Generators of long-range dependent processes: A survey

Jean-Marc Bardet

Gabriel Lang

Georges Oppenheim,

Anne Philippe

Murad S. Taqqu

ABSTRACT We describe methods for generating discretized sample paths of long-range dependent processes such as fractional Brownian motion, its increment fractional Gaussian noise and FARIMA processes.

1 Introduction

We shall describe practical methods to generate discrete time sample paths of length N of a process with long-range dependence. These simulated paths can be used, for example, in Monte Carlo procedures to evaluate the small sample size properties of various statistics. While several methods have been developed in the last 20 years, only a few articles discuss and compare generators of long-range dependent processes. Beran [4] and Jennane *et. al.* [20] present a number of generators. Voss [42], Lowen [26], Crouse and Baraniuk [11] and Coeurjolly [9] discuss some generators for fractional Brownian motion. Corsini and Saletti [10] suggest using an ARMA approximation to generate a Gaussian stationary process with a power spectrum over a finite frequency band. Hauser and Hörmann [19] present a nice overview of three methods of generation of Gaussian time series, the Cholesky decomposition, the Durbin-Levinson algorithm and Fourier transform methods, with particular emphasis on long-range dependent processes.

Denote by X the process to be simulated and by $\widetilde{X}$ the process resulting from the simulation. We assume that the random number generator used is

AMS Subject classification: Primary 65C05, 60G18; secondary 62M10.
Keywords and phrases: Monte Carlo, fractional Brownian motion, FARIMA

close to perfect (in terms of distribution and independence). We start with some general remarks.

1. If the process X is Gaussian and its covariance function known, one generates a sequence of Gaussian variables whose covariance is the given covariance. This is usually done by evaluating the square root of a matrix, using a Choleski decomposition, for example, or some other technique. If the square root computation step is perfectly done, the generated process has the same distribution as the original one.

2. There often exists several representations of the process. The classical ones involve a time or spectral domain stochastic integral. If the process is defined in continuous time, one can approximate the integral by a sum of finitely many terms and simulate this approximation. If a discrete time process is represented by a moving average with infinitely many terms, it will have to be truncated and the generated process $\widetilde{X}$ only replicates the truncated one. For processes with long-range dependence, however, difficulties arise from the irregularities of the kernel in the integral representation or from the fact that the coefficients of the moving average converge slowly to zero.

   One can also use alternative representations. We present one which is based on a stochastic first order linear differential equation with a parameter defined through a random measure. An approximation of the measure by a discrete one whose support is a finite set, produces the approximation $\widetilde{X}$. The resulting $\widetilde{X}_t$ approximates X_t in mean square.

3. When X is the limit in distribution of a sequence $X(n)$ of processes, one can approximate X by $X(n)$. The following are natural questions. Are the $X(n)$ processes easy to simulate? In what sense does the convergence hold? How large must n be for the approximation to be adequate?

There is no simply way to validate the generators. One should test, a priori, several properties, for example, the matching of the marginal distributions of $\widetilde{X}$ to those of X, the stationarity of $\widetilde{X}$ or of its increments (if X has these properties), and the matching of the theoretical and simulated structures of dependence. We shall use only two criteria, and only for some of the methods. The first criterion involves the parametric Whittle estimator of the long-range dependence parameter, the second one is a test of fit for the spectral density of the process.

We start with a general presentation of the generators. For each generator, we give a brief explanation of the method and a description of the algorithm. We specify the models that the method is able to simulate. Some of the methods are not specific to long-range dependence; others use, for example, the self-similarity of fractional Brownian motion. We then apply these methods and analyze the resulting simulations.

2 Description of the methods

This section explains the main procedures for generating long-range dependent processes. We shall, in most cases, generate the sample path of an equispaced discrete stationary long-range dependent process. We denote by $X = \{X_t, t \in T\}$ the original process (with $T \subset \mathbb{R}$). Our goal is to generate a "realization" or "sample path" $(X_1(\omega), .., X_N(\omega))$, of the process X with $\omega \in \Omega$, where N denotes the length of the sample path.

The methods described below can be used to generate sample paths of a Gaussian random process (or field) defined by its autocovariance function. In other words, given a covariance matrix Σ of size $N \times N$, we want to generate a Gaussian random vector of length N, with covariance Σ. These methods are exact, the quality of the simulation depending only on the precision of the computations and that of the Gaussian white noise generator. On the other hand, some of these methods are not iterative: the sample path has a fixed length N and, if N increases, one has to start all over again. These methods also require an explicit form of the autocovariance function.

2.1 Basic Choleski

This method involves computing the square root of the positive matrix Σ, using the LU (lower-upper triangular) decomposition. If A is such that $A^t A = \Sigma$, then A is called the square root of the matrix Σ and if ξ is a Gaussian white noise vector of length N, then $X = A\xi$ is a centered Gaussian vector with covariance matrix Σ (the superscript t denotes a transpose).

Complexity of the algorithm: Algorithms based on the LU decomposition are widespread, but they are relatively slow, since their complexity is of order $\mathcal{O}(N^3)$.

Simulated models and possible extensions: This method allows the generation of any non-degenerate Gaussian vector if its covariance function is known.

2.2 Durbin-Levinson

The Durbin-Levinson algorithm is a conditional distribution method. Its advantage stems from the fact that for a Gaussian stationary series, the conditional means and variances can be obtained through Yule-Walker-type equations, which are recursive. The method is described in detail in Brockwell and Davis [5].

Concrete simulation procedure: To generate a Gaussian time series $X_i,\ i=1,\dots,N$ with mean zero and autocovariance function γ, using the Durbin-Levinson algorithm, generate a sequence $\varepsilon_i,\ i\geq 1$ of independent and identically distributed $\mathcal{N}(0,1)$ random variables.

Then set $X_1 = \gamma(0)^{1/2}\varepsilon_1$ and suppose that $X_1, X_2, \dots, X_n$ have been obtained. Then

$$X_{n+1} = \phi_{n,1}X_n + \cdots + \phi_{n,n}X_1 + v_n^{1/2}\varepsilon_{n+1}. \tag{2.1}$$

The variances $v_i,\ i=1,\dots,n$ and the coefficients $\{\phi_{n,i},\ i=1,\cdots n\}$ are computed recursively. Set $v_0=\gamma(0)$ and $v_n = v_{n-1}\left(1-\phi_{n,n}\right)^2$. For $n\geq 1$,

$$\begin{aligned}\phi_{n,n} &= \Big[\gamma(n) - \sum_{j=1}^{n-1}\phi_{n-1,j}\gamma(n-j)\Big]v_{n-1}^{-1},\\ \phi_{n,i} &= \phi_{n-1,i} - \phi_{n,n}\phi_{n-1,n-i} \text{ for } i<n.\end{aligned}$$

Simulation results using this approach are analyzed in Taqqu, Teverovsky and Willinger [40] and Taqqu and Teverovsky [39].

Complexity of the algorithm: $\mathcal{O}(N^2)$

Possible extensions: The models are Gaussian, with an autocovariance function given in closed form. Kokoszka and Taqqu [22] discuss conditions under which the Durbin-Levinson algorithm can be used to generate infinite variance FARIMA sequences.

2.3 The circulant matrix embedding method

The method consists of embedding the covariance matrix $\Sigma = (r(i-j))_{1\leq i,j\leq N}$ in a non-negative definite matrix R of size $M\geq 2(N-1)$ which is a circulant (see Dietrich and Newsam [14], Wood and Chan [45], Beran [4], Davies and Harte [12]). Circulants are easy to diagonalize, by using the Discrete Fourier Transform.

Concrete simulation procedure: The algorithm uses the fast Fourier transform which bypasses matrix computations:

- Choose $M=2^p\geq 2(N-1)$. Define the M-vector

$$V = (r(0), r(1), \dots, r(M/2-1), r(M/2), r(M/2-1), \dots, r(1)). \tag{2.2}$$

- Compute $W = (w_1, \cdots, w_M)$, the fast Fourier transform of V. Check if all the coordinates are either positive or zero. If not, go back to the first step and choose a larger value of p.

- Simulate a random complex M-vector $Z = X + iY$, where the coordinates of X and Y are independent centered Gaussian variables and have variances equal to $1/\sqrt{2}$. Then define the vector $U = (U_k)_{k=1,\dots,M}$ by $U_k = \sqrt{W_k} Z_k$, $k = 1, \dots, M$.

- Compute X as the Inverse fast Fourier transform of the complex vector U. Extract the real part of the first N coordinates of X to obtain the simulated sample path.

The simulated path has the correct covariances up to a multiplicative factor of M. Note that this algorithm can be easily implemented to simulate the FARIMA[0,d,0] processes since the covariance function r is known explicitly and is given by

$$r(k) = \sigma_\epsilon^2 \frac{(-1)^k \Gamma(1-2d)}{\Gamma(k-d+1)\Gamma(1-k-d)}.$$

For the FARIMA[p,d,q] model, it is necessary to approximate the covariance function using the Sowell's method (see Sowell [36]). An alternative is to apply a program for the simulation of an ARMA process, using a FARIMA[0,d,0] process as innovations.

The simulation of processes X with stationary increments follows the same approach. First a stationary process Y with the same distribution as the increments of X is generated. Then the cumulative sums of the components of Y are computed. They provide the desired simulated path of X (see Wood and Chan [45]). For instance, using the covariance function

$$r(k) = \frac{1}{2} \frac{\sigma^2}{N^{2H}} \left[(k+1)^{2H} - 2k^{2H} + (k-1)^{2H} \right],$$

we simulate a fractional Gaussian noise (fGn) Y which is the increment process of a fBm X. Crouse and Baraniuk [11] show that one can apply the circulant method to fractional Gaussian noise using any $M \geq 2(N-1)$ because the coordinates of W are always positive.

Implementation: In the different examples studied (fGn, FARIMA models for $N = 100,\ 1000$ and 10000), the value of M chosen to ensure that all the coordinates of W are either positive or null is $M = 2^p$, where p is the smallest integer such that $2^p \geq 2(N-1)$. The choice $M = 2^p$ enables us to use the fast Fourier transform. However, no theoretical result exists to prove that such a choice works for all processes. We then extract the first N values of the generated series.

Complexity of the algorithm: The Discrete Fourier Transform is computed using the fast Fourier transform algorithm which is faster than the LU decomposition. The complexity of the algorithm is $\mathcal{O}(M\log(M))$. The procedure fails if W has a negative coordinate. In this case, as indicated, one may try a larger M, or a smaller discretization step in the case of fractional Brownian motion.

Simulated models and possible extensions: The models are Gaussian, with an autocovariance function given in closed form. There are no general theoretical results, stating that there exists an M for which the covariance function r can be embedded into a M-vector such that the resulting circulant matrix is non-negative definite. If the covariance function r has a finite support, or if $r(k),\ k=0,\dots,N$ is non-negative, decreasing and convex, then M can be chosen to be $2(N-1)$ (see Dietrich and Newsam [14]).

2.4 Lowen's circulant method

The following exact technique for generating fractional Brownian motion $B_H(n),\ n=0,\dots,N$ with $0<H<1/2$ is a given in Lowen [26]*. The mean of B_H is zero and its covariance is

$$EB_H(n)B_H(m)=C_N[n^{2H}+m^{2H}-|n-m|^{2H}],\quad 0\le n,\ m\le N. \tag{2.3}$$

We will choose $C_N=2^{-1}N^{-2H}$ so that $EB_H^2(N)=1$, although the constant C_N can be easily replaced by any other one since, here, N is a fixed number. If one wanted to generate a standard fractional Brownian motion on $[0,1]$ sampled at fixed intervals of length $1/N$, namely $B_H^{(st)}(t),\ t=0,1/N,2/N,\dots,1$, then it suffices to set $B_H^{(st)}(n/N)=N^{-H}B_H(n)$ since the choice of C_N given above ensures that $E(B_H^{(st)}(1))^2=1$.

Start with a function

$$R(n)=\begin{cases}2^{-1}[1-(n/N)^{2H}] & \text{for } 0\le n\le N,\\ R(2N-n) & \text{for } N<n\le 2N.\end{cases} \tag{2.4}$$

It is easy to see that the function R is symmetric around $n=N$, reaches a minimum at this point and its second difference satisfies $\nabla^2 R(n)\ge 0$. Extend now R periodically to all $n\in\mathbb{N}$. Since its finite Fourier transform $S(\nu)=\sum_{n=0}^{2N-1}\exp\{in\nu\pi/N\}R(n)$ will be real, symmetric and nonnegative for all $\nu\in\mathbb{N}$, S can serve as a discrete spectrum. Now generate

$$\widetilde{X'}(\nu)=\begin{cases}0 & \text{for } \nu=0,\\ \sqrt{S(\nu)/2}\ (G_1(\nu)+iG_2(\nu)) & \text{for } 1\le\nu\le N-1,\\ \sqrt{S(\nu)}\ \varepsilon & \text{for } \nu=N,\\ \widetilde{X}^*(2N-\nu) & \text{for } N+1\le\nu\le 2N-1.\end{cases} \tag{2.5}$$

*The version in [26] contains some minor errors.

In this expression, $*$ denotes the complex conjugate operation, ε, $G_1(\nu)$, $G_2(\nu)$, $\nu = 1, \dots, N-1$, are i.i.d. $\mathcal{N}(0,1)$. Then compute the inverse Fourier transform $X'(n) = (2N)^{-1} \sum_{\nu=0}^{2N-1} \exp\{-in\nu\pi/N\} \widetilde{X'}(\nu)$, $n = 0, \dots, 2N-1$. The sequence $X'(n)$, $n \in \mathbb{N}$ is a real Gaussian mean-zero, $2N$-periodic and stationary with covariance

$$EX'(n)X'(m) = \frac{1}{2N} R(n-m) - \frac{1}{(2N)^2} S(0). \tag{2.6}$$

In other words,

$$X'(n) = \frac{1}{\sqrt{2N}} (X(n) - \eta), \tag{2.7}$$

where X is the Gaussian sequence with covariance R and where η is a fixed mean-zero Gaussian random variable dependent on the sequence X. (In fact, $\eta = (1/2N) \sum_{n=0}^{2N-1} X(n)$).

Since the sequence $X(n)$, $n \in \mathbb{N}$, has the covariance $R(n)$, $n \in \mathbb{N}$, it is easy to see by using (2.6) that

$$B_H(n) = \sqrt{2N}(X'(n) - X'(0)), \quad 0 \le n \le N,$$

has the covariance (2.3) and hence is fractional Brownian motion.

Concrete simulation procedure: To generate fractional Brownian motion B_H with the covariance (2.3):

- Choose N of the form 2^p and generate $2N-1$ i.i.d. $\mathcal{N}(0,1)$ random variables ε, $G_1(\nu)$, $G_2(\nu)$, $\nu = 1, \dots, N-1$.
- Using the fast Fourier transform, compute $S(\nu)$ and define $\widetilde{X'}(\nu)$ by (2.5).
- Using the inverse fast Fourier transform, compute $X'(n)$.
- Set $B_H(n) = \sqrt{2N}(X'(n) - X'(0))$, $0 \le n \le N$.

Complexity of the algorithm: $\mathcal{O}(N \log N)$ if $N = 2^p$. Advantage: the method is fast and exact. Drawback: it is not recursive in N and it works only for $0 < H < 1/2$.

Implementation: Instead of choosing $N = 2^p$, we generated independent sequences with $N = 100$ and $N = 1000$.

2.5 Direct spectral method

Voss [42], Feder [15], Peitgen and Saupe [33] and Paxson [32] generate an approximation of fractional Gaussian noise (increments of fractional Brownian motion) by using the Discrete Fourier Transform without applying the circulant method. The strategy behind this method is to start with the spectral density of the process at Fourier frequencies, to simulate a random complex sequence with this spectral density and then use the inverse fast Fourier transform to obtain an approximation of the process at times $t = k/N$, $k = 0, 1, \ldots, N-1$. Although fast, there are two problems with this method : one is aliasing, the second is that the generated sequence is periodic i.e. the end of the sequence is highly correlated with the beginning (see Kasdin [21]). To alleviate these facts, one can generate a sequence that is several times longer than what one needs and extract a sequence of length N from it.

Concrete simulation procedure : To generate fractional Gaussian noise:

- Choose N an even integer. Define the vector of the Fourier frequencies $\Lambda = (\lambda_1, \ldots, \lambda_{N/2})$ where $\lambda_k = 2\pi k/N$ and compute the vector $F = (f_H(\lambda_1), \ldots, f_H(\lambda_{N/2}))$ where

$$f_H(\lambda) = (C_2(H))^{-2} 2(1 - \cos\lambda) \sum_{k\in\mathbb{Z}} |2\pi k + \lambda|^{-2H-1} \tag{2.8}$$

  and $(C_2(H))^{-2} = (1/2\pi)\sin(\pi H)\Gamma(2H+1)$. $f_H(\lambda)$ is the spectral density of fractional Gaussian noise (fGn).

