Fast Self-Similar Teletraffic Generation Based on FGN and Wavelets

Hae-Duck J. Jeong†, Don McNickle‡ and Krzysztof Pawlikowski†
Department of †Computer Science and ‡Management
University of Canterbury, Christchurch
New Zealand
joshua@cosc.canterbury.ac.nz

Abstract

It is generally accepted that self-similar *(or* fractal*) processes may provide better models of teletraffic in modern computer networks than Poisson processes. Thus, an important requirement for conducting simulation studies of telecommunication networks is the ability to generate long synthetic stochastic self-similar sequences. A new generator of pseudo-random self-similar sequences, based on the fractional Gaussian noise (FGN) and a wavelet transform, is proposed and analysed in this paper. Specifically, this generator uses Daubechies wavelets. The motivation behind this selection of wavelets is that Daubechies wavelets lead to more accurate results by better matching the self-similar structure of long range dependent processes, than other types of wavelets. The statistical accuracy and time required to produce sequences of a given (long) length are experimentally studied. This generator shows a high level of accuracy of the output data (in the sense of the Hurst parameter) and is fast. Its theoretical algorithmic complexity is* $O(n)$.

1. Introduction

The search for accurate mathematical models of data streams in modern computer networks has attracted a considerable amount of interest in the last few years.

Several recent teletraffic studies of local and wide area networks, including the World Wide Web, have shown that commonly used teletraffic models, based on Poisson or related processes, are not able to capture the self-similar (or fractal) nature of teletraffic [16], [17], [24], [26], especially when these networks are engaged in such sophisticated services as variable-bit-rate (VBR) video transmission [8], [14], [25]. The properties of teletraffic in such scenarios are very different from both the properties of conventional models of telephone traffic and the traditional models of data traffic generated by computers.

The use of traditional models of teletraffic can result in overly optimistic estimates of performance of computer networks, insufficient allocation of communication and data processing resources, and difficulties in ensuring the quality of service expected by network users [2], [22], [24]. On the other hand, if the strongly correlated character of teletraffic is explicitly taken into account, this can also lead to more efficient traffic control mechanisms.

Several methods for generating pseudo-random self-similar sequences have been proposed. They include methods based on fast fractional Gaussian noise [19], fractional ARIMA processes [10], the $M/G/\infty$ queue model [14], [16], autoregressive processes [4], spatial renewal processes [28], etc. Some of them generate asymptotically self-similar sequences and require large amounts of CPU time. For example, Hosking's method [10], based on the F-ARIMA$(0, d, 0)$ process, needs 1.5 hours to produce a self-similar sequence with 131,072 (2^{17}) numbers on a Pentium II [12], [16]. It requires $O(n^2)$ computations to generate n numbers. Even though exact methods of generation of self-similar sequences exist (for example: [19]), they are only fast enough for short sequences. They are usually inappropriate for generating long sequences because they require multiple passes along generated sequences. To overcome this, approximate methods for generation of self-similar sequences in simulation studies of computer networks have been proposed [15], [23].

The evaluation of our generator based on Daubechies wavelets (DW) concentrates on two aspects: (i) how accurately a self-similar process can be generated; and (ii) how quickly the method generates long self-similar sequences. Our method, based on the fractional Gaussian noise (FGN) and Daubechies wavelets, will be called the FGN-DW method.

A summary of the basic properties of self-similar processes is given in Section 2. Section 3 describes the spectral density of FGN processes, while a discrete wavelet transform (DWT) for synthesising approximate FGN is presented in Section 4. In Section 5, a generator of pseudo-random self-similar sequences, based on FGN and DW, is described. Numerical results of analysis of sequences generated by this generator are discussed in Section 6.

2. Self-Similar Processes and Their Properties

Basic definitions of self-similar processes are as follows:

A continuous-time stochastic process $\{X_t\}$ is strongly *self-similar* with a self-similarity parameter $H(0 < H < 1)$, know as the Hurst parameter, if for any positive stretching factor c, $c > 0$, the rescaled process with time scale $ct, c^{-H}X_{ct}$, is equal in distribution to the original process $\{X_t\}$ [3]. This means that, for any sequence of time points $t_1, t_2, \ldots, t_n$, and for all $c > 0$, $\{c^{-H}X_{ct_1}, c^{-H}X_{ct_2}, \cdots, c^{-H}X_{ct_n}\}$ has the same distribution as $\{X_{t_1}, X_{t_2}, \cdots, X_{t_n}\}$.

