A Search for Computationally Efficient Generators of Synthetic Self-Similar Teletraffic

H.-D. J. Jeong†, D. McNickle‡ and K. Pawlikowski†

†Department of Computer Science and ‡Department of Management
University of Canterbury
Christchurch, New Zealand
{joshua, krys@cosc, d.mcnickle@mang}.canterbury.ac.nz

Abstract. Recent traffic studies have shown that *self-similar* (or *fractal*) processes may provide better models for teletraffic than Poisson processes. If this is not taken into account, it can have serious effects on performance evaluation of computer networks. Thus, an important requirement for conducting simulation studies of telecommunication networks is the ability to generate synthetic stochastic self-similar sequences.
Three generators of pseudo-random self-similar sequences, based on the FFT [Paxson, 1995], RMD [Lau et al., 1995] and SRA method [Jeong et al., 1998], are compared in this paper. This study has indicated that (i) the SRA method is faster than two other methods when long sequences are generated; (ii) the SRA method produces more accurate self-similar sequences (in the sense of Hurst parameter) than the RMD method.

Keywords: teletraffic generators, complexity, self-similar processes, Hurst parameter

1 Introduction

The search for accurate mathematical models of data streams flowing in modern data communication networks has attracted a considerable amount of interest in the last few years. The reason is that several recent teletraffic studies of local area networks: broadband-integrated services digital networks and wide area networks [Likhanov et al., 1995], [Leland et al., 1994], [Paxson and Floyd, 1995], [Ryu, 1996]; the world wide web, especially when engaged in such sophisticated services as variable-bit-rate (VBR) video transmission [Garrett and Willinger, 1994], [Krunz and Makowski, 1997], [Rose, 1997], have shown that commonly used teletraffic models, based on Poisson or related processes, are not able to capture

---

Proceedings of the Twenty Second Australasian Computer Science Conference, Auckland, New Zealand, January 18–21 1999.

the self-similar (or fractal) nature of teletraffic in real networks. 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 [Beran, 1992], [Paxson and Floyd, 1995], insufficient allocation of communication and data processing resources, and difficulties in ensuring the quality of service expected by network users. 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 self-similar sequences have been proposed. They include methods based on fast fractional Gaussian noise [Mandelbrot, 1971], fractional ARIMA processes [Hosking, 1984], the $M/G/\infty$ queue model [Krunz and Makowski, 1997], [Leland et al., 1994] and autoregressive processes [Cario and Nelson, 1998], [Granger, 1980]. Most of them generate asymptotically self-similar sequences and require large amounts of CPU time. For example, Hosking's method [Hosking, 1984], based on the F-ARIMA$(0, d, 0)$ process, needs many hours to produce a self-similar sequence with 131,072 (2^{17}) numbers on a Sun SPARCstation 4. It requires $O(n^2)$ computations to generate n numbers. Even though exact methods of generation of self-similar sequences exist (for example: [Mandelbrot, 1971]), they are only fast enough for short sequences. In addition, they are often inappropriate when long sequences are generated because the whole sequences need to be generated in advance. To overcome this, an approximate method of generating self-similar sequences is required for simulation studies of telecommunication networks.

Our comparative evaluation of three methods proposed for generating self-similar sequences concentrates on two aspects: (i) how accurately self-similar processes can be generated, and (ii) how fast the methods generate long self-similar sequences. We consider three methods: (i) Paxson method [Paxson, 1995] based on the *fast Fourier transform (FFT)* algorithm, which we call FFT method; (ii) a method based on the *random midpoint displacement (RMD)* algorithm, implemented by Lau, Erramilli, Wang and Willinger [Lau et al., 1995] and (iii) a method based on the *successive random addition (SRA)* algorithm, implemented by us [Jeong et al., 1998] (adapted from Saupe, D. in Chapter 5 of [Crilly et al., 1991]).