- Generate $N/2$ i.i.d. exponential $\mathcal{E}(1)$ random variables $E_1, \cdots, E_{N/2}$ and $N/2$ i.i.d. uniform $\mathcal{U}[0,1]$ random variables $U_1, \cdots, U_{N/2}$.

- Compute $Z_k^* = \sqrt{F_k E_k} e^{2i\pi U_k}$ for $k = 1, \cdots, N/2$.

- Form the N-vector

$$Z = (0, Z_1^*, \cdots, Z_{(N/2)-1}^*, Z_{N/2}^*, \overline{Z_{(N/2)-1}^*}, \cdots, \overline{Z_1^*}).$$

- Compute the inverse fast Fourier transform of the complex Z to obtain the simulated sample path.

Complexity of the algorithm: $\mathcal{O}(N \log N)$ if N is of the form 2^p.

Implementation: We truncated the sum in (2.8) to $k \in \{-200, \ldots, 200\}$ and used $N = 100$ and $N = 1000$ to obtain time series of lenght N.

Extensions: The summation in (2.8) must be recomputed for each value of λ. Ledesma and Liu [24] show however that $\sum_{|k|\geq 3} |2\pi k+\lambda|^{-2H-1}$ can be well approximated by a linear function $p\lambda + q$. The coefficients p and q are given by infinite sums that can be safely truncated. This linear approximation reduces the running time in half: it takes about 400 seconds to generate 2 million data points for fGn using a Sun SPARC 20 workstation and matlab.

2.6 Discretization of the integral in time or spectral domain

Start with the representation of standard fractional Brownian motion with parameter $H \in (0,1)$ as a stochastic integral with respect to a standard Brownian motion:

$$X_t = (C_1(H))^{-1} \int_{\mathbb{R}} \left\{ |t-s|^{H-\frac{1}{2}} \mathbb{I}_{(-\infty,t]}(s) - |s|^{H-\frac{1}{2}} \mathbb{I}_{(-\infty,0]}(s) \right\} dB(s),$$

where $dB(\cdot)$ is a Brownian measure and where the constant $(C_1(H))^{-1} = \Gamma(H+1/2)/(\Gamma(2H+1)\sin \pi H)^{1/2}$ (see Taqqu [37]). The generation of the process is based on an approximation of the integral using truncated Riemann sums. We obtain, for $t = 1, \cdots, N-1$,

$$X\left(\frac{t}{N}\right) = \frac{(C_1(H))^{-1}}{N^H} \Big\{ \sum_{k=-a_N}^{0} \left((t-k)^{H-\frac{1}{2}} - (-k)^{H-\frac{1}{2}} \right) \varepsilon_k + \sum_{k=0}^{t} (t-k)^{H-\frac{1}{2}} \varepsilon_k \Big\}, \tag{2.9}$$

where ε_k are i.i.d. $\mathcal{N}(0,1)$ random variables, a_N is a sequence that satisfies $\lim_{N\to\infty} a_N = +\infty$. One can, however, be somewhat more sophisticated by introducing different parameters for the mesh and the cutoff in the limits of integration and adding other correction terms (see Mandelbrot and Wallis [28], Voss [42] and Samorodnitsky and Taqqu [34], Section 7.11).

Concrete simulation procedure: To generate a fractional Brownian motion X:

- Generate two independent sequences of i.i.d. random variables B_1 and B_2 from the Gaussian distribution $\mathcal{N}(0,1)$.
- Compute $X(t/N)$ for $t = 1, \cdots, N$ according to (2.9).

Implementation: The performance of this algorithm depends on the choice of a_N; we take $a_N = N^{1.5}$.

2.7 Pseudo-Markovian representation of the process

Carmona and Coutin [6] have introduced a new approach for the simulation of a process from its diffusive representation. It is based on a representation of the process as a stochastic integral with respect to a martingale. We present only the case of the fractional Brownian motion

$$X_t \;=\; \int_{-\infty}^{t} K_H(t,s) dB(s) \;\text{ with }\; K_H(t,s) = \left((t-s)^{H-1/2} - (-s)_+^{H-1/2} \right),$$

with $0 < H < 1/2$ and with $dB(\cdot)$ a Brownian measure. This representation cannot easily be discretized because $(t-s)^{H-1/2}$ is singular when $s \to t$. We shall modify it in the following way as suggested by Carmona, Coutin and Montseny [7]. Using the Laplace transform

$$\frac{1}{u^a} = \int_0^\infty \frac{x^{a-1}}{\Gamma(a)} e^{-ux} dx \quad a > 0, u > 0,$$

and setting $c_H = (\Gamma(H+1/2))^{-1}$ and $\gamma(v) = B(-v)$ for $v \geq 0$, one gets

$$X_t = c_H \int_0^\infty x^{-(H+1/2)} dx \left(\int_0^t e^{-x(t-s)} dB(s) + \int_0^\infty (e^{-x(t+v)} - e^{-xv}) d\gamma(v) \right),$$

by applying Fubini's stochastic theorem. Hence

$$X_t = \int_0^\infty Y(x,t) \mu_H(dx) \quad \text{with} \quad Y(x,t) = M(x,t) + (e^{-xt} - 1) Z(x), \tag{2.10}$$

where

$$M(x,t) = \int_0^t e^{-x(t-s)} dB(s), \quad Z(x) = \int_0^\infty e^{-xv} d\gamma(v), \quad \mu_H(dx) = x^{-(H+1/2)} dx.$$

Note that M is a Markovian process (Ornstein-Uhlenbeck process). In fact, $\partial M(x,t)/\partial t = -xM(x,t)dt + dB(t)$. One then performs a polygonal approximation of X.

Concrete simulation procedure: To generate a fractional Brownian motion X:

1. Define a discretization of $[0,T]$ by N points $0 = t_1 < t_2 < \ldots < t_{N-1} < t_N = T$ (typically, $t_k = (k-1)T/N$).
2. Let $\Pi = (0 = \eta_1 < \eta_2 < \ldots < \eta_K)$ be a finite partition of $[0, \infty)$ (typically, Π is a geometric sequence).
3. Let X^Π be the spatial polygonal approximation of X defined by

   $$X^\Pi_{t_j} = \sum_{k \in K} c_k Y(\eta_k, t_j) \quad \text{for each } t_j,$$

   with $c_k = \int_{\eta_k}^{\eta_{k+1}} x^{-(H+1/2)} dx$. The $Y(\eta_k, t_j)$ are defined by (2.10) and the following:

   - For each t_j, m Ornstein-Uhlenbeck processes $M(\eta_k, t_j)$ are generated from an Euler's scheme as follows: generate N normalized centered Gaussian random variables $(\zeta_j)_{1 \leq j \leq N}$; then, $B(t_j)$ is such that $B(t_{j+1}) - B(t_j) = \zeta_j \sqrt{t_{j+1} - t_j}$ and for each $k = 1, .., K$,

     $$M(\eta_k, t_{j+1}) - M(\eta_k, t_j) = -\eta_k M(\eta_k, t_j)(t_{j+1} - t_j) + B(t_{j+1}) - B(t_j).$$

- For each η_k, K random variables $Z(\eta_k)$ are generated from K normalized centered Gaussian random variables, with

$$\mathbb{E}Z(\eta_k)Z(\eta_l) = \int_0^{\infty} e^{-v\eta_k} e^{-v\eta_l} dv = (\eta_k + \eta_l)^{-1}.$$

  A Choleski procedure can be used to generate $(Z(\eta_k))_{(k=1,\dots,K)}$ quickly.

The choice of K depends on the desired precision of the polygonal approximation and Carmona and Coutin provide an inequality

$$\sup_{t\in[0,T]} \left\| X_t - X_t^{\Pi} \right\|_2 \leq C f(\Pi),$$

with $f(\Pi) \to 0$ when $K \to \infty$ (theoretical considerations indicate that K can be N^{β} for any $\beta > 0$).

Implementation: In practice, we chose $K = 2 * \lfloor \log(N) N^{\delta} \rfloor + 1$, where δ depends on H and is of the form $\delta = \beta/(1+\beta)$ with $\beta = \min(H, 1/2 - H)$.

Simulated models and possible extensions: This method has two advantages: the Markovian approximation M of Y can be computed with an iterative method and the spatial approximation Z of Y does not require a lot of computations. Carmona and Coutin [6] have extended this approach to generate fractional Brownian motion for $H > 1/2$ from the stochastic representation of this process introduced by Decreusefond and Üstünel [13]. But because of delicate convergence problems, this extension cannot be implemented numerically. Carmona and Coutin [6] also provide approximations of processes $W_K(t) = \int_0^t K(t,s)dM(s)$ where $M = (M(t), t \geq 0)$ is a square integrable continuous martingale, K is a deterministic kernel such that $\mathbb{E}(\int_0^t K^2(t,s)dA(s))) < \infty$, where $A = (A(t), t \geq 0)$ is the bracket process of M.

2.8 Aggregation of renewal processes

In a network traffic framework, the presence of fractional Brownian motion can be explained as resulting from the superposition of many ON/OFF sources that are heavy-tailed (see Willinger, Paxson, Riedi and Taqqu [43] and Taqqu [38]). Suppose that a traffic source $(W(t), t \geq 0)$ is a renewal-reward process that takes values 0 or 1 (the source is OFF at time t if $W(t) = 0$ and it is ON if $W(t) = 1$). Assume that the lengths of the ON-periods are i.i.d., those of the OFF-periods are i.i.d., the lengths of ON- and OFF-periods are independent and that the common distribution F of the length of the ON- and OFF-periods is

$$1 - F(x) \sim a x^{-\alpha} \text{ when } x \to \infty, \text{ with } 1 < \alpha < 2. \tag{2.11}$$

The distribution F is heavy-tailed and is typically stable or Pareto. It has finite mean and infinite variance. Now suppose that there are M i.i.d. sources denoted by $(W_m(t))_{1 \le m \le M}$. The number of sources that are ON at $t \ge 0$ is $\sum_{m=1}^{M} W_m(t)$ and the total activity from time 0 to time t of all the sources is thus

$$S_M(t) = \int_0^t \Big(\sum_{m=1}^{M} W_m(u) \Big) du.$$

Then the following central limit theorem holds:

$$\mathcal{L} \lim_{T \to \infty} \mathcal{L} \lim_{M \to \infty} \frac{1}{T^H \sqrt{M}} \left(S_M(Tt_i) - \frac{1}{2} M T t_i \right)_{1 \le i \le n} = \sigma \left(B_H(t_i) \right)_{1 \le i \le n}, \tag{2.12}$$

where $\mathcal{L}\lim$ denotes limit in the sense of finite-dimensional distributions, $H = (3 - \alpha)/2 > 1/2$, σ is a positive constant and $B_H(t)$ is a standard fractional Brownian motion with self-similarity parameter H.

Concrete simulation procedure and complexity of the algorithm: Suitable for parallel processing (see Willinger, Taqqu, Sherman and Wilson [44]).

Simulated models and possible extensions: Taqqu *et al.* [41] proved similar results in the following first three cases:

- If the lengths of ON- and OFF-periods have a finite second moment, the limit is Brownian motion.
- If the distributions of the length of ON- and OFF-periods are different but both satisfy (2.11) with perhaps different parameters α_1, α_2, then the limit is a fractional Brownian motion with $H = (3 - \min(\alpha_1, \alpha_2))/2$.
- If the number M of finite sources stays constant, the limit involves only T and is a Lévy stable motion, a process with independent increments. The normalization is $T^{1/\alpha}$.
- The W_ms do not have to be 0 or 1, nor do they need to have finite variance. If, however, they do not have finite variance, the limit will be an infinite variance stable process with dependent increments (see Levy and Taqqu [25] and Taqqu [38]).

2.9 Aggregation of autoregressive processes

Granger [18] proved that the aggregation of short-range dependent processes can lead to long-range dependence. Consider a white noise $(\varepsilon_i)_{i\in\mathbb{N}}$ of variance σ^2. Let $(X_j)_{j\in\mathbb{N}}$ be independent autoregressive AR[1] processes with parameters $(\alpha_j)_{j\in\mathbb{N}} \in (-1,1)^{\mathbb{N}}$ such that $X_j = (X_j(i), i \in \mathbb{N})$ and for each $(i,j) \in \mathbb{N}^2$,

$$X_j(i+1) = \alpha_j X_j(i) + \varepsilon_{i+1},$$

and define the aggregated time series $S_M = (S_M(i), i \in \mathbb{N})$ of M such processes by

$$S_M(i) = \sum_{j=1}^{M} X_j(i).$$

Now assume that $(\alpha_j)_{j\in\mathbb{N}}$ are i.i.d. random variables with density f and independent of $(\varepsilon_j)_{j\in\mathbb{N}}$. Granger [18] showed that if f is a density on $[0,1]$ of the form $f(u) = \phi(u)(1-u)^{q-1}$, $q > 0$, where ϕ is continuous at the point $u = 1$ and $\phi(1) \neq 0$, then S_M/M approximates a dependent process when $M \to \infty$, whose covariance function satisfies $r(k) \sim k^{1-2\beta}$ when k goes to infinity. Therefore ,we obtain a long range dependent process when $\beta \in (1/2, 1)$. For example, if the density f is a beta distribution with parameters (α, β) (i.e. $f(u) \propto u^{\alpha-1}(1-u)^{\beta-1}$), we obtain the FARIMA[0,d,0] processes, $0 < d < 1/2$, by taking $\beta = 1 - d$ and $\alpha = d$ (see Gonçalvez and Gouriéroux [17]). We describe hereafter the concrete simulation procedure for the FARIMA[0,d,0] processes.

Concrete simulation procedure:

1. Generate the i.i.d. $\mathcal{N}(0,1)$ sequence $\varepsilon = (\varepsilon_i)_{0\le i\le N}$.
2. Generate the i.i.d. random variables $(\alpha_j)_{0\le j\le M}$ with the Beta distribution $\mathcal{B}(d, 1-d)$, $0 < d < 1/2$.
3. For each $j \in \{1, \cdots, M\}$, generate AR[1] process X_j with coefficient α_j and the common white noise sequence ε

$$X_j(i+1) = \alpha_j X_j(i) + \varepsilon_{i+1}.$$

4. The approximate FARIMA[0,d,0] process is given by $\frac{1}{M}\sum_{j=1}^{M} X_j(i)$ for $i = 0, .., N$.

Implementation: We take $M = 50,\ 100,\ 1000$. Refer to the results of the simulations.

Simulated models and possible extensions: For a parallel approach, see Mandelbrot [27]. Oppenheim and Viano (see [31, 30]) have generalized Granger's approach. They obtain processes with seasonal effects by aggregating autoregressive AR[p] processes with random coefficients.

2.10 Truncated moving average expansion

Assume that $X = (X(i), i \in \mathbb{N})$ is a long-range dependent linear process such that for all $i \in \mathbb{N}$,

$$X(i) = \sum_{k=-\infty}^{\infty} a(k)\varepsilon_{i-k}, \text{ with } \sum_{k=-\infty}^{\infty} a^2(k) < \infty, \tag{2.13}$$

and $\varepsilon = (\varepsilon_k, k \in \mathbb{Z})$ is an i.i.d. sequence of random variables with mean 0 and variance 1. For instance, in the particular case of a FARIMA[0,d,0] with $d \in (-1/2, 1/2)$,

$$X(i) = \sum_{k=0}^{\infty} \frac{\Gamma(k+d)}{\Gamma(k+1)\Gamma(d)}\varepsilon_{i-k}. \tag{2.14}$$

Since $\lim_{k\to\infty} a(k) = 0$, with $a(k) = \frac{\Gamma(k+d)}{\Gamma(k+1)\Gamma(d)} = \mathcal{O}(k^{d-1})$ when k goes to infinity. Therefore, a natural approximation of X is $X^{(M)}(i) = \sum_{k=-M}^{M} a(k)\varepsilon_{i-k}$, and clearly for each $i \in \mathbb{N}$, as $M \to \infty$, $X^{(M)}(i)$ converges in $\mathbb{L}^2$ and almost surely to $X(i)$.

Concrete simulation procedure: To generate an approximate FARIMA[p,d,q] process $X(i)$, $i = 0, 1, .., N$:

1. Generate i.i.d. $\mathcal{N}(0, 1)$ random variables $(\varepsilon_k)_{-M \le k \le N+M}$.
2. Generate an approximate FARIMA[0,d,0] process $Y^{(M)}(i)$, $i = 0, .., N$ using (2.14) but with only M terms.
3. Generate an approximate FARIMA[p,d,q] process $X^{(M)}(i)$, $i = 0, 1, .., N$, by generating an ARMA[p,q] with the usual white noise replaced by $(Y^{(M)}(i))$.

$X^{(M)}$ is a short-range dependent approximation of X. The choice of M depends on the required precision. In practice, $M = 100$ or $M = 1000$ are often chosen.

Implementation: We take $M = 50,\ 100,\ 1000$. Refer to the results of the simulations.