In the discrete-time case, let $\{X_k\} = \{X_k : k = 0, 1, 2, \cdots\}$ be a (discrete-time) stationary process with mean μ, variance σ^2, and autocorrelation function (ACF) $\{\rho_k\}$, for $k = 0, 1, 2, \cdots$, and let $\{X_k^{(m)}\}_{k=1}^{\infty} = \{X_1^{(m)}, X_2^{(m)}, \cdots\}$, $m = 1, 2, 3, \cdots$, be a sequence of batch means, that is,

$$X_k^{(m)} = (X_{km-m+1} + \cdots + X_{km})/m, k \geq 1.$$

The process $\{X_k\}$ with $\rho_k \rightarrow k^{-\beta}$, as $k \rightarrow \infty, 0 < \beta < 1$, is called *exactly self-similar* with $H = 1 - (\beta/2)$, if $\rho_k^{(m)} = \rho_k$, for any $m = 1, 2, 3, \cdots$. In other words, the process $\{X_k\}$ and the averaged processes $\{X_k^{(m)}\}$, $m \geq 1$, have an identical correlation structure. The process $\{X_k\}$ is *asymptotically self-similar* with $H = 1 - (\beta/2)$, if $\rho_k^{(m)} \rightarrow \rho_k$, as $m \rightarrow \infty$.

The most frequently studied models of self-similar traffic belong either to the class of fractional autoregressive integrated moving-average (F-ARIMA) processes or to the class of fractional Gaussian noise processes; see [10], [16], [23]. F-ARIMA(p, d, q) processes were introduced by Hosking [10], who showed that they are asymptotically self-similar with Hurst parameter $H = d + \frac{1}{2}$, as long as $0 < d < \frac{1}{2}$, where p is the order of autoregression in the ARIMA process and q is the order of the moving average in the ARIMA process. For the second class, the FGN process is the incremental process $\{Y_k\} = \{X_k - X_{k-1}\}, k \geq 0$, where $\{X_k\}$ designates a fractional Brownian motion (FBM) random process. This process is a (discrete-time) stationary Gaussian process with mean μ, variance σ^2 and $\{\rho_k\} = \{\frac{1}{2}(|k+1|^{2H} - 2|k|^{2H} + |k-1|^{2H})\}, \quad k > 0$. An FBM process, which is the sum of FGN increments, is characterised by three properties [20]: (i) it is a continuous zero-mean Gaussian process $\{X_t\} = \{X_s : s \geq 0$ and $0 < H < 1\}$ with ACF given by $\rho_{s,t} = \frac{1}{2}(s^{2H} + t^{2H} - |s - t|^{2H})$ where *s* is time lag and *t* is time; (ii) its increments $\{X_t - X_{t-1}\}$ form a stationary random process; (iii) it is self-similar with Hurst parameter H, that is, for all $c > 0$, $\{X_{ct}\} = \{c^H X_t\}$, in the sense that, if time is changed by the ratio c, then the distribution of $\{X_t\}$ and $\{c^H X_t\}$ remains the same.

The main properties of self-similar processes include ([3], [5], [16]):

- *Slowly decaying variance*. The variance of the sample mean decreases more slowly than the reciprocal of the sample size, i.e., $Var[\{X_k^{(m)}\}] \rightarrow c_1 m^{-\beta_1}$, as $m \rightarrow \infty$, where c_1 is a constant and $0 < \beta_1 < 1$.

- *Long-range dependence*. A process $\{X_k\}$ is called a stationary process with *long-range dependence (LRD)* if its ACF $\{\rho_k\}$ is non-summable, i.e., $\sum_{k=0}^{\infty} \rho_k = \infty$. The speed of decay of autocorrelations is more hyperbolic than exponential.

- *Hurst effect*. Self-similarity manifests itself by a straight line of slope β_2, $0.5 < \beta_2 < 1$, on a log-log plot of the R/S statistic. For a given set of numbers $\{X_1, X_2, \cdots, X_n\}$ with sample mean $\mu = E\{X_i\}$ and sample variance $S^2(n) = E\{(X_i - \mu)^2\}$, the Hurst parameter H is presented by the *rescaled adjusted range* $\frac{R(n)}{S(n)}$ (or R/S *statistic*), where

$$R(n) = \max\{\sum_{i=1}^{k}(X_i - \mu), 1 \leq k \leq n\}$$

$$- \min\{\sum_{i=1}^{k}(X_i - \mu), 1 \leq k \leq n\}$$

  and S is estimated by $S(n) = \sqrt{E\{(X_i - \mu)^2\}}$. Hurst found empirically that for many time series observed in nature, the expected value of $\frac{R(n)}{S(n)}$ asymptotically satisfies the power law relation: $E[\frac{R(n)}{S(n)}] \rightarrow c_2 n^H$, as $n \rightarrow \infty$, with $0.5 < H < 1$ and c_2 is a finite positive constant [3].

- $1/f$-*noise*. The spectral density $f(\lambda; H)$ obeys a power law near the origin, i.e., $f(\lambda; H) \rightarrow c_3 \lambda^{\beta_3}$, as $\lambda \rightarrow 0$, where c_3 is a finite positive constant and $H = \frac{1-\beta_3}{2}$.

We will use these properties to investigate characteristics of generated self-similar sequences.