Our results indicate that the SRA method is faster than two other methods when long sequences are generated. The SRA method produces sequences with more accurate values of the Hurst parameter than the RMD method. This suggests that the SRA method should be a serious candidate for generating self-similar teletraffic in performance evaluation studies of communication networks. For more detailed discussions, also see [Jeong et al., 1998].

In the next section basic definitions of self-similar processes and their properties, including *slowly decaying variance, long-range dependence* and *Hurst effect*, are described. A comparative analysis of three methods for generating synthetic self-similar sequences is presented in section 3. The efficiency of each method in the sense of its accuracy and complexity is studied in section 4. Section 5 summarises the paper.

2 Self-Similar Processes and Their Properties

Traditionally, teletraffic has been modeled by Poisson processes, but several recent traffic studies in real networks have shown that packet inter-arrivals and service demands do not follow exponential distributions. Poisson models fail to accurately capture real traffic behaviour where traffic bursts appear over a long range of time scales or when correlations persist over large time scales. As mentioned, it has been shown that better models of teletraffic are based on self-similar stochastic processes. These models are defined as follows:

A continuous-time stochastic process $\{X_t\}$ is *self-similar* with a self-similarity parameter $H(0 < H < 1)$, if for any positive stretching factor c, the rescaled process with time scale $ct, c^{-H}X_{ct}$, is equal in distribution to the original process $\{X_t\}$ [Beran, 1994]. This means that, for any sequence of time points $t_1, t_2, \ldots, t_n$, and for all $c > 0$, $\{c^{-H}X_{ct_1}, c^{-H}X_{ct_2}, \ldots, c^{-H}X_{ct_n}\}$ has the same distribution as $\{X_{t_1}, X_{t_2}, \ldots, X_{t_n}\}$.

Self-similar processes are characterised by a single key parameter called the *Hurst parameter H*. This parameter is designed to capture the degree of self-similarity in a given sequence of empirical data.

In more detail, let $\{X_k\} = \{X_k : k = 0, 1, 2, \ldots\}$ be a (discrete-time) stationary process with mean μ, variance σ^2, and autocorrelation function (ACF) $\rho(k)$, for $k = 0, 1, 2, \ldots$, and let $\{X_k^{(m)}\}_{k=1}^{\infty} = \{X_1^{(m)}, X_2^{(m)}, \ldots\}$, $m = 1, 2, 3, \ldots$, be a sequence of batch means, i.e., $X_k^{(m)} = (X_{km-m+1} + \ldots + X_{km})/m, k \geq 1$.

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

The process $\{X_k\}$ is *asymptotically self-similar* with $H = 1 - (\beta/2)$, if the ACF $\rho^{(m)}(k) \to \rho(k)$, as $m \to \infty$.

The most frequently studied self-similar traffic models belong either to the class of fractional autoregressive integrated moving-average (F-ARIMA) processes or to the class of fractional Gaussian noise processes; see [Hosking, 1984], [Leland et al., 1994], [Paxson, 1995]. F-ARIMA(p, d, q) processes were introduced by Hosking [Hosking, 1984] who showed that they are asymptotically self-similar with Hurst parameter $H = d + \frac{1}{2}$, as long as $0 < d < \frac{1}{2}$. In addition, the incremental process $\{Y_k\} = \{X_k - X_{k-1}\}, k \geq 0$, is called the *fractional Gaussian noise* (FGN) process, 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 ACF $\rho_k = \frac{1}{2}(|k+1|^{2H} - 2|k|^{2H} + |k-1|^{2H}), \quad k > 0$. A FBM process, which is the sum of FGN increments, is characterised by three properties [Mandelbrot and Wallis, 1969]:

(i) it is a continuous zero-mean Gaussian process $\{X_t\} = \{X_s : s \geq 0$ and $0 < H < 1\}$ with ACF $\rho_{s,t} = \frac{1}{2}(s^{2H} + t^{2H} - |s-t|^{2H})$ 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, the function $\{X_{ct}\}$ is changed by c^H.