Simulated models and possible extensions: Linear processes with coefficients available either in a closed form, through an iterative computational formula, or through an expansion in rational functions.

2.11 Expansion in a wavelet basis

The generation of a process from its expansion in a wavelet basis is a new and attractive domain of research. We only report on the case of fractional

Brownian motion but this approach can be extended to another processes, for example, elliptic Gaussian processes (see Benassi, Jaffard and Roux [3]) or non-Gaussian continuous time processes that have the same second order properties as fractional Brownian motion (see Meyer, Sellan and Taqqu [29]).

Let $X = (X_t, t \in \mathbb{R})$ be a fractional Brownian motion with parameter $H \in (0,1)$. The main idea is to expand X as

$$X_t(\omega) = \sum_{j=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} g_{j,k}(t)\varepsilon_{j,k}(\omega), \tag{2.15}$$

for all $t \in \mathbb{R}$ and all $\omega \in \Omega$. Here, $(g_{j,k}(t))_{j,k}$ is a non-random family of functions depending only on t and $(\varepsilon_{j,k})_{j,k}$ are i.i.d. random variables with a $\mathcal{N}(0,1)$ distribution not depending on t. We present two equivalent expansions of the form (2.15) using a wavelet decomposition of X.

The first expansion is stated in Benassi, Jaffard and Roux [3]. Let ϕ and ψ be the "father" and "mother" wavelets respectively of the Lemarié-Meyer wavelet basis, then the family $(\psi_{j,k}(.))_{(j,k)\in\mathbb{Z}^2}$ provides an orthonormal basis of $\mathbb{L}^2(\mathbb{R})$ with $\psi_{j,k}(x) = 2^{j/2}\psi(2^j x - k)$ for $x \in \mathbb{R}$ and $(j,k) \in \mathbb{Z}^2$. Denote by $\widehat{\psi_{j,k}}$ the Fourier transformation of ψ. Then X has the following wavelet representation,

$$X(t) = \sum_{j=0}^{\infty} \sum_{k=-\infty}^{\infty} < K(\cdot,t)|\widehat{\psi_{j,k}}(\cdot) > \varepsilon_{j,k}, \quad K(s,t) = \frac{e^{ist}-1}{|s|^{H+1/2}}, \tag{2.16}$$

for all $t \in \mathbb{R}$, $< f|g >= \int f\bar{g}$.

The second wavelet expansion of fractional Brownian motion is based on the time domain. It was introduced by Sellan [35] and implemented by Abry and Sellan [1]. A correct and generalized version was developed by Meyer, Sellan and Taqqu [29]. It is an analog to the "midpoint displacement method" used for generating Brownian motion. Starting with a wavelet expansion of a continuous-time Gaussian white noise, Meyer *et al.* obtain

$$X(t) = \sum_{j=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} 2^{-jH}\left[\psi_{-H-1/2}(2^j t - k) - \psi_{-H-1/2}(-k)\right]\varepsilon_{j,k}, \tag{2.17}$$

for all $t \in \mathbb{R}$ and with ψ_d such that $\widehat{\psi_d}(\lambda) = (i\lambda)^d\widehat{\psi}(\lambda)$ (i.e. ψ_d is the derivative of order d of ψ). The mother wavelet ψ has M vanishing moments with $M > H + 1/2$. For instance, Daubeshies wavelets of order $M \geq 2$ or Lemarié-Meyer wavelets can be used.

The low frequencies become more and more prominent as $j \to -\infty$. It is, in fact possible, to replace the terms in (2.17) corresponding to $j < 0$ by terms involving partial sums of FARIMA. One gets (Meyer *et al.* [29]),

$$X(t) = \sum_{k=-\infty}^{\infty} \Phi_{-H-1/2}(t-k)S_k^{(H)} + \sum_{j=0}^{\infty}\sum_{k=-\infty}^{\infty} 2^{-jH}\psi_{-H-1/2}(2^j t - k)\varepsilon_{j,k} - b_0, \tag{2.18}$$

with

- $\widehat{\Phi_d}(\lambda) = \left(\frac{i\lambda}{1-e^{-i\lambda}}\right)^d \widehat{\phi}(\lambda)$ where ϕ is the father wavelet,
- $(\varepsilon_{j,k})_{j,k}$ are i.i.d. $\mathcal{N}(0,1)$ random variables,
- $S_k^{(H)} = \sum_{j=1}^{k} y_j$ if $k \geq 1$, $S_0^{(H)} = 0$ and $S_k^{(H)} = -\sum_{j=0}^{-k} y_j$ if $k \leq 1$ where $(y_j)_{j\in\mathbb{N}}$ is a Gaussian FARIMA[0,H-1/2,0] process (independent of ε, with the spectral density $g(\lambda) = (2\pi)^{-1}|1-e^{-i\lambda}|^{1-2H}$),
- b_0 is a random variable that does not depend on t. It equals the sum of the first two terms in the right-hand side of (2.17) with $t = 0$, and hence ensures that the whole right-hand side vanishes at $t = 0$.

The first term in the right-hand side of (2.17) can be regarded as a "trend" (low frequencies) and the second as "fluctuations" (high frequencies).

Concrete simulation procedure: The representation (2.18) provides the following procedure to generate a fractional Brownian motion $(X(k))_{k\in\mathbb{N}}$:

1. Generate $(y_j)_{j\in\mathbb{N}}$ from a simulation of i.i.d. $\mathcal{N}(0,1)$ random variables $(z_k)_{k\in\mathbb{Z}}$ and keep only M terms in the following representation of FARIMA[0,H-1/2,0]
$$y_j = \sum_{l=0}^{\infty} \frac{\Gamma(k+H-1/2)}{\Gamma(k+1)\Gamma(H-1/2)} z_{j-k}.$$
2. Compute the partial sums $S_k^{(H)}$, $k \leq K$, and convolve them with the low-pass filter associated with $\phi_{-(H+1/2)}$.
3. Generate $(\varepsilon_{j,k})_{j,k}$, $k \leq K$ and $j = 0, 1, \ldots, J$, and convolve them with $\psi_{-(H+1/2)}$ as in (2.18). Use the multi resolution algorithm to compute this convolution from octave j to octave $j+1$.
4. Sum the "trend" term (with $S_k^{(H)}$) and the "fluctuations" term (with $(\varepsilon_{j,k})_{j,k}$) to obtain $X^*(t) = X(t) + b_0$. In practice, the random variable b_0 vanishes from X by taking $X(t) = X^*(t) - X^*(0)$ (so that $X(0) = 0$).

5. Extract the process by sub-sampling $X(t)$ every τ observations, the resulting process is $\left(\tilde{X}_{1+t\tau}\right)$ (with τ the lag value). We used $\tau = 10$.

From a theoretical point of view, the sub-sampling step is not necessary to simulate fBm. However, in practice, the high frequencies of the fBm process are not well approximated since when this step is omitted one obtains bias in the self-similarity parameter estimators.

Implementation: We use the Daubeshies wavelet of order 10 as mother wavelet. The number of octaves is $J = 10$. In the subsampling step, we fix the lag value equal to $\tau = 10$, and thus the effective data size becomes $N_s = N/\tau$. To approximate the process y_t, for instance, we choose M such that $\left|\frac{\Gamma(l+H-1/2)}{\Gamma(l+1)\Gamma(H-1/2)}\right| < 10^{-2}$ for $l > M$. The same strategy is used to truncate the infinite filters, where we kept only the coefficients larger than 10^{-2}. We used the program of Abry and Sellan [1].

Simulated models and possible extensions: The preceding discussion involved generating fractional Brownian motion. Because the preceding wavelet representations ensure only that the covariances fit, it is possible to use them even when the $\varepsilon_{j,k}$s are not Gaussian. The resulting process will not be Gaussian and will be only second-order self-similar, that is, only the covariances will scale. Benassi, Jaffard and Roux [3] have also developed a wavelet representation (similar to (2.16)) of filtered white noise and elliptic Gaussian processes, as well as one for multi-fractional Brownian motion.

2.12 Random midpoint displacement method

This method which is used to generate Brownian motion $B(t)$, $0 \leq t \leq 1$, has been adapted by different authors, e.g. Voss [42] and Lau, Erramilli, Wang and Willinger [23], to generate an approximation of fractional Brownian motion $B_H(t)$, $0 \leq t \leq 1$.

Concrete simulation procedure: The goal is to generate an approximation of $B_H(t)$ for $t = k/2^M,\ k = 0, \cdots, 2^M$. One does this recursively as follows.

- Start first with $B_H(0) = 0$ and $B_H(1) \sim \mathcal{N}(0, 1)$. Draw a line between these two points. This is the approximation of order 0.

- To get the next level of approximation, namely the polygonal line connecting the points $B_H(0), B_H(1/2), B_H(1)$, generate $B_H(1/2)$ as a Gaussian random variable with mean $1/2(B_H(0) + B_H(1))$ and variance $(1/2^2)^{2H} = (1/4)^{2H}$.

- Proceed recursively. At the level of approximation $j \geq 1$, generate $B_H(t)$, $t = k/2^j$, $k = 1, \cdots, 2^{j-1}$, k odd, as a Gaussian random variable with mean $1/2(B_H((k-1)/2^{j-1}) + B_H((k+1)/2^{j-1}))$ and variance $(1/2^{j+1})^{2H}$. Connect the points by a polygonal line. This polygonal line is the sample path at the level of approximation j.

Remark: The method is exact if $H = 1/2$ but incorrect if $H \neq 1/2$, because it focuses on the marginal distribution and not on the dependence structure. Since the dependence structure of the simulated sequence is incorrect, the variance at many dyadic points is incorrect as well unless $H = 1/2$.

The construction for $H = 1/2$ is in fact wavelet-based because it involves integrals of Haar wavelets (Shauder basis) as explained for instance in Meyer, Sellan and Taqqu [29].

Extensions: The method of Wornell [46], Theorem 3, for generating an approximation of fractional Gaussian noise is similar. There are two differences, however. In the Wornell method, the Haar wavelets are replaced by arbitrary ones and one does not integrate the wavelets because the goal is not to obtain fractional Brownian motion but an approximation of fractional Gaussian noise. All there methods, however, are inadequate because they do not take into account the dependence structure.

3 Empirical results from simulations

3.1 How to test a generator?

We evaluate the performance of the different generators of long-range dependent processes using the following two criteria:

- a test procedure on the estimation of the long-range dependence parameter;
- a test procedure on the goodness of fit of the spectral density of the process.

Let θ_0 denote the true value of the long-range dependence parameter (for example θ_0 is the Hurst parameter H for the fBm model or d for the FARIMA[0,d,0] model) .

Estimation of the long-range dependence parameter

We use the Whittle estimator. This choice is motivated by the fact that the model is known explicitly and that the asymptotic distribution of $\hat{\theta}_N$ is known as well. Indeed,

$$\widetilde{\theta}_N := \sqrt{N}\,\frac{\hat{\theta}_N - \theta_0}{\sigma_{\theta_0}} \xrightarrow[N\to\infty]{\mathcal{L}} \mathcal{N}(0,1), \qquad (3.1)$$

with

$$\sigma_{\theta_0}^2 = 2\left\{\frac{1}{2\pi}\int_{-\pi}^{\pi}\left(\frac{\partial}{\partial\theta}\log(f(\lambda,\theta))\right)^2 d\lambda\Big|_{\theta=\theta_0}\right\}^{-1},$$

where $f(\lambda,\theta)$ $(\theta\in\Theta\subset\mathbb{R})$ is the spectral density (see Fox and Taqqu [16]). Note that for FARIMA[0,d,0] models, the asymptotic variance is independent of d and equal to $6/\pi^2$ (see Beran [4]). For this model, we denote $\widetilde{d}_N = \widetilde{\theta}_N$. For the fBm model, an explicit form of σ_H^2 is not available. The integral in the expression of $\sigma_{\theta_0}^2$ is approximated using Riemann sums (refer to the `Splus` program provided by Beran [4]). In this particular model, we denote $\widetilde{H}_N = \widetilde{\theta}_N$. This result allows to implement the following test procedure:

$$H_0\ :\quad \theta=\theta_0 \quad \text{versus} \quad H_1\ :\quad \theta\neq\theta_0.$$

Sometimes we use the *local Whittle estimator* where one focuses only on the first m Fourier frequencies $\lambda_j = 2\pi j/N, j=1,\dots,m$ (see Bardet, Lang, Oppenheim, Philippe, Stoev, Taqqu [2]). We take $m=30$ if $N=1000$ and $m=300$ if $N=10000$.

Goodness of fit of the spectral density

We consider a test on the spectral density to assess the performance of the different generators. Let I be the periodogram of the observed values $(X_t)_{t\in\{1,\dots,N\}}$, i.e.

$$I(\lambda_k) = \frac{1}{2\pi N}\left|\sum_{t=1}^{N} X_j e^{-it\lambda_k}\right|^2,$$

where $\lambda_k = 2\pi k/N$ is the kth Fourier frequency.

Consider the test of hypothesis:

$$H_0\ :\ f(\lambda)\equiv f(\lambda,\theta_0) \quad \text{versus} \quad H_1\ :\ f(\lambda)\neq f(\lambda,\theta_0),$$

where $f(\cdot,\theta_0)$ denotes the spectral density of the process. We will use the following statistic based on the periodogram:

$$T_N = \left\{\frac{2\pi}{N}\sum_{t=0}^{N-1}\widetilde{f}_e(\lambda_t)\right\}^{-2}\left\{\frac{2\pi}{N}\sum_{t=0}^{N-1}\widetilde{f}_e(\lambda_t)^2\right\},$$

where

$$\widetilde{f}_e(\lambda) = \frac{2\pi}{N}\sum_{j=1}^{N-1} W(\lambda-\lambda_j)\frac{I(\lambda_j)}{f(\lambda_j,\theta_0)},$$

and where W is defined by

$$W(\lambda) = \frac{1}{2\pi}\sum_{|h|<N} k(h/p_N)e^{-ih\lambda},$$

with the Bartlett kernel function $k(x) = (1 - |x|)\mathbb{I}_{[-1,1]}(x)$ and the bandwidth $p_N = 3N^{0.4}$. Under the hypothesis H_0, we have,

$$\widetilde{T}_N := \frac{N(T_N - C_N(k))}{\sqrt{D_N(k)}} \xrightarrow[N\to\infty]{\mathcal{L}} \mathcal{N}(0,1),$$

with

$$C_N(k) = \frac{1}{N\pi}\sum_{i=1}^{N-1}(1 - i/N)k^2(i/p_N) + \frac{1}{2\pi}$$

and

$$D_N(k) = \frac{2}{\pi^2}\sum_{i=1}^{N-2}(1 - i/N)(1 - (i+1)/N)k^4(i/p_N).$$

(See Chen and Deo [8], for the proof.)

3.2 How to read the tables and figures

We illustrate the performance of the different generators for the following models:

- fractional Brownian motion (fBm) with $H \in \{0.1, 0.2, \cdots, 0.9\}$ or in a subset thereof, depending on the assumptions of the generator.
- the fractional ARIMA model (FARIMA[0,d,0]) with $d \in \{0.05, 0.25, 0.45\}$.

For each procedure, 100 independent paths of a chosen process are generated, i.e. 100 time series of length 100 or 1000 (and sometimes 10000). The results are presented in the following tables and boxplots.

The tables, for instance Table 1.1, are associated with the distribution of $\widetilde{\theta}_n$ and $\widetilde{T}_n$, the two test statistics. Let us explain the lines of Table 1.1. For each value of (N, θ_0) (where N is the length of the path), we simulate N_r independent samples. We can therefore estimate, for each test statistic, the frequency $\hat{F}_{N_r}$ of acceptance of H_0, when the nominal level of the test is fixed (here $\alpha = 5\%$). The expected probability of acceptance, which is 95%, must be compared to the frequency $\hat{F}_{N_r}$ of acceptance. The value of $\hat{F}_{N_r}$ is displayed on the first line of the table for each test procedure (indicated by %). If the generator is well designed, we expect F_{N_r} to belong to the interval

$$\left[.95 - 1.96\sqrt{\frac{.95 * .05}{N_r}}, .95 + 1.96\sqrt{\frac{.95 * .05}{N_r}}\right] \simeq [0.907, 0.992],$$

since here $N_r = 100$. This confidence region is obtained using a classical test procedure for the null hypothesis $H_0' \ : \ p = 0.95$.

The second and third lines of the table display the sample mean and the sample standard deviation of $\widetilde{\theta}_N$ and $\widetilde{T}_N$ computed for the N_r independent samples. The expected mean (resp. standard deviation) which is 0 (resp. 1) must be compared to the sample mean (resp. the sample standard deviation) of both statistics. We will again construct confidence intervals based on the classical test procedures for the mean and the variance of a Gaussian sample. For the variance, we use the normal approximation of the corresponding χ^2 confidence interval.