3. Spectral Density of FGN Processes

In our generator, numbers representing the spectral density function of FGN are obtained by applying appropriate transformations to originally uniformly distributed pseudo-random numbers. The spectral density $f(\lambda; H)$ of an FGN process is given by

$$f(\lambda; H) = 2c_f(1 - cos(\lambda)) \sum_{k=-\infty}^{\infty} |2\pi k + \lambda|^{-2H-1} \quad (1)$$

for $0 < H < 1$ and $-\pi \leq \lambda \leq \pi$, where

$$c_f = \sigma^2 (2\pi)^{-1} sin(\pi H)\Gamma(2H + 1)$$

and σ^2 = Var[X_k]; see [3].

The main difficulty with using Equation (1) to compute the spectral density is the vexing infinite summation. The approximation of the above $f(\lambda; H)$ is given in [3] as

$$f(\lambda; H) = c_f |\lambda|^{1-2H} + O(|\lambda|^{min(3-2H,2)}) \qquad (2)$$

where $O(\cdot)$ represents the residual error.

This formula is used by us in the generator of self-similar sequences proposed in this paper. Another generator of self-similar sequences based on FGN was also proposed by Paxson [23], but his method was based on a more complicated approximation of $f(\lambda; H)$ than this one given by Equation (2).

4. Discrete Wavelet Transform

Our method for generating synthetic self-similar FGN sequences in a time domain is based on the discrete wavelet transform (DWT). It has been shown that wavelets can provide compact representations for a class of FGN processes [7], [13]. This is because the structure of wavelets naturally matches the self-similar structure of the long range dependent processes [1], [6], [29].

Wavelets are complete orthonormal bases which can be used to represent a random time series in two domains: time and frequency. In Hilbert space $L^2(R)$, scaled and shifted functions $\psi_{j,m}(k)$ of wavelets can be represented as

$$\psi_{j,m}(k) = 2^{-j/2}\psi_0(2^{-j}k - m)$$

where j and m are positive integers [18]. Since such wavelets are obtained by scaling and shifting a single function, $\psi_0(k)$, this function is called the mother wavelet. Moreover, all base functions $\psi_{j,m}(k)$ have the same shape as the mother wavelet and therefore are self-similar with each other.

For our generator, we chose Daubechies wavelets, which belong to the class of orthonormal wavelets, because they produce more accurate coefficients of wavelets than Haar wavelets (for more detailed discussions, see also [6], [29]; and our results of the comparison in Section 6). They are defined as

$$\psi(k) = \sum_{i=-2^S+1}^{1} (-1)^i p_{1-i}\phi(2k - i),$$

where $\{p_i\}$ is the two-scale sequence of $\phi(k)$ and

$$\phi(k) = \sum_{i=0}^{2^S} p_i\phi(2k - i).$$

A discrete-time process $\{X_k\}$ can be represented through the inverse DWT of

$$\{X_k\} = \{\sum_{j=1}^{S} \sum_{m=0}^{2^{S-j}-1} d_{j,m}\psi_{j,m}(k)\},$$

where $0 \leq k < 2^S$; and S is a positive integer which characterises the limited resolution in time; $d_{j,m}$'s are wavelet coefficients which can be obtained through the DWT, since $d_{j,m} = \sum_{t=0}^{2^S-1} X_k\psi_{j,m}(k)$.

5. A Fast Algorithm for Generating Self-Similar Teletraffic

We claim that the FGN and Daubechies wavelet-based transformation is sufficiently fast for practical generation of synthetic self-similar sequences to be used as simulation input data. The general strategy behind our method, called FGN-DW, is the same as in [23]. The algorithm consists of the following steps:

Step.1 Calculate a sequence of values $\{f_1, f_2, \cdots, f_n\}$ using Equation (2), $f_i = \hat{f}(\frac{\pi i}{n})$, corresponding to the spectral density of an FGN process for frequencies f_i ranging between $\frac{\pi}{n}$ and π.

Step.2 Multiply $\{f_i\}$ by realisations of an independent exponential random variable with a mean of 1 to obtain $\{\hat{f}_i\}$.

Step.3 Generate a sequence $\{y_1, y_2, \cdots, y_p\}$ of complex numbers such that $|y_i| = \sqrt{\hat{f}_i}$ and the phase of y_i is uniformly distributed between 0 and 2π. This random phase technique, taken from [27], preserves the spectral density corresponding to $\{\hat{f}_i\}$, but ensures that different sequences generated using the method will be independent. It also ensures that the marginal distribution of the final sequence is normal. The phase randomisation makes the different frequency components independent and was also applied in [23].

Step.4 Calculate two synthetic coefficients of orthonormal Daubechies wavelets which are used in the inverse DWT [21]. Then, generate the approximately self-similar FGN sequence in the time domain by using the inverse DWT from $\{y_i\}$.

Using the above steps, the proposed FGN-DW method generates a fast and well approximated self-similar FGN process.

