstab.m**, **d2.m** and **tan2.m**, used to generate SαS variables. This code can be obtained on request from the authors of the present paper (see also Ch. 1.7 in Samorodnitsky and Taqqu (1994).) The software for estimation of the self-similarity parameter H can also be obtained from the authors.

- *Code for generating linear fractional stable noise*

```
function y = fftlfsn(H,alpha,m,M,C,N,n);
%
% This function generates LFSN (Linear Fractional Stable Noise) Y_k
%
%  Y_k = K(H,alpha,C) \int_R (s)_+^d - (s-1)_+^d M(ds),  d = H-1/\alpha
%
% by using a Riemann sum approximation of its stochastic integral
% representation.  The constant K(H,alpha,C) is chosen so that
% ||Y(k)||_alpha = C, where C is an input parameter.
%
% input:
%     H <- the self-similarity parameter.
% alpha <- the tail-exponent of the stable distribution
%     m <- 1/m = the mesh-size
%     M <- the kernel cut-off parameter
%     C <- the scale parameter
%     N <- size of the sample
%     n <- number of independent paths to be simulated
%
% output:
%     y <- (nxN) matrix the rows give independent paths.
%
%****************************************************************
%* NOTES: 1. When the parameters m and (M+N) are integer powers of 2,
%*       the fast Fourier transform algorithm is used.
%*        2. Symmetric alpha-stable variates are generated by using
%*       rstab.m (with d2.m and tan2.m), which is a MATLAB version
%*       of the classical RSTAB.FOR Chambers, Mallows and Stuck, JASA
%*       This code can be obtained from Dimitar L. Vandev,
%*       vandev@fmi.uni-sofia.bg or sstoev@math.bu.edu.
%****************************************************************
```

```
% Written by Stilian Stoev 05.06.2002, sstoev@math.bu.edu
%
% Usage:
%     y = fftlfsn(H,alpha,m,M,C,N,n)
%
 mh = 1/m;
 d  = H-1/alpha;
 t0 = [mh:mh:1];
 t1 = [1+mh:mh:M];
 A  = mh^(1/alpha)*[t0.^d, t1.^d-(t1-1).^d];
 C  = C*(sum(abs(A).^alpha)^(-1/alpha));
 A  = C*A;
 Na = m*(M+N);
 A  = fft(A,Na);
 y  = [];
 for i=1:n,
  if alpha<2,
    Z = rstab(alpha,0,Na)';
  elseif alpha==2,
    Z = randn(1,Na);
  end;
  Z = fft(Z,Na);
  w = real(ifft(Z.*A,Na));
  y = [y; w(1:m:N*m)];
 end;
```

- *Code for generating FARIMA time series with SαS innovations*

```
function Y = fftfarima(alpha,d,n,Ph,Th,M,N);
%
% Generates fractional ARIMA time series with stable symmetric innovations
% by using a truncated version of the moving average representation developed
% by Kokoszka & Taqqu, Stoch. Processes and Their Appl. 60, 19-47, 1995.
%
% input:
%   alpha <- the tail-exponent of the innovations (1<alpha)
%   d     <- the fractional differencing exponent (0<d<1-1/alpha)
%   n     <- a number of independent paths to be generated
%   Ph    <- a vector of the AR (autoregressive) coefficients
%            Ph = 0 or [] means "no AR part"
%   Th    <- a vector of the MA (moving average) coefficients
%            Th = 0 or [] means "no MA part"
%   M     <- the cut-off parameter (recommended M > N)
%   N     <- the desired length of the moving average
%
% output:
%   Y     <- a (nxN) matrix each row corresponds to a generated path
%            of FARIMA(p,d,q). The paths are "independent".
% ***************************************************************************
% *NOTEs: 1. When (M+N) is an integer power of 2, the FFT algorithm is used.
% *       2. Calls: fracdiff.m
% *       3. Phi(z)  = 1 - phi(1)z - phi(2)z^2 - ... - phi(p)z^p
% *          Theta(z)= 1 - theta(1)z- theta(2)z^2 - ... - theta(q)z^q,
% *          where Ph = [phi(1) .. phi(p)]; Th = [theta(1) .. theta(q)]
% ***************************************************************************
%  Written by Stilian Stoev, 05.05.2002, sstoev@math.bu.edu.
%
% usage:
%   Y = fftfarima(alpha,d,n,Ph,Th,M,N);
%
 c = fracdiff(d,M);
 c = filter([1, -Th],[1, -Ph],c);
 c = fft(c,M+N);
 Y = [];
 for i=1:n,
  if alpha<2,
    Z = rstab(alpha,0,M+N)';
  elseif alpha==2
    Z = randn(1,M+N);
  end;
  Z = fft(Z,M+N);
  y = real(ifft(c.*Z,M+N));
  Y = [Y; y(1:N)];
 end;
function b = fracdiff(d,N);
%
% This function computes the moving average coefficients b(j)
% of the MA representation of a FARIMA(0,d,0). We use formula
% (1.3) on p. 318 of "Can one use the Durbin-Levinson
% algorithm to generate infinite variance fractional ARIMA
% time series?" by Kokoszka and Taqqu, J. Time Ser. Anal.,
% Vol. 22, No. 3, 2001:
%
%  b(0) = 1, b(j) = Gamma(j+d)/(Gamma(d)*Gamma(j+1))
%
% input:
%   d <- the fractional differencing exponent (must be positive)
%   N <- number of coefficients b(j) to be computed
%
% output:
%   b <- N-vector b(j+1) corresponds to b(j), above, j=0,1,...,N-1.
%
% usage:
%   b = fracdiff(d,N);
%
 j1 = [1:N-1];
 jd = (j1-1+d);
 b  = [1, cumprod(jd./j1)];
```

References

Adler, R., Feldman, R. & Taqqu, M. S., eds (1998), *A Practical Guide to Heavy Tails: Statistical Techniques and Applications*, Birkhäuser, Boston.