We will say that the generator is a "good" generator if:

- the % line displays values close to 0.95, in the interval $[0.90, 0.99]$,
- the MEAN line displays values close to 0, in the interval

$$\left[\frac{-1.96}{\sqrt{N_r}}, \frac{1.96}{\sqrt{N_r}}\right] = [-0.196, 0.196],$$

- the SD line displays value close to 1, in the interval

$$\left[\sqrt{1 - 1.96\sqrt{2/N_r}}, \sqrt{1 + 1.96\sqrt{2/N_r}}\right] = [0.85, 1.13].$$

Let us now turn to the boxplots which display the values of the biases $\widehat{H}_N - H$ or $\widehat{d}_N - d$. These boxplots are generated by `Matlab`. Each box has lines at the lower quartile (quantile 25%), median, and upper quartile (quantile 75%) values. The whiskers are lines extending from each end of the box to show the extent of the rest of the data. Outliers are values beyond the ends of the whiskers, they are marked by a + symbol. The length of the whiskers varies as a function of the distance between the upper and lower quartile.

3.3 Results of the simulation

TO THE EDITOR: PLACE ALL THE TABLES AND FIGURES HERE, EXCEPT TABLE 1.12

Table 1.1 and Figure 1 illustrate the *Durbin-Levinson* algorithm described in Section 2.2 to simulate fBm models. For $N = 100$ and $H > 0.4$, the algorithm satisfies our definition of "a good generator". However, this efficiency is not preserved when we increase the sample size.

Tables 1.2, 1.3 and Figures 2 and 3 illustrate the algorithm based on the *circulant matrix method* described in Section 2.3 to simulate fBm and FARIMA models. The results generally satisfy our definition of "a good generator". Note that for the fBm model, when the parameter H is close to zero and the length of the trajectory increases, the generator becomes less accurate. This may be due to problems with the accuracy of the numerical computations. The results are particularly bad when $N = 10000$.

The *Lowen circulant method* for fBm, $0 < H < 1/2$, is particularly good (Table 1.4 and Figure 4).

The *direct spectral method* for fBm performs very badly on the spectral density test procedure (Table 1.5). For the estimation of the parameter H, however, the generator becomes more accurate when the sample size increases (Figure 5).

Table 1.6 and Figure 6 illustrate the algorithm given in Section 2.6 (called the *discretization of the integral* method) to simulate fBm with $H \geq 0.5$. The results are good for the test procedure on the goodness of fit of the spectral density. The estimation of the long range dependence parameter is somewhat biased, however.

Table 1.7 and Figure 7 illustrate the algorithm given in Section 2.7 to simulate fBm with $H < 0.5$, based on *pseudo-Markovian representations*. Note that, the method does not work for $H > 0.5$. The test procedure on the spectral density gives good results except for values of H close to zero. However, there is a sizable empirical bias.

Table 1.8 and Figure 8 illustrate the *aggregation of autoregressive AR[1] processes* algorithm given in Section 2.9 to simulate a FARIMA model. The performance is clearly sensitive to the number M of simulated AR processes and it seems that a long trajectory needs a larger number of AR processes. In many cases, the generator is not accurate when $M < 1000$.

Table 1.9 and Figure 9 illustrate the *truncated moving average* algorithm given in Section 2.9 to simulate the FARIMA model. This algorithm works quite well even with $M = 100$.

Table 1.10 and Figure 10 illustrate the *expansion in a wavelet basis* algorithm given in Section 2.11 to simulate fBm model. The results satisfy our definition of "a good generator" for long-range dependence processes (i.e. $H > 1/2$). Note that when the sub-sampling step is not omitted, the results of both test procedures are very poor. More precisely, the parametric Whittle estimators is biased with a positive (resp. negative) bias when $H < 1/2$ (resp $H > 1/2$). When we replace the Whittle estimator by the local Whittle estimator (see Taqqu and Teverovsky [39]), this bias is not observed since the estimation is based on the lower frequencies of the process. Figure 11 shows the estimation results using the local Whittle estimator and the Whittle estimator, applied to the full sample path. Finally, Figure 12 gives a comparison of the spectrum when the sub-sampling step is or not omitted. More precisely we represent the ratio $\hat{f}_H(\lambda_k)/f_H(\lambda_k)$ where $\hat{f}_H(\lambda_i)$ is an estimation of the spectral density f_H and $\lambda_k = 2\pi k\tau/N$ for $k = 1, \ldots, N/(2\tau)$. The figure shows that for $H = 1/2$, no subsampling is necessary; for $H > 1/2$, the low-frequency behavior is correct and the high frequencies are adjusted by the subsampling step; for $H < 1/2$, the opposite phenomenon takes place.

Table 1.11 and Figure 13 illustrate the algorithm given in Section 2.12 to simulate fBmn, based on the *random midpoint dispacement* method. Note that the method does not work for $H \neq 0.5$.

Table 1.12 summarizes the properties of the generators.

TO THE EDITOR: PLACE TABLE 1.12 HERE

Acknowledgment. Murad S. Taqqu was partially supported by the NSF Grant ANI-9805623 at Boston University. We would like to thank Stilian Stoev for his work on the Lowen circulant method and for his comments.

References

[1] P. Abry and F. Sellan. The wavelet-based synthesis for the fractional Brownian motion proposed by F. Sellan and Y. Meyer: Remarks and fast implementation. *Applied and Computational Harmonic Analysis*, 3(4):377–383, 1996.

[2] J.-M. Bardet, G. Lang, G. Oppenheim, A. Philippe, S. Stoev, and M.S. Taqqu. Semi-parametric estimation of the long-range dependence parameter : A survey. In P. Doukhan, G. Oppenheim, and M. S. Taqqu, editors, *Long-range Dependence: Theory and Applications.* Birkhäuser, 2001. Appears in this volume.

[3] A. Benassi, S. Jaffard, and D. Roux. Elliptic Gaussian random processes. *Revista Matematica Iberoamericana*, 13:19–90, 1997.

[4] J. Beran. *Statistics for Long-Memory Processes.* Chapman & Hall, New York, 1994.

[5] P. J. Brockwell and R. A. Davis. *Time Series: Theory and Methods.* Springer-Verlag, New York, 2nd edition, 1991.

[6] P. Carmona and L. Coutin. Fractional Brownian motion and the Markov property. *Electronic Communications in Probability*, 3:95–107, 1998.

[7] P. Carmona, L. Coutin, and G. Montseny. Approximation of some gaussian processes. *Statistical Inference for Stochastic Processes*, 3:161–171, 2000.

[8] W. Chen and R. Deo. A generalized portmanteau goodness-of-fit test for time series models. Technical report, Stern School of Business, New York Univ, 2000.

[9] J.-F. Coeurjolly. *Inference statistique pour les mouvements browniens fractionnaires et multifractionnaires.* PhD thesis, Université Grenoble, 2000.

[10] G. Corsini and R. Saletti. A $1/f^\gamma$ power spectrum noise sequence generator. *IEEE Transactions on Instrumentation and Measurement*, 37:615–619, 1988.

[11] M. S. Crouse and R. G. Baraniuk. Fast, exact synthesis of gaussian and nongaussian long-range-dependent processes. submitted to *IEEE Transactions on Information Theory*, 1999.

[12] R. B. Davies and D. S. Harte. Tests for Hurst effect. *Biometrika*, 74:95–101, 1987.

[13] L. Decreusefond and A. S. Üstünel. Stochastic analysis of the fractional Brownian motion. *Potential Analysis*, 10:177–214, 1999.

[14] C. R. Dietrich and G. N. Newsam. Fast and exact simulation of stationary Gaussian processes through circulant embedding of the covariance matrix. *SIAM Journal on Scientific Computing*, 18(4):1088–1107, 1997.

[15] J. Feder. *Fractals*. Plenum Press, New York, 1988.

[16] R. Fox and M. S. Taqqu. Large-sample properties of parameter estimates for strongly dependent stationary Gaussian time series. *The Annals of Statistics*, 14:517–532, 1986.

[17] E. Gonçalvez and C. Gouriéroux. Agrégation de processus autorégressifs d'ordre 1. *Annales d'Economie et de Statistique*, 12:127–149, 1988.

[18] C. W. Granger. Long memory relationships and the aggregation of dynamic models. *Journal of Econometrics*, 14:227–238, 1980.

[19] M. A. Hauser and W. Hörmann. The generation of stationary Gaussian time series. Technical report. Department of Statistics, University of Economics and Business Administration, Vienna, Austria, 1997.

[20] R. Jennane, R. Harba, and G. Jacquet. Estimation de la qualité des méthodes de synthèse du mouvement brownien fractionnaire. *Traitement du Signal*, 13(4):289–302, 1996.

[21] N.J. Kasdin. Discrete simulation of colored noise and stochastic processes and $1/f^\alpha$ power law noise generation. In *Proceedings of the IEEE*, volume 83, pages 802–827, 1995.

[22] P. S. Kokoszka and M. S. Taqqu. Can one use the Durbin-Levinson algorithm to generate infinite variance fractional ARIMA time series. *Journal of Time Series Analysis*, 2001. To appear.

[23] W.-C. Lau, A. Erramilli, J. L. Wang, and W. Willinger. Self-similar traffic generation: The random midpoint displacement algorithm and its properties. In *Proceedings of the ICC '95*, pages 466–472, Seattle, WA, 1995.

[24] S. Ledesma and D. Liu. A fast method for generating self-similar network traffic. In *Proceedings of the 2000 International Conference on Communication Technologies*, pages 54–61, Beijing, China, 2000.

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

[26] S. Lowen. Efficient generation of fractional Brownian motion for simulation of infrared focal-plane array calibration drift. *Methodology and Computing in Applied Probability*, 1(4):445–456, 1999.

[27] B. B. Mandelbrot. A fast fractional Gaussian noise generator. *Water Resources Research*, 7:543–553, 1971.

[28] B .B. Mandelbrot and J. R. Wallis. Computer experiments with fractional Gaussian noises, Parts 1,2,3.˙ *Water Resources Research*, 5:228–267, 1969.

[29] Y. Meyer, F. Sellan, and M. S. Taqqu. Wavelets, generalized white noise and fractional integration: the synthesis of fractional Brownian motion. *The Journal of Fourier Analysis and Applications*, 5(5):465–494, 2000.

[30] G. Oppenheim, M. Ould Haye, and M.-C. Viano. Long memory with seasonal effects. *Statistical Inference for Stochastic Processes*, 3:53–68, 2000.

[31] G. Oppenheim and M.-C. Viano. Obtaining long memory by aggregating random coefficients discrete and continuous time simple short memory processes. Technical Report 49-V, Pub. IRMA, Lille, 1999.

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

[33] H.-O. Peitgen and D. Saupe, editors. *The Science of Fractal Images*. Springer-Verlag, New York, 1988.

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

[35] F. Sellan. Synthèse de mouvements browniens fractionnaires à l'aide de la transformation par ondelettes. *Comptes Rendus de l'Académie des Sciences de Paris*, 321:351–358, 1995. Série I.

[36] F.B. Sowell. Maximum likelihood estimation of stationary univariate fractionally integrated time series models. *Journal of Econometrics*, 53:165–188, 1992.

[37] M. S. Taqqu. Fractional Brownian motion and long-range dependence. In P. Doukhan, G. Oppenheim, and M. S. Taqqu, editors, *Long-range Dependence: Theory and Applications*. Birkhäuser, 2001. Appears in this volume.

[38] M. S. Taqqu. The modeling of Ethernet data and of signals that are heavy-tailed with infinite variance. Preprint. To appear in the *Scandinavian Journal of Statistics*, 2001.

[39] M. S. Taqqu and V. Teverovsky. On estimating the intensity of long-range dependence in finite and infinite variance series. In R. Adler, R. Feldman, and M. S. Taqqu, editors, *A Practical Guide to Heavy Tails: Statistical Techniques and Applications*, pages 177–217, Boston, 1998. Birkhäuser.

[40] M. S. Taqqu, V. Teverovsky, and W. Willinger. Estimators for long-range dependence: an empirical study. *Fractals*, 3(4):785–798, 1995. Reprinted in *Fractal Geometry and Analysis*, C.J.G. Evertsz, H-O Peitgen and R.F. Voss, editors. World Scientific Publishing Co., Singapore, 1996.

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

[42] R. F. Voss. Fractals in nature: from characterization to simulation. In H.-O. Peitgen and D. Saupe, editors, *The Science of Fractal Images*, chapter 1, pages 21–70. Springer-Verlag, New York, 1988.

[43] W. Willinger, V. Paxson, R. H. Riedi, and M. S. Taqqu. Long-range dependence and data network traffic. In P. Doukhan, G. Oppenheim, and M. S. Taqqu, editors, *Long-range Dependence: Theory and Applications*. Birkhäuser, 2001. Appears in this volume.

[44] W. Willinger, M. S. Taqqu, R. Sherman, and D. V. Wilson. Self-similarity through high-variability: statistical analysis of Ethernet LAN traffic at the source level. *IEEE/ACM Transactions in Networking*, 5(1):71–86, 1997. Extended Version of the paper with the same title

that appeared in *Computer Communications Review* Vol. 25 (1995) 100-113.

[45] A. T. A. Wood and G. Chan. Simulation of stationary Gaussian processes in $[0,1]^d$. *Journal of Computational and Graphical Statistics*, 3:409–432, 1994.

[46] G. W. Wornell. Wavelet-based representations for the $1/f$ family of fractal processes. In *Proceedings of the IEEE*, volume 81, pages 1428–1450, 1993.

Jean-Marc Bardet, Laboratoire de Statistique et Probabilités, 118 route de Narbonne, 31062 Toulouse, FRANCE,
e-mail: `bardet@cict.fr`

Gabriel Lang, Laboratoire GRESE, Ecole Nationale du Genie Rural, des Eaux et des Forets, 19 avenue du Maine, 75732 Paris Cedex 15, FRANCE,
e-mail: `lang@engref.fr`

Anne Philippe, Laboratoire de Probabilités et Statistique, bat M2, UFR de mathematiques, Université des Sciences et Technologies, 59655 Villeneuve d'Ascq Cedex, FRANCE,
e-mail: `Anne.PHILIPPE@univ-lille1.fr`

Georges Oppenheim, Laboratoire de Modélisation Stochastique et Statistique, Bat 425, Université de Paris Sud, 91 405 Orsay Cedex, FRANCE,
e-mail: `georges.oppenheim@math.u-psud.fr`

Murad S. Taqqu, Department of Mathematics, 111 Cummington St., Boston University, Boston, MA 02459, U.S.A.,
e-mail: `murad@math.bu.edu`

TABLE 1.1. fGn model: Percentage of sample accepted at 5% level of significance, mean and standard deviation of the statistic $\widetilde{H}_N$ and $\widetilde{T}_N$ for *Durbin-Levinson* method. Values in boldface indicate that the statistic is outside the confidence band. (H: Hurst parameter, N: length of the trajectory).

N	H	.1	.2	.3	.4	.5	.6	.7	.8	.9
Whittle estimation										
100	%	**.47**	**.73**	.90	.95	.95	.95	.97	.98	.98
	Mean	**2.070**	**1.279**	**.536**	.101	-.059	-.058	-.022	-.081	-.180
	SD	1.101	1.105	1.076	1.060	1.046	1.028	.996	.920	.951
1000	%	**0**	**.01**	**.52**	.93	.94	.93	.90	**.85**	**.87**
	Mean	**8.71**	**5.01**	**1.926**	.190	-.136	.088	**.503**	**.804**	**.731**
	SD	**1.34**	**1.20**	1.105	1.071	1.068	1.074	1.075	1.069	1.013
Test on the spectral density										
100	%	**.59**	**.86**	.96	.96	.97	.97	.97	.96	.97
	Mean	**1.901**	**.552**	.142	.104	.109	.096	.083	.077	.070
	SD	**3.08**	**1.601**	1.120	1.083	1.073	1.068	1.070	1.0792	1.087
1000	%	**0**	**.03**	**.69**	.96	.95	.96	.96	.93	.91
	Mean	**82.19**	**13.68**	**1.268**	.053	.030	.032	.110	**.257**	**.348**
	SD	**59.24**	**1.797**	**1.890**	.999	.972	.986	1.058	**1.172**	**1.241**

FIGURE 1. fGn Model: Boxplot of the bias of the estimated parameter H simulated by *Durbin-Levinson* method. (H: Hurst parameter, N: length of the trajectory).