6. Analysis of Self-Similar Sequences

The generator of self-similar sequences of pseudo-random numbers described in Section 5 has been implemented in Matlab on a Pentium II (233 MHz, 64 MB). The mean times required for generating sequences of a given length were obtained by using the Matlab `clock` command and averaged over 30 iterations, having generated sequences of 32,768 (2^{15}), 131,072 (2^{17}), 262,144 (2^{18}), 524,288 (2^{19}) and 1,048,576 (2^{20}) numbers.

We have also analysed the accuracy of the method. For each of $H = 0.5, 0.55, 0.7, 0.9, 0.95$, the method was used to generate over 100 sample sequences of 32,768 (2^{15}) numbers starting from different random seeds. Self-similarity and marginal distributions of the generated sequences were assessed by applying the best currently available techniques. These include:

- *Anderson-Darling goodness-of-fit test*: used to show that the marginal distribution of sample sequences generated by the method is, as required, normal (or almost normal). This test is more powerful than *Kolmogorov-Smirnov* when testing against a specified normal distribution [9].

- *Periodogram plot*: used to show whether a generated sequence is LRD or not. It can be shown that if the autocorrelations are summable, then, near the origin in frequency domain, the periodogram should be scattered randomly around a constant level. If the autocorrelations are non-summable, i.e., LRD-type, the points of a sequence are scattered around a negative slope. The periodogram plot is obtained by plotting $log_{10}(periodogram)$ against $log_{10}(frequency)$. An estimate of the Hurst parameter is given by $\hat{H} = (1 - \hat{\beta}_3)/2$, where $\hat{\beta}_3$ is the slope [3].

- *R/S statistic plot*: used to estimate the Hurst parameter H from empirical data. An estimate of H is given by the asymptotic slope $\hat{\beta}_2$ of the R/S statistic plot, i.e., $\hat{H} = \hat{\beta}_2$ [3].

- *Variance-time plot*: obtained by plotting $log_{10}(Var(X^{(m)}))$ against $log_{10}(m)$ and by fitting a simple least square line through the resulting points in the plane. An estimate of the Hurst parameter is given by $\hat{H} = 1 - \hat{\beta}_1/2$ where $\hat{\beta}_1$ is the slope [3].

- *Whittle's approximate maximum likelihood estimate(MLE)*: for a more refined data analysis, used to obtain confidence intervals (CIs) for the Hurst parameter H [3]. It examines the properties in frequency domain, while the R/S statistic plot and variance-time plot focus on the time domain. Suppose $\{x_1, x_2, \cdots, x_n\}$ is a sequence of a self-similar process $\{X_k\}$ for which all parameters are known except Var[X_i] and H. Let $f(\lambda; H)$ be the spectral density of $\{X_k\}$ when normalised to have variance 1, and $I(\lambda)$ be the periodogram of $\{X_k\}$. Then to estimate H, find $\hat{H}$ that minimises the following equation: $g(\hat{H}) = \int_{-\pi}^{\pi} \frac{I(\lambda)}{f(\lambda;\hat{H})} d\lambda$.

6.1. Analysis of Accuracy

We have summarised the results of our analysis in the following table:

- The Anderson-Darling goodness-of-fit test was applied to test the normality of sample sequences. The results of the tests, executed at the 5% significance level, showed that the generated sequences could be considered as normally distributed for all but a few sequences with a high value of H; for more discussions, see in [11].

The estimates of the Hurst parameter obtained from the periodogram, the R/S statistic, the variance-time and Whittle's MLE, have been used to analyse the accuracy of the generator. The relative inaccuracy ΔH is calculated using the formula:

$\Delta H = \frac{\hat{H} - H}{H} * 100\%$, where H is the required value of the Hurst parameter and $\hat{H}$ is the empirical mean value over a number of independently generated sequences. The presented numerical results are all averaged over 100 sequences.

- The periodogram plots have slopes decreasing as H increases. The results for $H = 0.5, 0.7, 0.9$ are shown in Figure 1. The negative slopes of all our plots for $H = 0.5, 0.55, 0.7, 0.9, 0.95$ were the evidence of self-similarity. The relative inaccuracy, ΔH, of the estimated Hurst parameters of the method, assessed using a periodogram plot, is given in Table 1. We see that, in most cases, parameter H of the FGN-DW method was close to the required value, although the relative inaccuracy degrades with increasing H (but never exceeds 2%). The analysis of the periodogram shows that the FGN-DW method always produces self-similar sequences with a negatively biased $\hat{H}$.

- The plots of R/S statistic indicate the self-similar nature of the generated sequences. The results for $H = 0.5, 0.7, 0.9$ are shown in Figure 2. The relative inaccuracy, ΔH, of the estimated Hurst parameter, obtained from the R/S statistic plot, is given in Table 1. This method of analysis of H does not show that this generator has a persistently positive or negative bias of $\hat{H}$, as the periodogram plots did.