Astrauskas, A. (1983), 'Limit theorems for sums of linearly generated random variables', *Lithuanian Mathematical Journal* **23**(2), 127–134.

Bardet, J.-M., Lang, G., Oppenheim, G., Philippe, A., Stoev, S. & Taqqu, M. (2003), Generators of long-range dependence processes: A survey, *in* P. Doukhan, G. Oppenheim & M. S. Taqqu, eds, 'Theory and Applications of Long-range Dependence', Birkhäuser, pp. 557–577.

Cambanis, S., Hardin, Jr., C. D. & Weron, A. (1984), Ergodic properties of stationary stable processes, Technical Report 59, Center for Stochastic Processes at the University of North Carolina, Chapel Hill.

Chambers, J. M., Mallows, C. & Stuck, B. W. (1976), 'A method for simulating stable random variables', *Journal of the American Statistical Association* **71**(354), 340–344. Theory and Methods Section.

Chambers, J. M., Mallows, C. & Stuck, B. W. (1987), 'Correction to: "A method for simulating stable random variables"', *Journal of the American Statistical Association* **82**(398), 704.

Crovella, M. E. & Bestavros, A. (1996), Self-similarity in World Wide Web traffic: evidence and possible causes, *in* 'Proceedings of the 1996 ACM SIGMETRICS. International Conference on Measurement and Modeling of Complex Systems', pp. 160–169.

Daubechies, I. (1992), *Ten Lectures on Wavelets*, SIAM Philadelphia. CBMS-NSF series, Volume 61.

Doukhan, P., Oppenheim, G. & Taqqu, M. S., eds (2003), *Theory and Applications of Long-range Dependence*, Birkhäuser, Boston.

Granger, C. W. J. & Joyeux, R. (1980), 'An introduction to long-memory time series and fractional differencing', *Journal of Time Series Analysis* **1**, 15–30.

Kokoszka, P. S. & Taqqu, M. S. (1995), 'Fractional ARIMA with stable innovations', *Stochastic Processes and their Applications* **60**, 19–47.

Kokoszka, P. S. & Taqqu, M. S. (1996), 'Infinite variance stable moving averages with long memory', *Journal of Econometrics* **73**, 79–99.

Leland, W. E., Taqqu, M. S., Willinger, W. & Wilson, D. V. (1994), 'On the self-similar nature of Ethernet traffic (Extended version)', *IEEE/ACM Transactions on Networking* **2**, 1–15.

Levy, J. B. & Taqqu, M. S. (2000), 'Renewal reward processes with heavy-tailed interrenewal times and heavy-tailed rewards', *Bernoulli* **6**(1), 23–44.

Mallat, S. (1998), *A Wavelet Tour of Signal Processing*, Academic Press, Boston.

Mikosch, T., Resnick, S., Rootzén, H. & Stegeman, A. (2002), 'Is network traffic approximated by stable Lévy motion or fractional Brownian motion?', *The Annals of Applied Probability* **12**(1), 23–68.

Park, K. & Willinger, W., eds (2000), *Self-Similar Network Traffic and Performance Evaluation*, J. Wiley & Sons, Inc., New York.

Paxon, V. & Floyd, S. (1994), Wide-area traffic: The failure of Poisson modeling, *in* 'Proceedings of the ACM Sigcomm '94', London, UK, pp. 257–268.

Pipiras, V., Taqqu, M. S. & Levy, J. B. (2000), Slow, fast and arbitrary growth conditions for renewal reward processes when both the renewals and the rewards are heavy-tailed, Preprint.

Press, W., Teukolsky, S., Vetterling, W. & Flannery, B., eds (1992), *Numerical Recipes in C*, 2nd edn, Cambridge University Press, Cambridge, UK.

Samorodnitsky, G. & Taqqu, M. S. (1994), *Stable Non-Gaussian Processes: Stochastic Models with Infinite Variance*, Chapman and Hall, New York, London.

Stoev, S. & Taqqu, M. S. (2003*a*), 'Asymptotic self-similarity and wavelet estimation for long-range dependent FARIMA time series with stable innovations', *Journal of Time Series Analysis.* To appear (Preprint 2002).

Stoev, S. & Taqqu, M. S. (2003*b*), Wavelet estimation for the Hurst parameter in stable processes, *in* G. Rangarajan & M. Ding, eds, 'Processes with Long-Range Correlations: Theory and Applications', Springer Verlag, Berlin, pp. 61–87. Lecture Notes in Physics 621.

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

Taqqu, M. S., Willinger, W. & Sherman, R. (1997), 'Proof of a fundamental result in self-similar traffic modeling', *Computer Communications Review* **27**(2), 5–23.

Willinger, W., Paxson, V. & Taqqu, M. S. (1998), Self-similarity and heavy tails: structural modeling of network traffic, *in* R. Adler, R. Feldman & M. S. Taqqu, eds, 'A Practical Guide to Heavy Tails: Statistical Techniques and Applications', Birkhäuser, Boston, pp. 27–53.