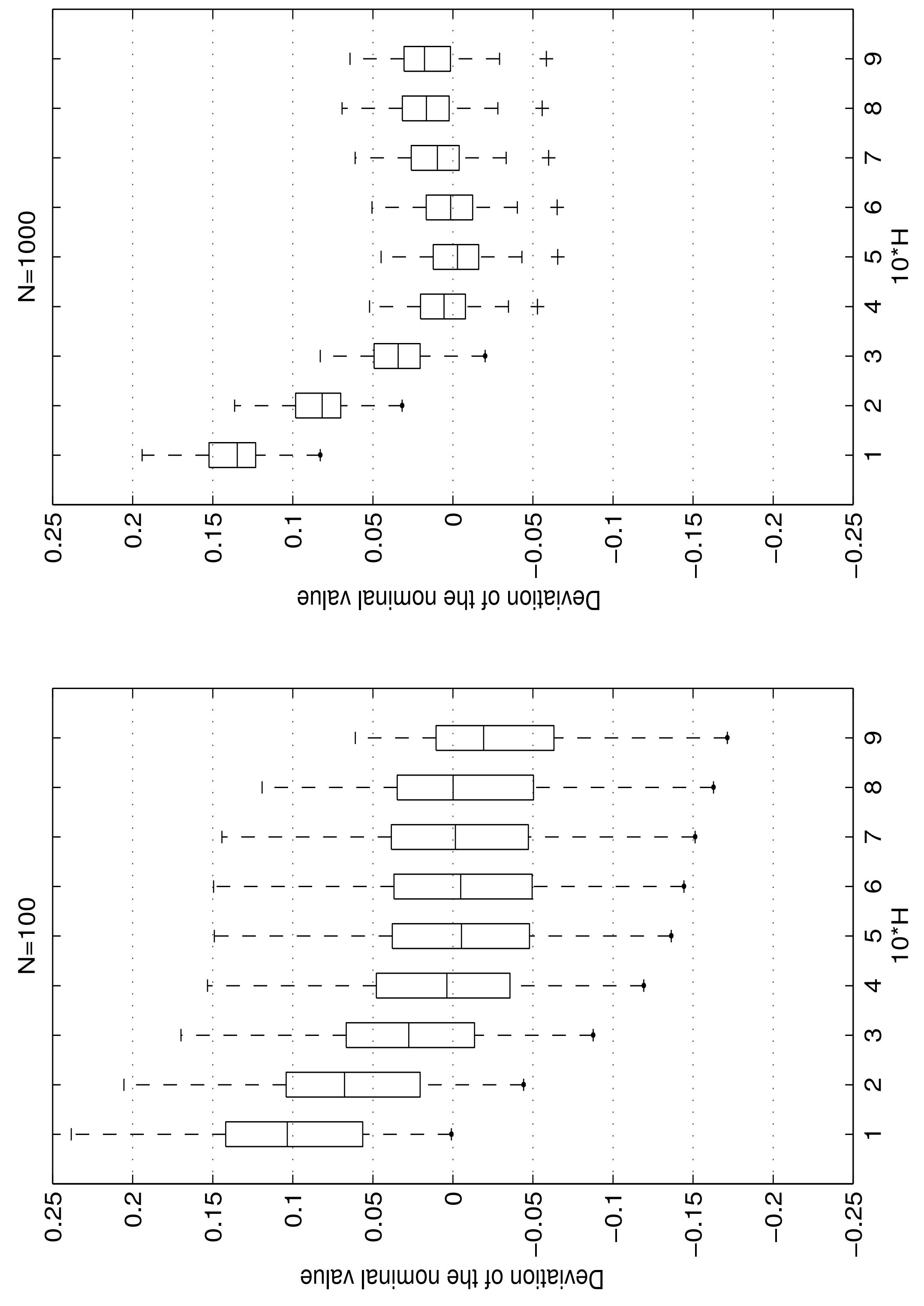

TABLE 1.2. fGn model: Percentage of sample accepted at 5% level of significance, mean and standard deviation of the statistic $\widetilde{H}_N$ and $\widetilde{T}_N$ for *circulant matrix method*. Values in boldface indicate that the statistic is outside the confidence band. (H: Hurst parameter, N: length of the trajectory).

N	H	.1	.2	.3	.4	.5	.6	.7	.8	.9
		Whittle estimation								
100	%	.99	.92	.93	.95	.95	.94	.94	.96	.94
	Mean	-.011	.016	.008	-.028	-.067	-.103	-.143	**-.211**	**-.398**
	SD	.897	1.084	1.099	1.099	1.096	1.089	1.071	1.027	.897
1000	%	**.82**	.96	.98	.97	.97	.97	.97	.97	.96
	Mean	**-1.285**	**-.308**	-.082	-.039	-.034	-.032	-.025	-.013	-.022
	SD	.881	.857	.863	.868	.869	.872	.877	.885	.887
10000	%	**0**	**.76**	.95	.95	.97	.96	.96	.96	.96
	Mean	**-4.978**	**-1.316**	**-.413**	-.190	-.138	-.124	-.117	-.105	-.092
	SD	.985	.968	.968	.965	.963	.961	.961	.961	.963
		Test on the spectral density								
100	%	.96	.97	.96	.96	.96	.96	.96	.96	.96
	mean	.013	-.042	-.043	-.041	-.039	-.041	-.044	-.049	-.055
	SD	.992	.945	.944	.947	.950	.954	.958	.963	.967
1000	%	**.85**	.94	.95	.94	.94	.94	.94	.94	.940
	Mean	**.613**	.080	.030	.024	.025	.027	.030	.035	.046
	SD	**1.277**	1.107	1.093	1.093	1.094	1.096	1.098	1.100	1.103
10000	%	**.10**	**.87**	.90	.92	.92	.92	.92	.92	.920
	Mean	**3.972**	**.417**	.144	.118	.115	.114	.114	.114	.117
	SD	**1.675**	1.119	1.065	1.059	1.057	1.056	1.056	1.055	1.055

FIGURE 2. fGn Model: Boxplot of the bias of the estimated parameter H simulated by *circulant matrix method.* (H: Hurst parameter, N: length of the trajectory).

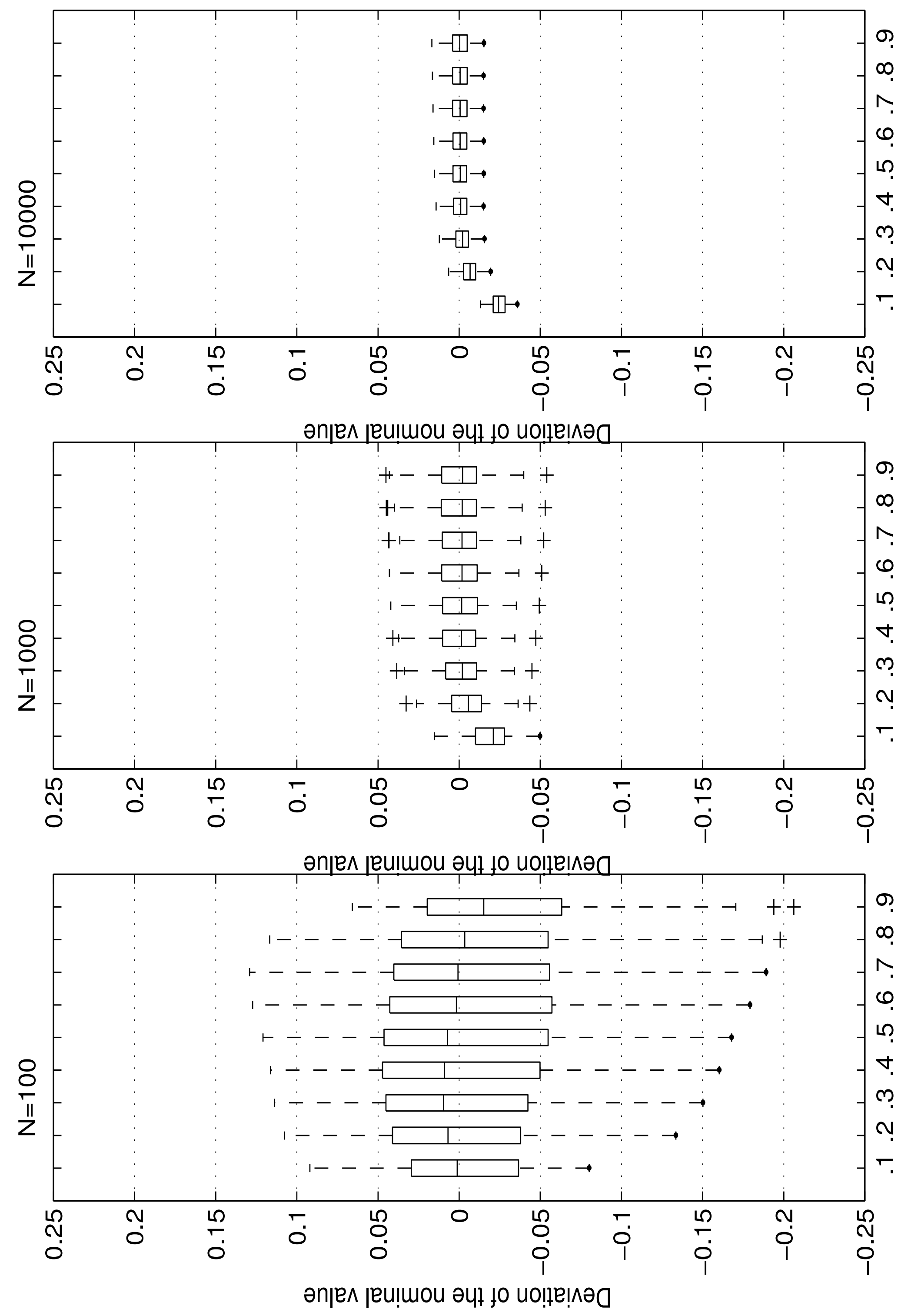

TABLE 1.3. FARIMA[0,d,0] model: Percentage of sample accepted at 5% level of significance, mean and standard deviation of the statistic $\widetilde{d}_N$ and $\widetilde{T}_n$ by *circulant matrix method*. Values in boldface indicate that the statistic is outside the confidence band. (d: long-range dependence parameter, N: length of the trajectory).

N	100			1000			10000		
d	.05	.25	.45	.05	.25	.45	.05	.25	.45
				Whittle estimation					
%	.93	.92	.91	.90	.90	.91	.98	.98	.97
Mean	-.042	-.006	.060	-.131	-.108	-.012	.004	.025	.088
SD	**1.182**	**1.171**	**1.172**	1.092	1.092	1.100	.850	.848	.840
				Test on the spectral density					
%	.99	.98	.97	.98	.98	.97	.98	.98	.97
Mean	.011	-.004	-.015	.031	.027	.017	.031	.027	.017
SD	.856	.874	.919	.942	.940	.950	.942	.940	.950

FIGURE 3. FARIMA[0,d,0] model: Boxplot of the bias of the estimated parameter d simulated by *circulant matrix method.*

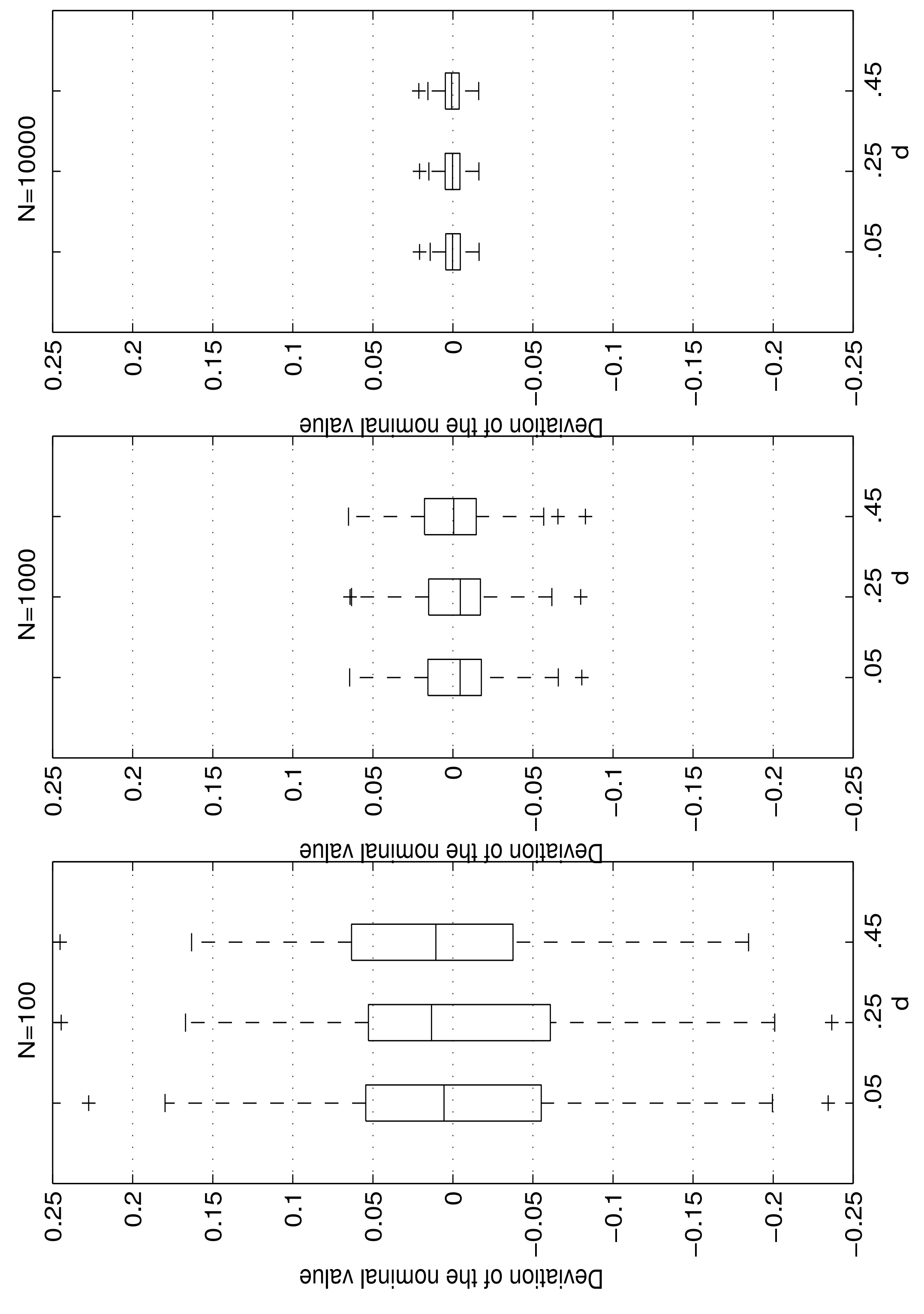

TABLE 1.4. fBm model: Percentage of sample accepted at 5% level of significance, mean and standard deviation of the statistic $\widetilde{H}_N$ and $\widetilde{T}_N$ for the *Lowen's circulant* method. Values in boldface indicate that the statistic is outside the confidence band. (H: Hurst parameter, N: length of the trajectory).

N	100					1000				
H	.1	.2	.3	.4	.5	.1	.2	.3	.4	. 5
Whittle estimation										
%	0.96	0.91	0.91	0.90	0.90	**0.75**	0.94	0.97	0.95	0.97
mean	0.138	0.084	0.055	0.006	-0.089	**-1.272**	-0.157	-0.005	0.043	0.010
SD	1.024	1.119	1.125	1.136	1.111	0.978	0.960	0.955	0.957	0.867
Test on the spectral density										
%	0.92	0.93	0.94	0.93	0.97	0.90	0.93	0.95	0.94	0.97
mean	0.187	0.194	0.187	0.173	0.018	**0.401**	-0.058	-0.080	-0.086	-0.074
SD	1.130	1.128	1.124	1.135	0.961	**1.148**	1.021	1.024	1.029	0.858

FIGURE 4. fBm Model: Boxplot of the bias of the estimated parameter H simulated by the *Lowen's circulant* method. (H: Hurst parameter, N: length of the trajectory).