- The variance-time plots also support the claim that generated sequences are self-similar; see results for $H = 0.5, 0.7, 0.9$ in Figure 3. Table 1 gives the relative inaccuracy, ΔH, of the estimated Hurst parameters obtained by the variance-time plot. Again, the method shows high quality in the sense of the accuracy of H in generated sequences, with the relative inaccuracy increasing with the increase in H, but remaining below 8%. This time, the results suggest that the output sequences have a negatively biased $\hat{H}$.

- The results for the Whittle estimator of H with the corresponding 95% CIs $\hat{H} \pm 1.96\hat{\sigma}_{\hat{H}}$, see Table 2, show that for all input H values (but $H = 0.5$), the FGN-DW method produces sequences with so negatively biased H values that the CIs for $H \geq 0.7$ do not contain theoretical values.

Our results show that the generator produces approximately self-similar FGN sequences, with the relative inaccuracy, ΔH, increasing with the increase of H, but always staying below 8%. Apparently there is a more general problem with more detailed studies of such generators, since different methods of analysis of the Hurst parameter can give different results for the bias of $\hat{H}$ in the same output sequence. More reliable methods for assessment of self-similarity in pseudo-random sequences are needed.

6.2. Computational Complexity

The results of our experimental analysis of mean times needed by the FGN-DW generator for generating pseudo-random self-similar sequences of a given length are shown in Table 3.

The main conclusion is that the FGN-DW method is fast. Table 3 shows that 2 seconds were needed to generate a sequence of 32,768 (2^{15}) numbers, while generation of a sequence with 1,048,576 (2^{20}) numbers took 51 seconds.

The theoretical algorithmic complexity of forming spectral density, and constructing normally distributed complex numbers, is $O(1)$, while the inverse DWT is $O(n)$ [23], [29]. Thus, the time complexity of FGN-DW is also $O(n)$.

In summary, our results show that a generator of pseudo-random self-similar sequences based on FGN and DW is sufficiently fast to make it applicable in practical computer simulation studies, when long self-similar sequences of numbers are needed.

Table 1. Relative inaccuracy, ΔH, estimated from periodogram, R/S statistic and variance-time plots.

H	Periodogram	R/S Statistic	Variance-Time
.5	- 0.07 %	+7.64 %	- 0.61 %
.55	- 0.49 %	+5.34 %	- 0.99 %
.7	- 1.31 %	+0.51 %	- 2.15 %
.9	- 1.85 %	- 5.25 %	- 5.94 %
.95	- 1.93 %	- 7.10 %	- 7.54 %

Table 2. Estimated mean values of H using Whittle's MLE. Each CI is for over 100 sample sequences. 95% CIs for the means are given in parentheses.

H	Estimated Mean Values
.5	.500 (.490, .509)
.55	.542 (.532, .551)
.7	.672 (.662, .681)
.9	.851 (.842, .861)
.95	.897 (.888, .906)

6.3. Using Haar Wavelets and Daubechies Wavelets for Generation of LRD Sequences

We use the Daubechies wavelets for generation of self-similar sequences because they produce more accurate coefficients of wavelets than Haar wavelets. For more detailed discussions of Haar wavelets, see [6], [29]. Our results of comparison of sequences produced by generators based on Haar wavelets and Daubechies wavelets are shown in Table 4.

7. Conclusions

In this paper we have proposed a generator of (long) pseudo-random self-similar sequences, based on the FGN and DW transform. It appears that this generator produces approximately self-similar sequences, with the relative inaccuracy of the resulted H below 8%, if $0.5 \leq H \leq 0.95$. On the other hand, the analysis of mean times needed

Table 3. Mean running times of generators. Running times were obtained by using the Matlab `clock` command on a Pentium II (233 MHz, 64 MB); each mean is averaged over 30 iterations.

Mean Running Time	Sequence of				
	32,768 Numbers	131,072 Numbers	262,144 Numbers	524,288 Numbers	1,048,576 Numbers
	Mean running time (*minute:second*)				
$T(n)$	0:02	0:07	0:13	0:25	0:51

Table 4. Comparison of sequences generated by using Haar wavelets and Daubechies wavelets (H = 0.9).

Method	Haar	Daub(8)	Daub(16)	Daub(32)	Daub(50)
Periodogram	.855	.881	.884	.885	.884
R/S-statistic	.856	.857	.857	.858	.856
Variance-time	.844	.852	.852	.852	.849
Whittle's MLE	.826 (.817, .835)	.850 (.840, .859)	.852 (.842, .861)	.853 (.844, .862)	.852 (.843, .862)

for generating sequences of a given length shows that this generator should be recommended for practical simulation studies of telecommunication networks, since it is very fast and accurate.

References

[1] P. Abry and D. Veitch. Wavelet Analysis of Long-Range-Dependent Traffic. *IEEE Transactions on Information Theory*, 44(1):2–15, 1998.

[2] J. Beran. Statistical Methods for Data with Long Range Dependence. *Statistical Science*, 7(4):404–427, 1992.