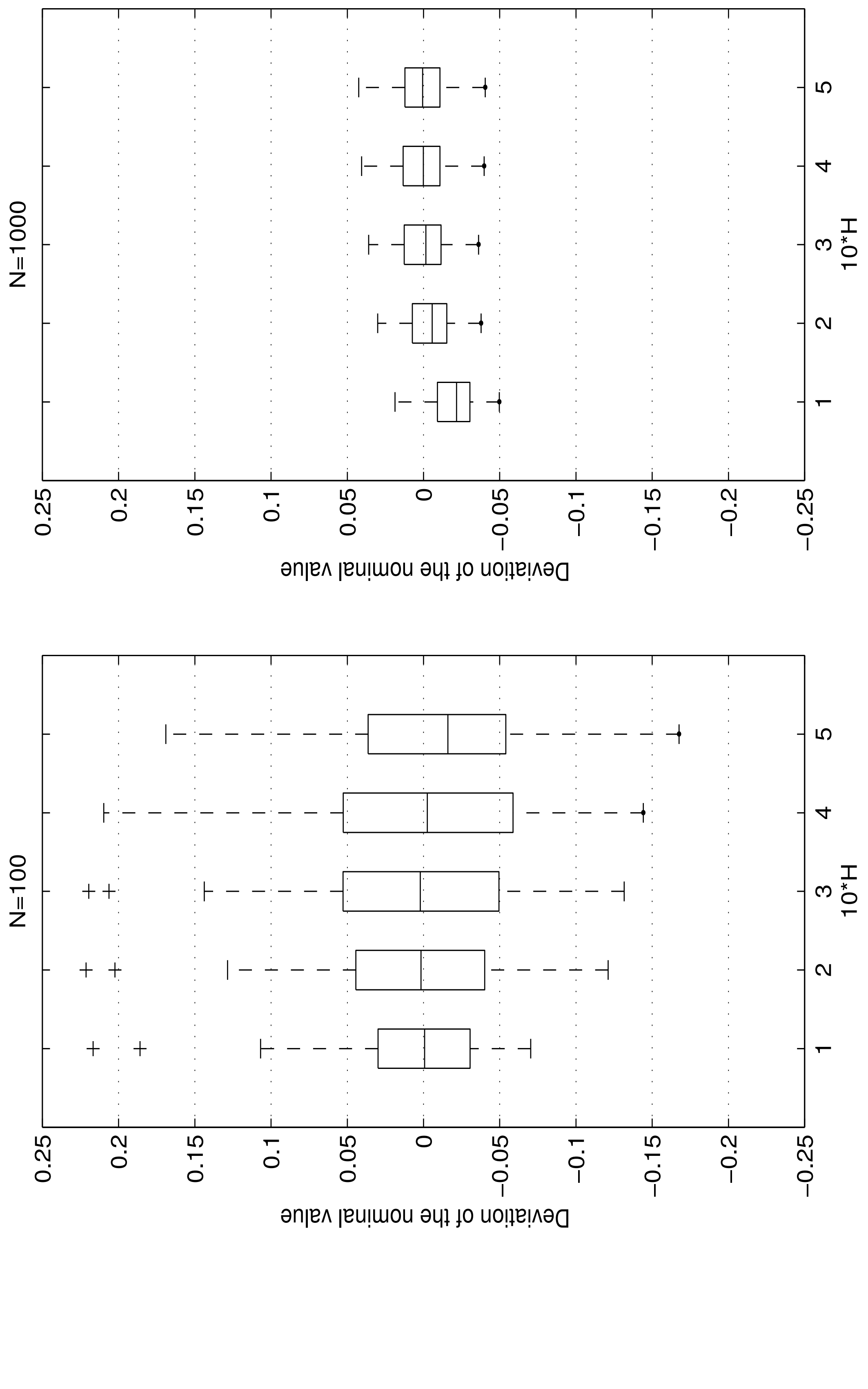

TABLE 1.5. fGn model: Percentage of sample accepted at 5% level of significance, mean and standard deviation of the statistic $\widetilde{H}_N$ and $\widetilde{T}_N$ for the *direct spectral method.* Values in boldface indicate that the statistic is outside the confidence band. (H: Hurst parameter, N: length of the trajectory).

N	H	.1	.2	.3	.4	.5	.6	.7	.8	.9
Whittle estimation										
100	%	**.81**	**.77**	**.85**	**.85**	**.84**	**.86**	**.81**	.93	**.87**
	Mean	**.838**	**.255**	.132	.055	-.121	**-.318**	**-.486**	**-.454**	**-.681**
	SD	**1.209**	**1.455**	**1.334**	**1.483**	**1.404**	**1.376**	**1.339**	**1.200**	1.045
1000	%	**.82**	.93	.95	.90	.91	.91	**.87**	**.80**	.94
	mean	**.385**	-.024	**.278**	.106	-.011	**-.276**	-.076	**-.201**	-.186
	SD	**1.306**	1.119	1.007	1.100	**1.177**	**1.240**	**1.224**	**1.462**	1.087
Test on the spectral density										
100	%	**.68**	**.69**	**.74**	**.65**	.76	.75	.81	.81	.76
	Mean	**1.462**	**1.537**	**1.049**	**1.449**	**1.154**	**1.025**	**.980**	**1.023**	**1.056**
	SD	**1.879**	**2.452**	**1.557**	**2.056**	**1.568**	**1.277**	**1.351**	**1.332**	**1.642**
1000	%	**.54**	**.66**	**.66**	**.56**	**.56**	**.60**	**.52**	**.66**	**.59**
	Mean	**2.096**	**1.693**	**1.548**	**1.894**	**1.665**	**1.800**	**1.926**	**1.736**	**1.906**
	SD	**1.726**	**1.690**	**1.596**	**1.570**	**1.631**	**1.554**	**1.610**	**1.760**	**1.603**

FIGURE 5. fGn Model: Boxplot of the bias of the estimated parameter H simulated by the *direct spectral method.* (H: Hurst parameter, N: length of the trajectory).

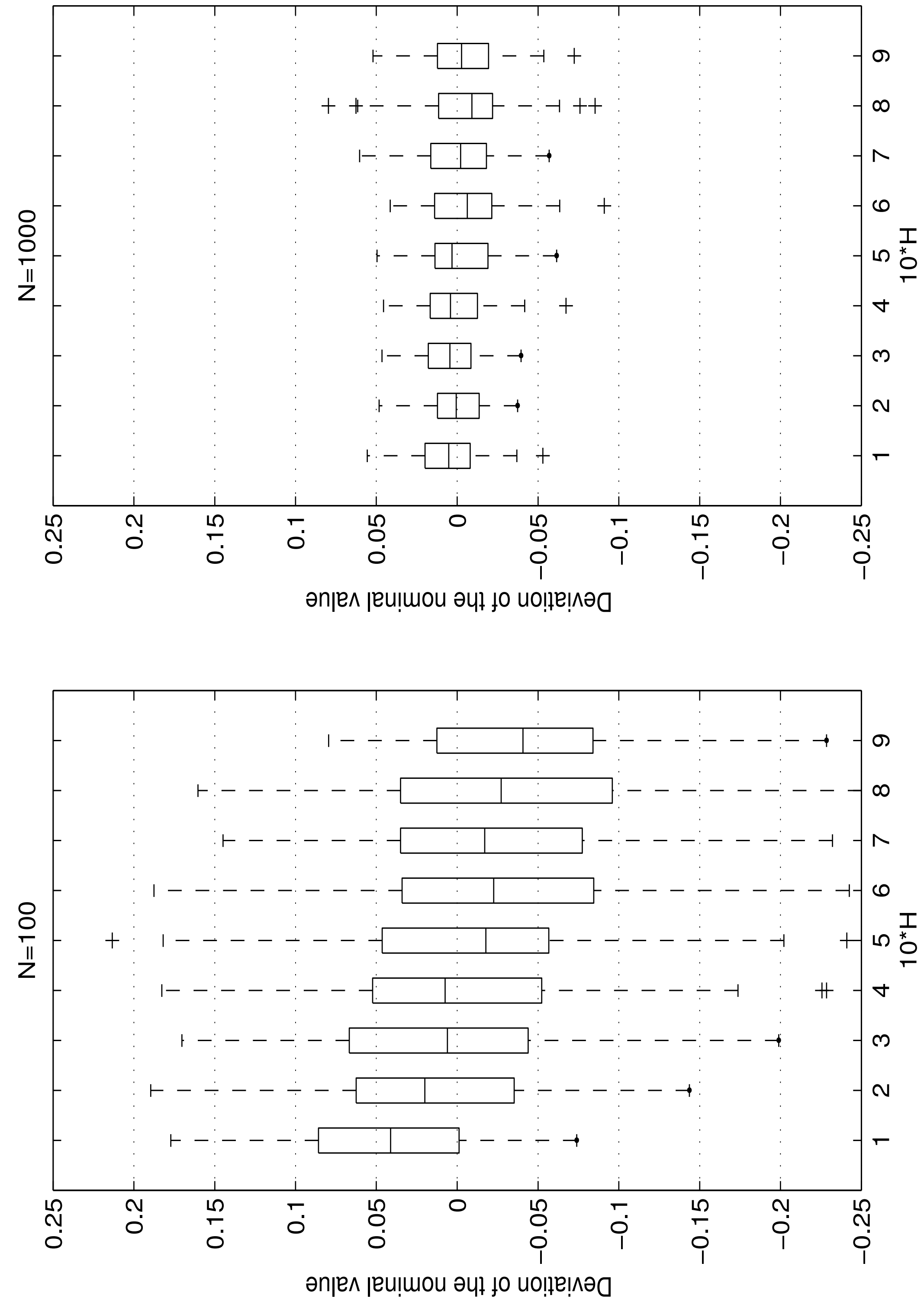

TABLE 1.6. fBm model: Percentage of sample accepted at 5% level of significance, mean and standard deviation of the statistic $\widetilde{H}_N$ and $\widetilde{T}_N$ for the *discretization of the integral* method. Values in boldface indicate that the statistic is outside the confidence band. (H: Hurst parameter, N: length of the trajectory).

N	100					1000				
H	.5	.6	.7	.8	.9	.5	.6	.7	.8	.9
Whittle estimation										
%	.94	.92	.90	.94	.94	.94	**.88**	**.88**	.95	.90
Mean	-.127	**-.351**	**-.397**	**-.306**	**-.241**	-.116	**-.663**	**-.693**	**-.238**	**.640**
SD	1.096	1.095	1.089	1.059	.912	1.029	1.027	1.028	1.026	.989
Test on the spectral density										
%	.94	.95	.95	.95	.95	.97	.94	.94	.96	.92
mean	.129	.121	.112	.091	.101	.107	**.217**	**.252**	.146	**.285**
SD	**1.227**	**1.200**	**1.206**	**1.234**	**1.283**	1.040	1.037	1.047	1.040	**1.189**

FIGURE 6. fBm model: Boxplot of the bias of the estimated parameter simulated by the *discretization of the integral* method. (H: Hurst parameter, N: length of the trajectory).

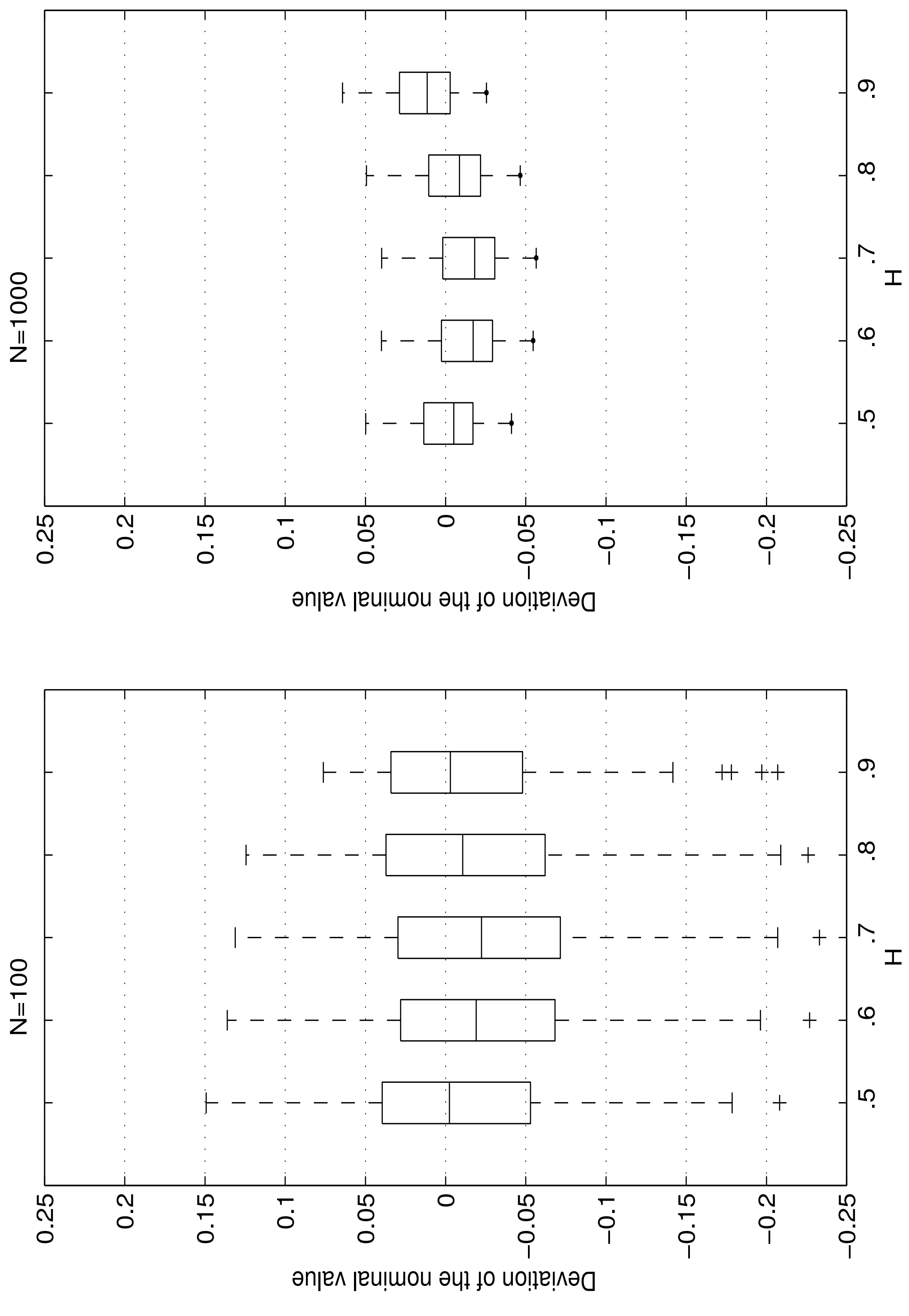

TABLE 1.7. fBm model: Percentage of sample accepted at 5% level of significance, mean and standard deviation of the statistic $\widetilde{H}_N$ and $\widetilde{T}_N$ for the *Pseudo-Markovian representation* method. Values in boldface indicate that the statistic is outside the confidence band. (H: Hurst parameter, N: length of the trajectory).

N	100				1000			
H	.1	.2	.3	.4	.1	.2	.3	.4
Whittle estimation								
%	**.87**	.93	**.88**	.90	**.57**	**.83**	**.87**	.94
Mean	**.868**	**.201**	**.230**	-.048	**1.750**	**.970**	**.530**	.1842
SD	**1.144**	1.098	**1.163**	**1.171**	**1.142**	1.038	1.130	1.019
Test on the spectral density								
%	**.89**	.94	.93	.97	**.82**	.91	.95	.92
Mean	**.287**	.094	.051	-.008	**.833**	**.285**	**.309**	.045
SD	1.098	**1.185**	**1.156**	.937	**1.489**	**1.259**	**1.137**	**1.138**

FIGURE 7. fBm model: Boxplot of the bias of the estimated parameter H simulated by the *Pseudo-Markovian representation method.* (H: Hurst parameter, N: length of the trajectory).

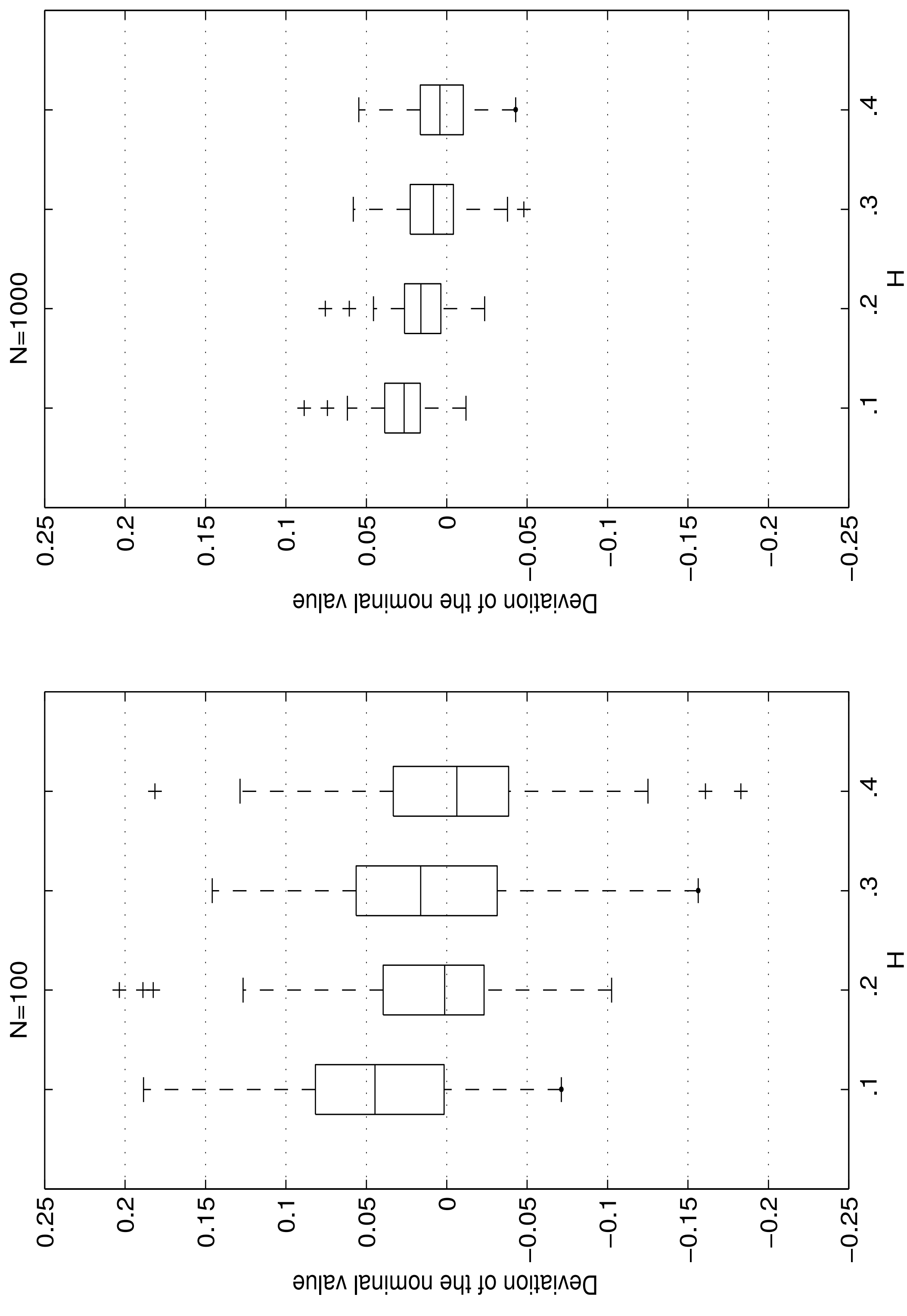

TABLE 1.8. FARIMA[0,d,0] model: Percentage of sample accepted at 5% level of significance, mean and standard deviation of the statistic $\widetilde{d}_N$ and $\widetilde{T}_n$ for *aggregation of autoregressive AR[1] processes.* Values in boldface indicate that the statistic is outside the confidence band. (d: long-range dependence parameter, N: length of the trajectory, M: number of simulated AR processes).

	d	.05			.25			.45		
N	M	50	100	1000	50	100	1000	50	100	1000
		Whittle estimation								
100	%	.91	**.89**	.90	**.89**	**.88**	.92	**.84**	**.87**	.91
	Mean	-.145	-.143	-.170	-.102	-.107	-.106	.056	.079	.009
	SD	**1.297**	**1.272**	**1.234**	**1.320**	**1.237**	**1.183**	**1.401**	**1.253**	**1.165**
1000	%	**.85**	**.86**	.95	**.58**	**.63**	.92	**.64**	**.73**	.95
	Mean	-.014	**-.208**	-.154	**.285**	-.111	-.074	-.044	.148	-.087
	SD	**1.445**	**1.414**	1.083	**2.356**	**1.940**	**1.157**	**2.223**	**1.688**	1.062
		Test on the spectral density								
100	%	.92	.91	.92	.93	.93	.93	**.88**	.93	.94
	Mean	**.315**	**.320**	**.307**	**.369**	**.316**	**.252**	**.396**	**.245**	**.205**
	SD	**1.228**	**1.191**	**1.205**	**1.190**	**1.149**	**1.141**	**1.314**	1.075	1.120
1000	%	.91	.95	.98	**.77**	**.88**	.96	**.81**	.90	.97
	Mean	**.462**	**.266**	.078	**1.390**	**.649**	.112	**1.251**	**.451**	.070
	SD	**1.565**	.984	**.841**	**2.522**	1.076	.914	**3.140**	**1.287**	.875

FIGURE 8. FARIMA[0,d,0] model: Boxplot of the bias of the estimated parameter d simulated by *aggregation of autoregressive* $AR[1]$ *processes*. (d: long-range dependence parameter, N: length of the trajectory, M: number of simulated AR processes).

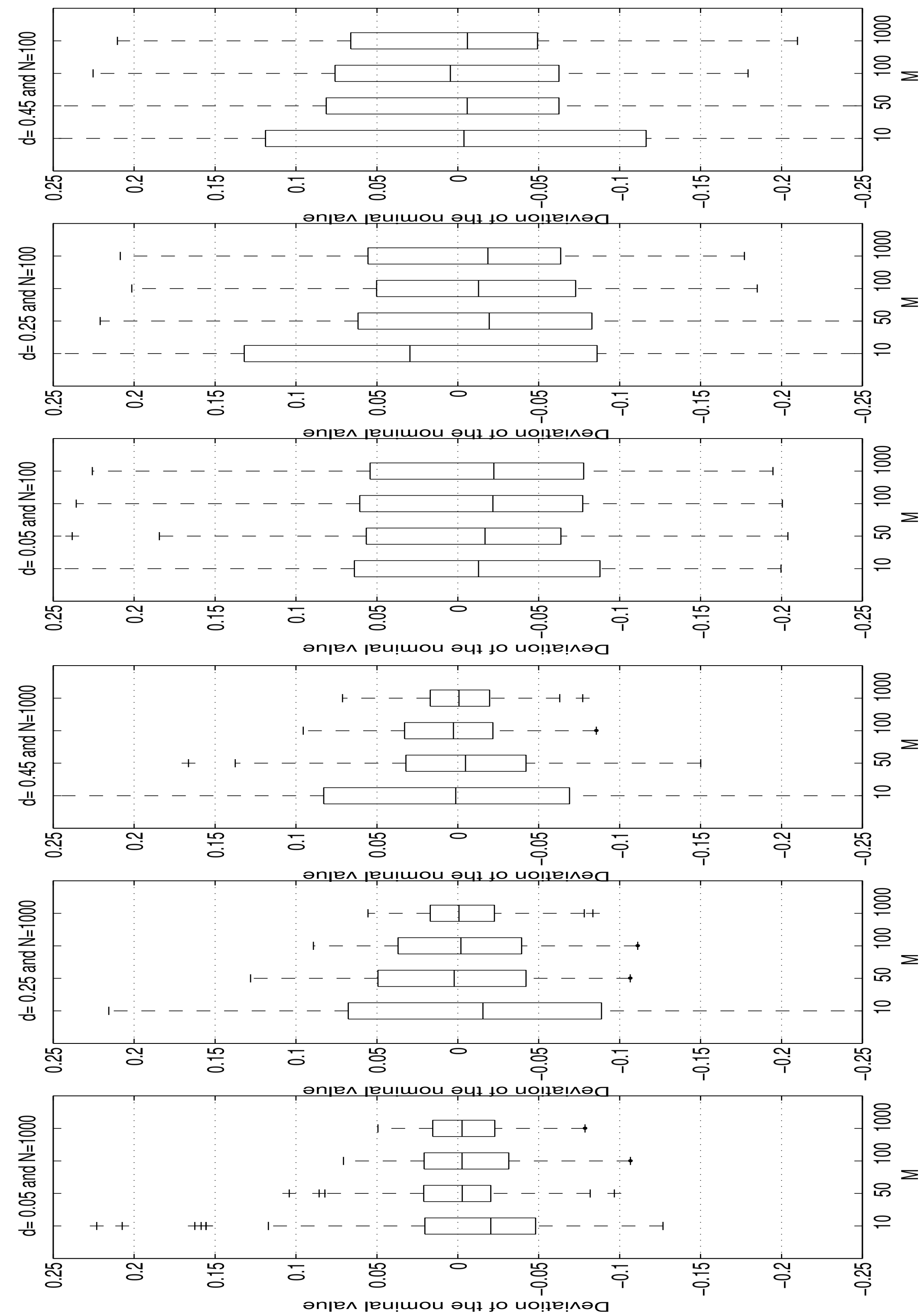

TABLE 1.9. FARIMA[0,d,0] model: Percentage of sample accepted at 5% level of significance, mean and standard deviation of the statistic $\widetilde{d}_N$ and $\widetilde{T}_n$ for the *truncated moving average expansion* algorithm. Values in boldface indicate that the statistic is outside the confidence band. (d: long-range dependence parameter, N: length of the trajectory, M: number of terms).

	d	.05			.25			.45		
N	M	50	100	1000	50	100	1000	50	100	1000
		Whittle estimation								
100	%	**.85**	.92	.93	**.85**	.92	.92	**.84**	.92	.92
	Mean	**-.266**	-.169	-.175	-.164	-.105	-.134	.078	.058	-.050
	SD	1.300	1.225	1.256	**1.324**	**1.175**	1.224	**1.292**	1.130	**1.194**
1000	%	.92	.95	.92	.90	.95	.92	.90	.94	.93
	Mean	-.147	-.106	.002	-.012	.014	.042	**.292**	**.283**	.194
	SD	1.130	1.103	1.020	1.140	1.078	1.030	**1.160**	1.068	1.054
		Test on the spectral density								
100	%	.96	.93	.91	.96	.93	.92	.93	.95	.92
	Mean	.181	**.300**	**.244**	.161	**.251**	**.209**	.136	.176	.167
	SD	.954	**1.195**	**1.191**	.945	**1.144**	**1.165**	.987	1.072	1.130
1000	%	.97	.99	.97	.96	.98	.97	.94	.97	.97
	Mean	-.008	.087	-.007	-.005	.076	-.005	.086	.107	.003
	SD	.885	**.848**	.925	.894	.857	.918	.981	.906	.907

FIGURE 9. FARIMA[0,d,0] model: Boxplot of the bias of the estimated parameter d simulated by a *truncated moving average expansion* algorithm. (d: long-range dependence parameter, N: length of the trajectory, M: number of terms).

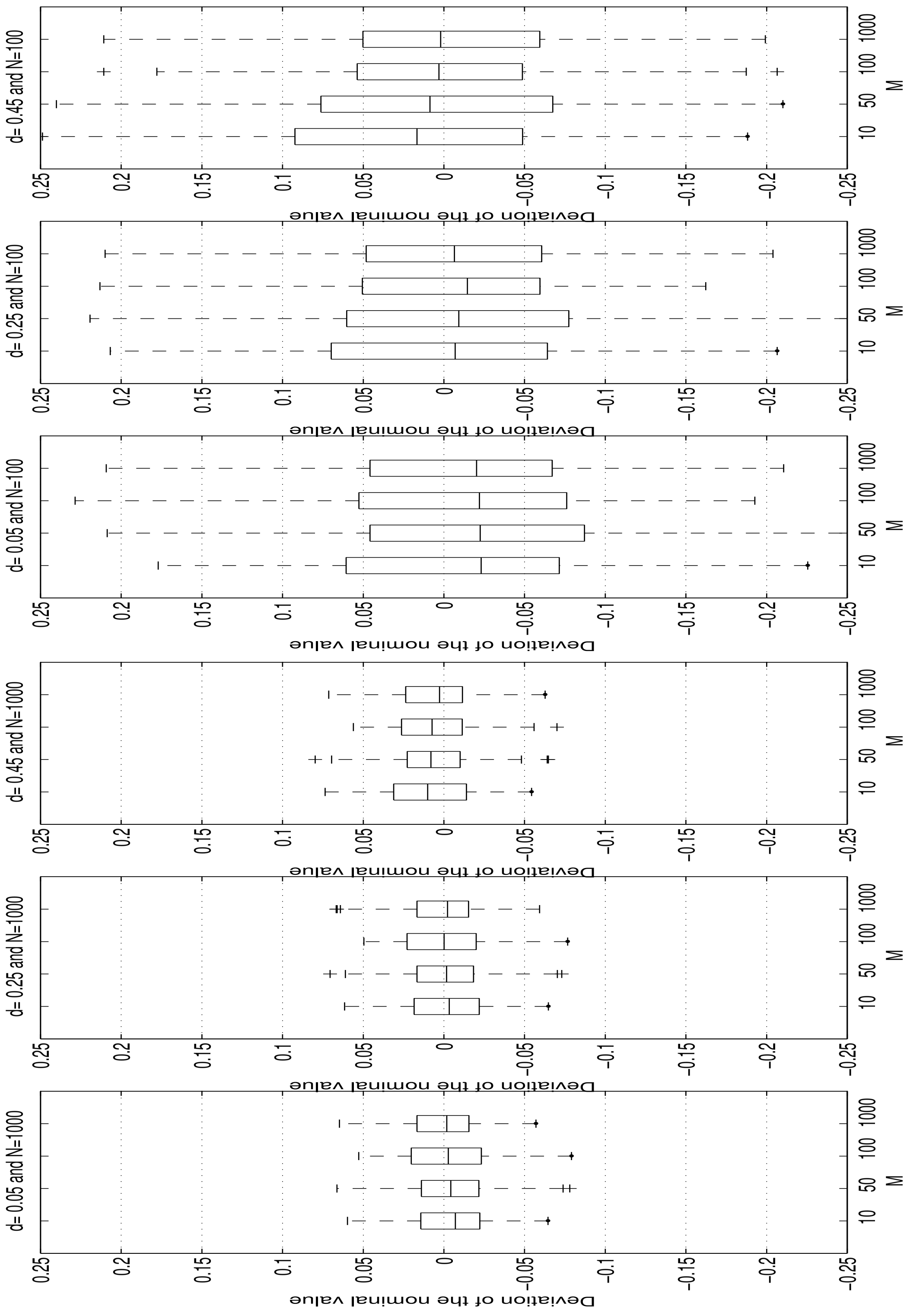

TABLE 1.10. fBm model: Percentage of sample accepted at 5% level of significance, mean and standard deviation of the statistic $\widetilde{H}_N$ and $\widetilde{T}_N$ for the *expansion in a wavelet basis* with subsampling step. (H: Hurst parameter, $N_s = N/\tau$: length of the trajectory after the subsampling step with a lag value of $\tau = 10$).

N_s	H	.1	.2	.3	.4	.5	.6	.7	.8	.9
				Whittle estimation						
100	%	**.70**	**.85**	.93	.90	.90	.92	.94	.94	.92
	mean	**1.201**	**.623**	.143	.002	-.076	-.130	-.122	-.161	-.161
	SD	1.105	**1.177**	1.016	1.133	1.125	1.138	.986	1.080	.936
1000	%	**.30**	**.71**	**.87**	.94	.92	.96	.95	.97	.96
	Mean	**2.473**	**1.296**	**.580**	.193	.021	-.008	-.150	-.107	-.182
	SD	1.078	1.074	1.070	1.005	1.068	1.017	1.058	1.008	.967
				Test on the spectral density						
100	%	**.89**	.94	.96	.97	.92	.97	.94	.95	.92
	mean	**.395**	.127	.003	.089	.360	.042	.012	.138	.117
	SD	**1.248**	1.062	1.157	.951	1.272	1.017	1.133	.982	.990
1000	%	**.660**	.93	.94	.93	.92	.93	.95	.94	.95
	Mean	**1.460**	**.388**	.005	.177	.111	.140	-.106	-.038	-.047
	SD	**1.581**	1.102	1.068	1.046	1.139	1.097	1.081	.980	.986

FIGURE 10. fBm model: Boxplot of the bias of the estimated parameter H simulated by the *expansion in a wavelet basis* method with the subsampling step.(H: Hurst parameter, $N_s = N/\tau$: length of the trajectory after the subsampling step with a lag value of $\tau = 10$).