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

[4] M. Cario and B. Nelson. Numerical Methods for Fitting and Simulating Autoregressive-to-Anything Processes. *INFORMS Journal on Computing*, 10(1):72–81, 1998.

[5] D. Cox. Long-Range Dependence: a Review. In H. David and H. David, editors, *Statistics: An Appraisal*, pages 55–74. Iowa State Statistical Library, The Iowa State University Press, 1984.

[6] I. Daubechies. *Ten Lectures on Wavelets*, volume 61 of *CBMS-NSF Regional Conference Series in Applied Mathematics*. SIAM Press, Philadelphia, Pennsylvania, 1992.

[7] P. Flandrin. Wavelet Analysis and Synthesis of Fractional Brownian Motion. *IEEE Transactions on Information Theory*, 38(2):910–917, 1992.

[8] M. Garrett and W. Willinger. Analysis, Modeling and Generation of Self-Similar VBR Video Traffic. In *Computer Communication Review, Proceedings of ACM SIGCOMM'94*, volume 24(4), pages 269–280, London, UK, Aug. 1994.

[9] J. Gibbons and S. Chakraborti. *Nonparametric Statistical Inference*. Marcel Dekker, Inc., 1992.

[10] J. Hosking. Modeling Persistence in Hydrological Time Series Using Fractional Differencing. *Water Resources Research*, 20(12):1898–1908, Dec. 1984.

[11] H.-D. Jeong, D. McNickle, and K. Pawlikowski. Fast Self-Similar Teletraffic Generation Based on FGN and Inverse DWT. Technical Report TR-COSC 03/99, Department of Computer Science, University of Canterbury, Christchurch, New Zealand, 1999.

[12] H.-D. Jeong, D. McNickle, and K. Pawlikowski. A Comparative Study of Three Self-Similar Teletraffic Generators. In *13th European Simulation Multiconference, ESM'99*, volume 1, pages 356–362, Warsaw, Poland, June 1999.

[13] L. Kaplan and C.-C. J. Kuo. Fractal Estimation from Noisy Data via Discrete Fractional Gaussian Noise (DFGN) and the Haar Basis. *IEEE Transactions on Signal Processing*, 41(12):3554–3562, Dec. 1993.

[14] M. Krunz and A. Makowski. A Source Model for VBR Video Traffic Based on $M/G/\infty$ Input Processes. In *Proceedings of IEEE INFOCOM'98*, pages 1441–1448, San Francisco, CA, USA, Mar. 1998.

[15] W.-C. Lau, A. Erramilli, J. Wang, and W. Willinger. Self-Similar Traffic Generation: the Random Midpoint Displace-

ment Algorithm and Its Properties. In *Proceedings of IEEE ICC'95*, pages 466–472, Seattle, WA, 1995.

[16] W. Leland, M. Taqqu, W. Willinger, and D. Wilson. On the Self-Similar Nature of Ethernet Traffic (Extended Version). *IEEE/ACM Transactions on Networking*, 2(1):1–15, Feb. 1994.

[17] N. Likhanov, B. Tsybakov, and N. Georganas. Analysis of an ATM Buffer with Self-Similar ("Fractal") Input Traffic. In *Proceedings of IEEE INFOCOM'95*, pages 985–992, 1995.

[18] S. Ma and C. Ji. Modeling Video Traffic in The Wavelet Domain. In *Proceedings of IEEE INFOCOM'98*, pages 201–208, San Francisco, CA, USA, 1998.

[19] B. Mandelbrot. A Fast Fractional Gaussian Noise Generator. *Water Resources Research*, 7:543–553, 1971.

[20] B. Mandelbrot and J. Wallis. Computer Experiments with Fractional Gaussian Noises. *Water Resources Research*, 5(1):228–267, 1969.

[21] C. Nartallo, N. Prelcic, S. Galan, and C. Cabanelas. Wavelet Toolbox for Matlab. Available through http://www.tsc.uvigo.es/ wavelets/uvi_wave.html, 1996.

[22] A. Neidhardt and J. Wang. The Concept of Relevant Time Scales and Its Application to Queueing Analysis of Self-Similar Traffic (or Is Hurst Naughty or Nice?). In *Proceedings ACM SIGMETRICS'98*, pages 222–232, Madison, Wisconsin, USA, Jun. 1998.

[23] V. Paxson. Fast, Approximate Synthesis of Fractional Gaussian Noise for Generating Self-Similar Network Traffic. *ACM SIGCOMM, Computer Communication Review*, 27(5):5–18, 1997.

[24] V. Paxson and S. Floyd. Wide-Area Traffic: the Failure of Poisson Modeling. *IEEE/ACM Transactions on Networking*, 3(3):226–244, Jun. 1995.

[25] O. Rose. *Traffic Modeling of Variable Bit Rate MPEG Video and Its Impacts on ATM Networks*. PhD thesis, Bayerische Julius-Maximilians-Universität Würzburg, 1997.

[26] B. Ryu. *Fractal Network Traffic: from Understanding to Implications*. PhD thesis, Graduate School of Arts and Science, Columbia University, 1996.

[27] S. J. Schiff. Resolving time-series structure with a controlled wavelet transform. *Optical Engineering*, 31(11):2492–2495, Nov. 1992.

[28] T. Taralp, M. Devetsikiotis, and I. Lambadaris. Efficient Fractional Gaussian Noise Generation Using the Spatial Renewal Process. In *Proceedings IEEE International Communications Conference (ICC'98)*, pages S41.3.1–S41.3.5, Atlanta, GA, USA, Jun. 1998.

[29] M. Wickerhauser. *Adapted Wavelet Analysis from Theory to Software*. A K Peters, Ltd., Wellesley, Massachusetts, 1994.

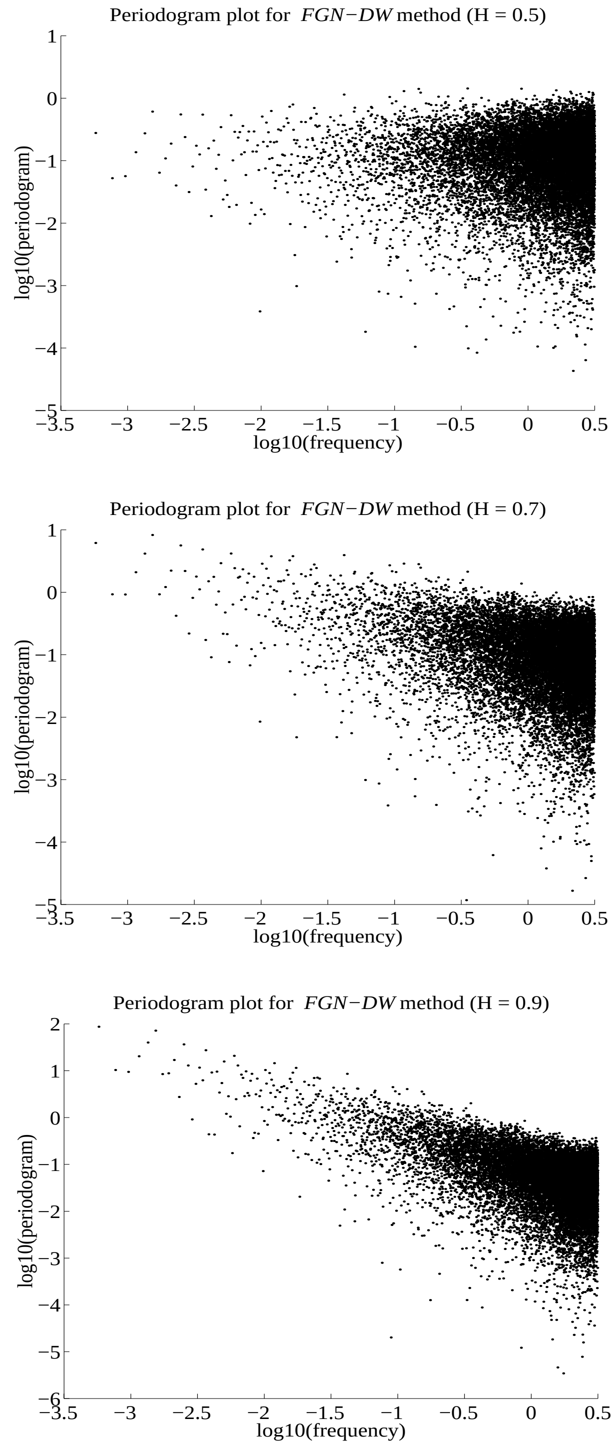

Figure 1. Periodogram plot for FGN-DW method (H = 0.5, 0.7, 0.9)