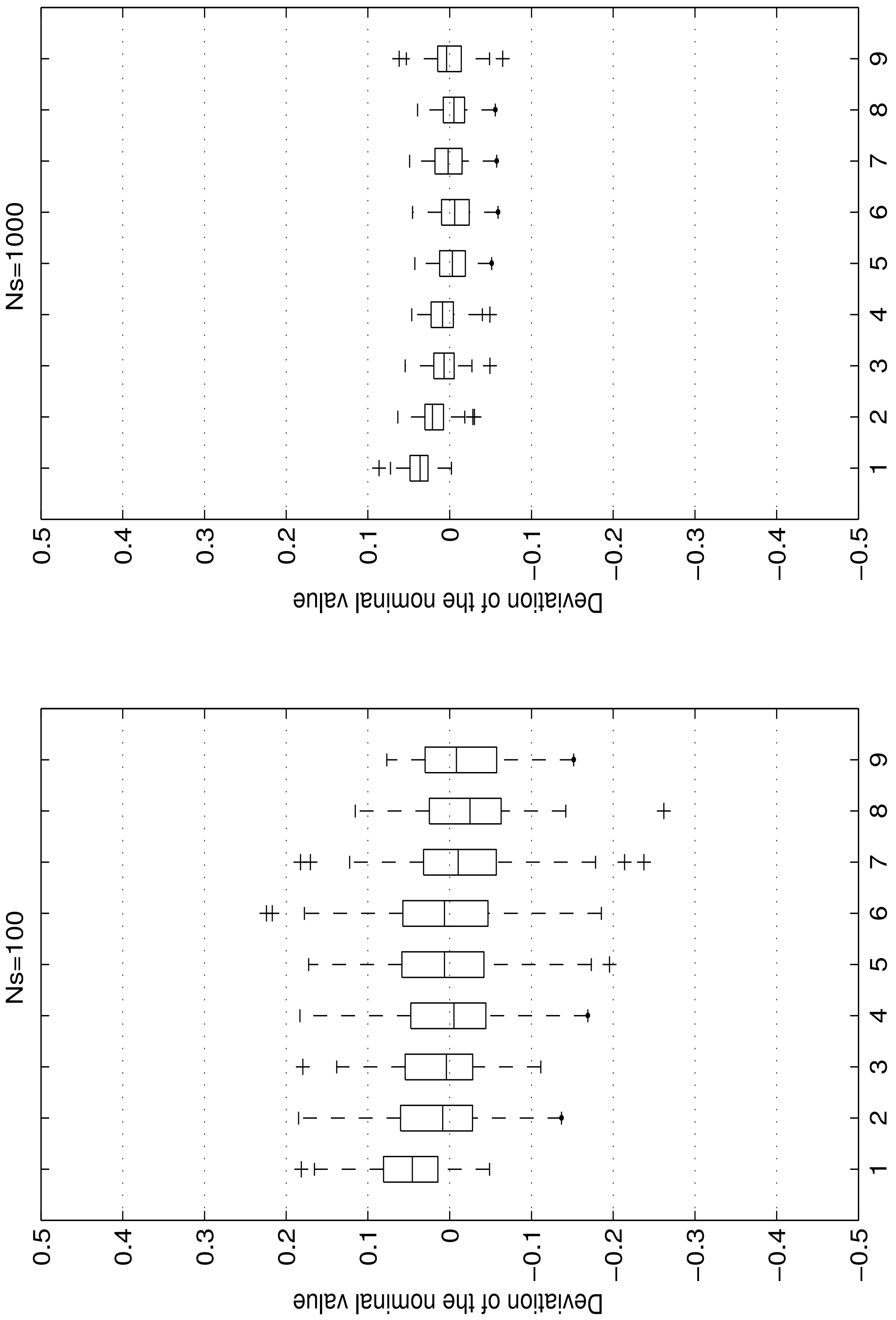

FIGURE 11. fBm model: Boxplot of the bias of the estimated parameter H ([top]: Local Whittle estimator ; [bottom]: Whittle estimator). Simulated by the *expansion in a wavelet basis* method using the same data as in Figure 10, but without subsampling. (H: Hurst parameter, N: length of the trajectory).

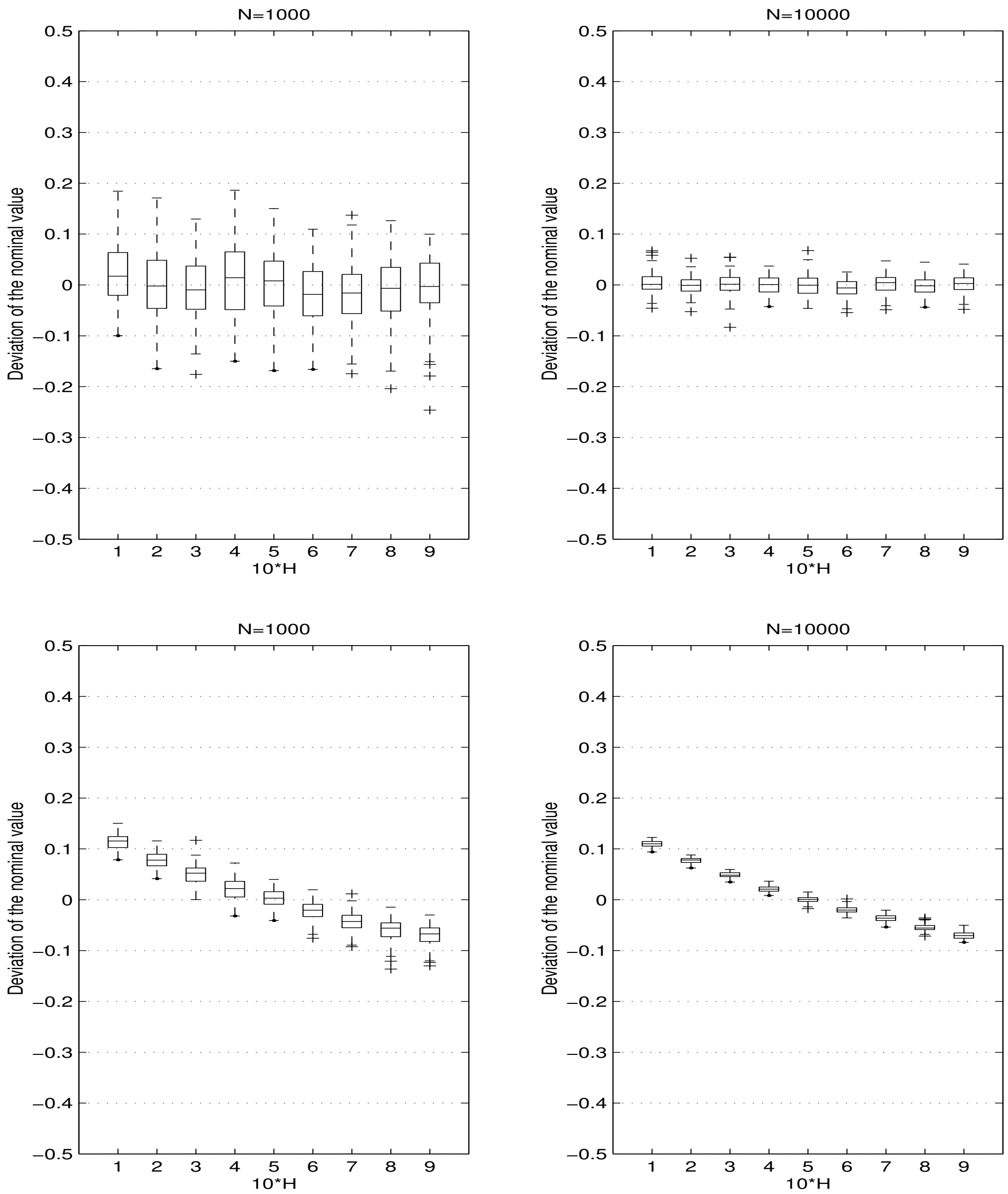

FIGURE 12. fBm model: Ratio between the estimated and the true spectral density of the increments of fBm, when fBM is simulated by the *expansion in a wavelet basis*. [left]: without subsampling [bottom]: with subsampling. (H: Hurst parameter, $N = 10000$ and $\tau = 10$).

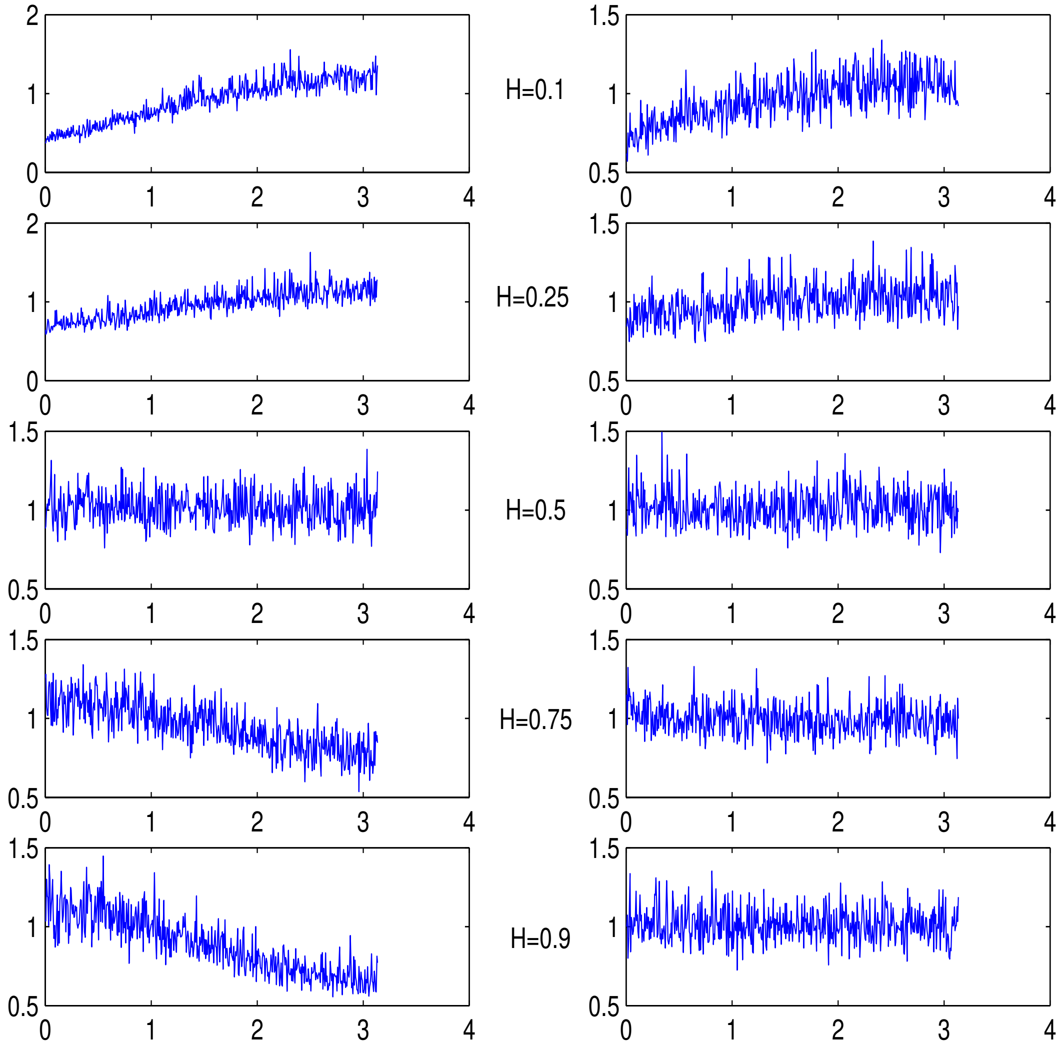

TABLE 1.11. fBm model: Percentage of sample accepted at 5% level of significance, mean and standard deviation of the statistic $\widetilde{H}_N$ and $\widetilde{T}_N$ for the *random midpoint displacement*. (H: Hurst parameter, $N_s = N/\tau$: length of the trajectory after the subsampling step with a lag value of $\tau = 10$).

N_s	H	.1	.2	.3	.4	.5	.6	.7	.8	.9
					Whittle estimation					
100	%	**.15**	**.49**	**.77**	.91	.93	.93	**.87**	**.77**	**.65**
	MEAN	**3.16**	**2.07**	**1.17**	**.469**	-.089	**-.553**	**-.957**	**-1.33**	**-1.712**
	SD	**1.229**	**1.199**	**1.141**	1.089	1.048	1.015	.984	.951	.901
1000	%	**0**	**0**	**.10**	**.73**	.93	**.69**	**.30**	**.16**	**.06**
	MEAN	**9.29**	**6.05**	**3.42**	**1.42**	**-.122**	**-1.34**	**-2.35**	**-3.20**	**-3.95**
	SD	**1.249**	**1.213**	**1.164**	1.128	1.108	1.100	1.105	1.123	**1.151**
					Test on the spectral density					
100	%	**.30**	**.64**	**.82**	.93	.94	.93	.90	**.84**	**.75**
	MEAN	4.10	1.80	.693	.230	.123	.218	.434	.728	1.08
	SD	**3.414**	**2.276**	**1.591**	**1.200**	1.040	1.051	**1.160**	**1.316**	**1.491**
1000	%	**0**	**0**	**.36**	**.85**	.93	**.86**	**.64**	**.39**	**.15**
	MEAN	**23.3**	**9.39**	**3.07**	**.584**	.050	**.558**	**1.64**	**3.07**	**4.71**
	SD	**7.610**	**4.162**	**2.349**	**1.340**	1.005	**1.252**	**1.693**	**2.154**	2.601

FIGURE 13. fBm model: Boxplot of the bias of the estimated parameter H ([top]: Local Whittle estimator ; [bottom]: Whittle estimator). Simulated by the *random midpoint displacement* method. (H: Hurst parameter, N: length of the trajectory).

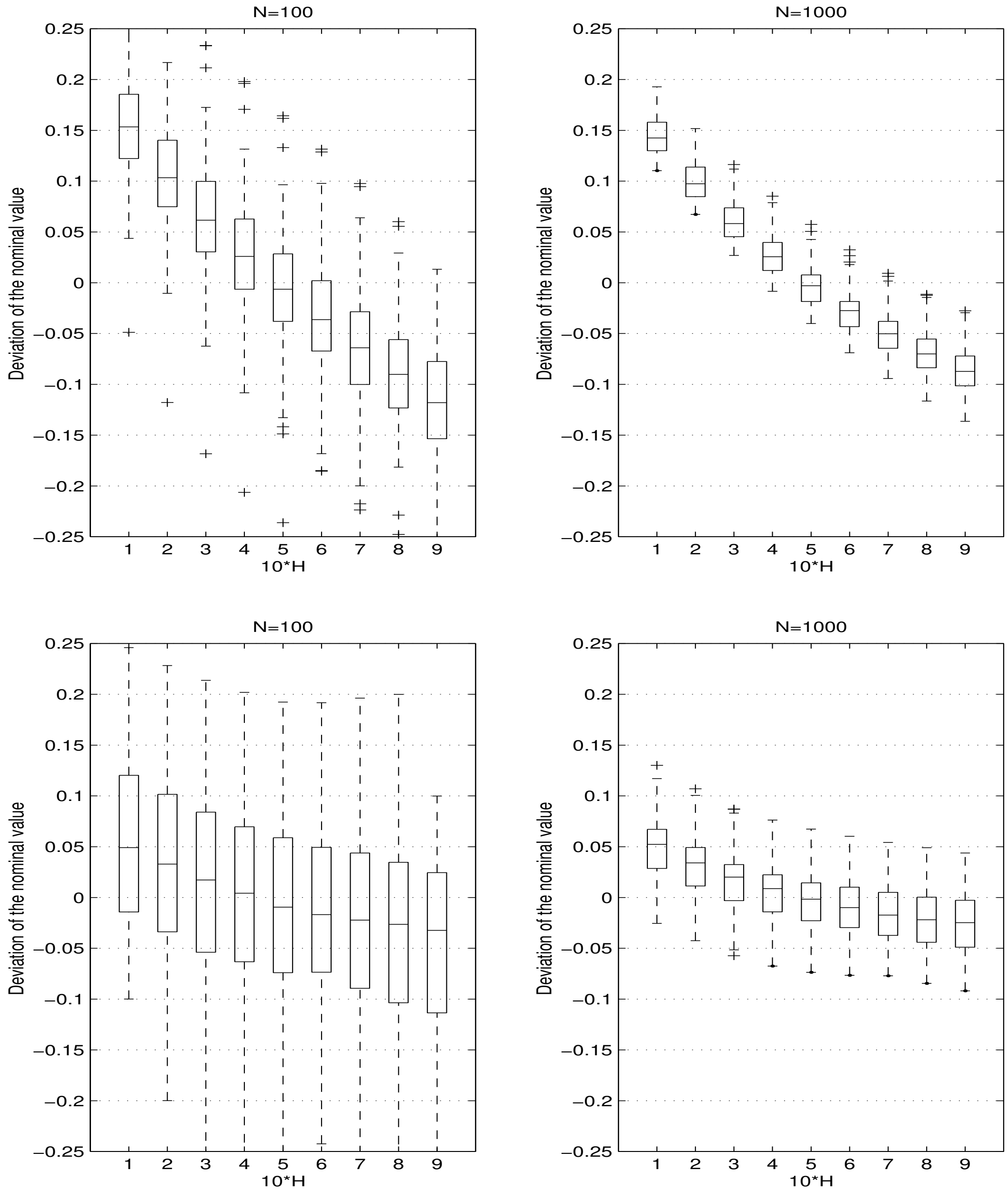

TABLE 1.12. Summary of simulation results. By "exact" we mean that the method involves no approximations. *Test 1*: Whittle estimation. *Test 2*: Test on the spectral density.

Name of the generator	Exact	Easy to implement	Time consuming	Gaussian restriction	Test 1	Test 2	Models studied
Durbin-Levinson §2.2	Yes	Yes	Yes	Yes	+	+	fGn
Circulant matrix §2.3	Yes.	Yes	No	Yes	++	++	fGn FARIMA
Lowen's circulant method §2.4	Yes	Yes	No	Yes	++	++	fBm ($H < .5$)
Direct spectral method §2.5	No	Yes	No	Yes	-	- -	fGn
Discretization of the integral §2.6	No	Yes	Yes	No	-	+	fBm ($H \geq .5$)
Pseudo- markovian representation §2.7	No	Yes	Yes	No	-	-	fBm ($H \leq .5$)
Aggregation of AR process §2.9	No	Yes	No	No	+	+	FARIMA
MA expansions §2.10	No	Yes	No	No	++	++	FARIMA
Decomposition in a wavelet basis §2.11	No	No	Yes	No	++	++	fBm
random midpoint displacement §2.12	No	Yes	No	Yes	- -	- -	fBm