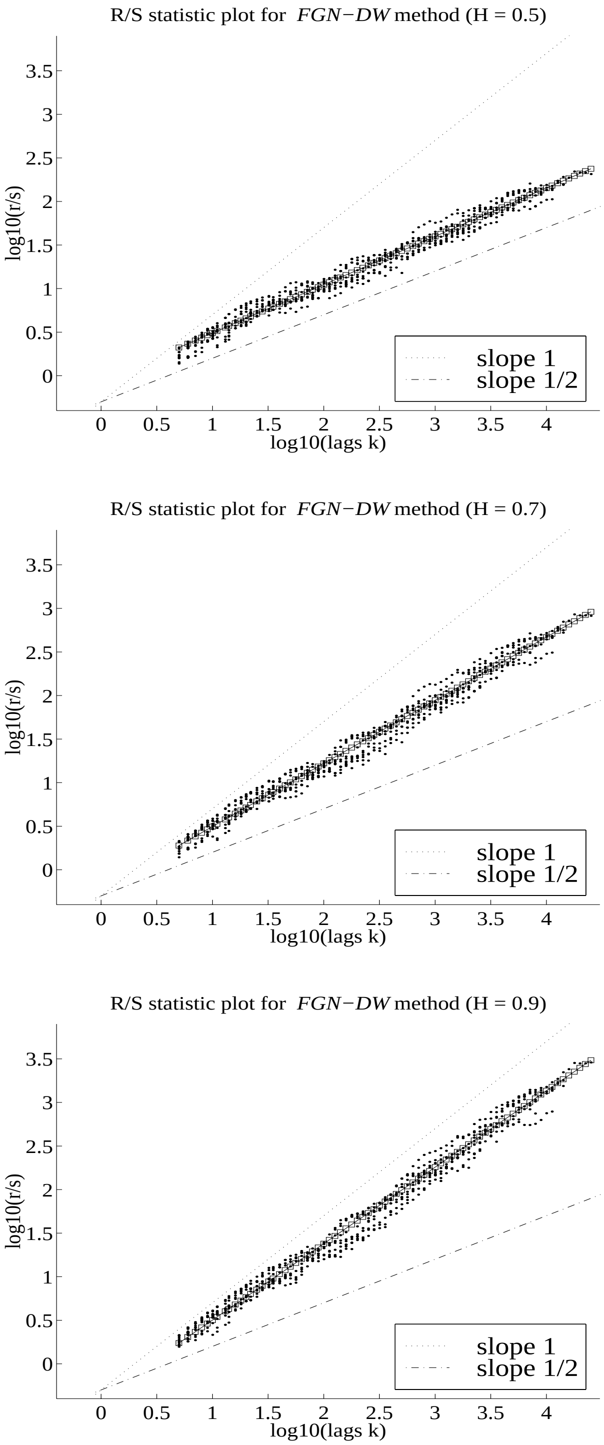

Figure 2. R/S statistic plot for FGN-DW method (H = 0.5, 0.7, 0.9)

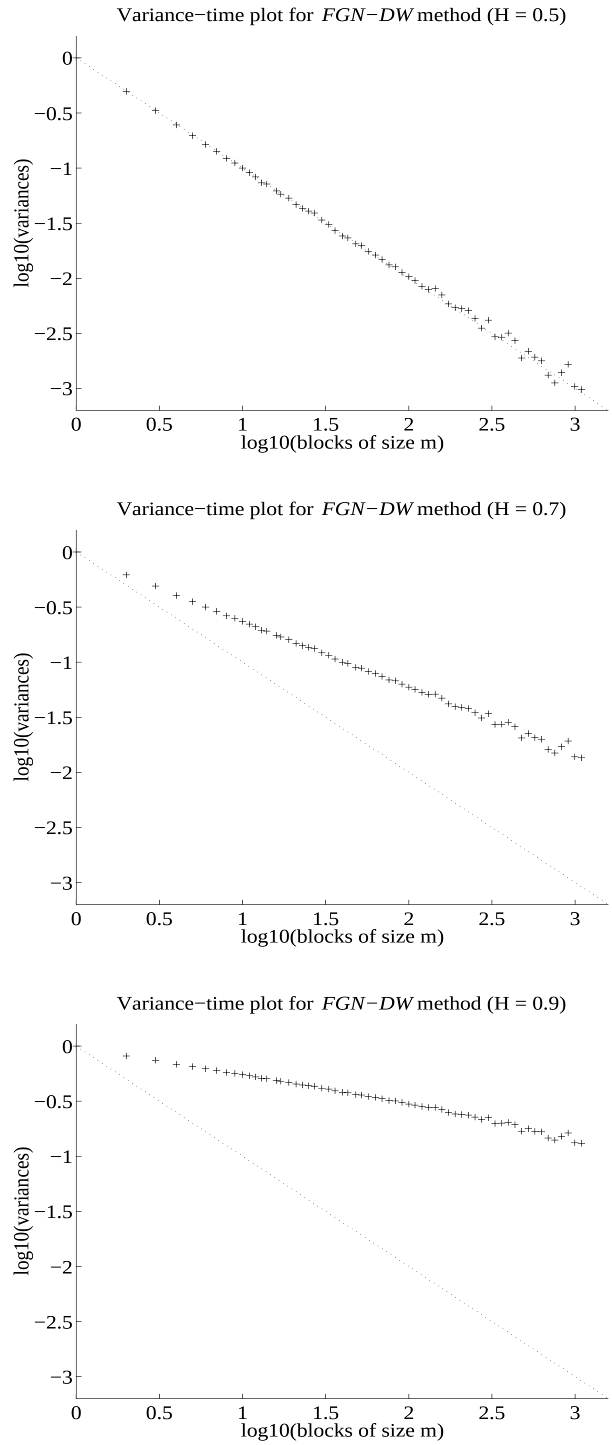

Figure 3. variance-time plot for FGN-DW method (H = 0.5, 0.7, 0.9)