Main properties of self-similar processes include ([Beran, 1994], [Cox, 1984], [Leland et al., 1994]):

- *Slowly decaying variance.* The variance of the sample mean decreases more slowly than the reciprocal of the sample size, that is, $Var[\{X_k^{(m)}\}] \rightarrow cm^{-\beta}$ as $m \rightarrow \infty$, where c is a constant and $0 < \beta < 1$.
- *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, that is, $\sum_{k=0}^{\infty} \rho(k) = \infty$. The speed of decay of autocorrelations is more like hyperbolic than exponential.
- *Hurst effect.* Self-similarity manifests itself by a straight line of slope β on a log-log plot of the R/S statistic. For a given set of numbers $\{X_1, X_2, \ldots, X_n\}$ with sample mean $\hat{\mu} = E\{X_i\}$ and sample variance $S^2(n) = E\{(X_i - \hat{\mu})^2\}$, Hurst parameter H is presented by the *rescaled adjusted range* $\frac{R(n)}{S(n)}$ (or R/S *statistic*) where $R(n) = \max\{\sum_{i=1}^{k}(X_i - \hat{\mu}), 1 \leq k \leq n\} - \min\{\sum_{i=1}^{k}(X_i - \hat{\mu}), 1 \leq k \leq n\}$ and S is estimated by $S(n) = \sqrt{E\{(X_i - \hat{\mu})^2\}}$. 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, i.e., $E[\frac{R(n)}{S(n)}] \rightarrow cn^H$ as $n \rightarrow \infty$ with $0.5 < H < 1$ and c is a finite positive constant.

3 Three Methods

Two methods, the FFT and RMD, were suggested as being sufficiently fast for practical applications in generation of simulation input data. In this paper, we have reported properties of these two methods (FFT and RMD) and compare them with SRA. These methods can be characterised as follows:

3.1 FFT method

This method generates approximate self-similar sequences based on the fast Fourier transform and a process known as the fractional Gaussian noise (FGN) process. Its main difficulty is connected with the power spectrum which is based on an infinite sum required. Paxson solves this by applying a special approximation.

Figure 1 shows how FFT method generates self-similar sequences. Briefly, these transformations (i) calculate the power spectrum using the periodogram (the power spectrum at a given frequency represents an independent exponential random variable); (ii) construct a sequence of complex values which are governed

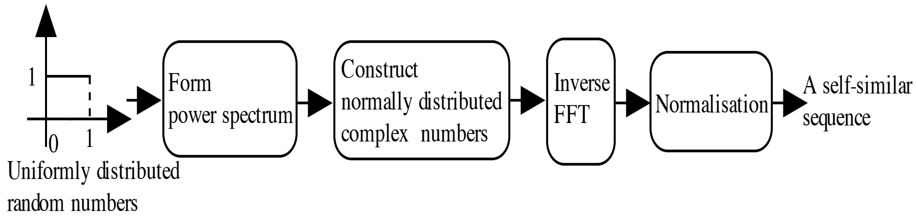

Fig. 1. FFT method

by normal distribution; (iii) apply the inverse FFT. For a more detailed reference, see [Paxson, 1995].

The FFT method has three input parameters; the Hurst parameter $H(0.5 < H < 1)$, the mean input rate M, and the peakedness factor A, defined as the ratio of variance to the mean.

3.2 RMD method

The basic concept of the *random midpoint displacement (RMD)* algorithm is to extend the sequence recursively, by adding new values at the midpoints from the values at the endpoints.

Figure 2 shows how the RMD algorithm works. Figure 3 illustrates the first three steps of the method, leading to generation of the sequence $(d_{3,1}, d_{3,2}, d_{3,3}, d_{3,4})$. The reason for subdividing the interval between 0 and 1 is to construct the Gaussian increments of X. Adding offsets to midpoints makes the marginal distribution of the final result normal. For more detailed discussions of the RMD method, see [Lau et al., 1995], [Peitgen et al., 1992].

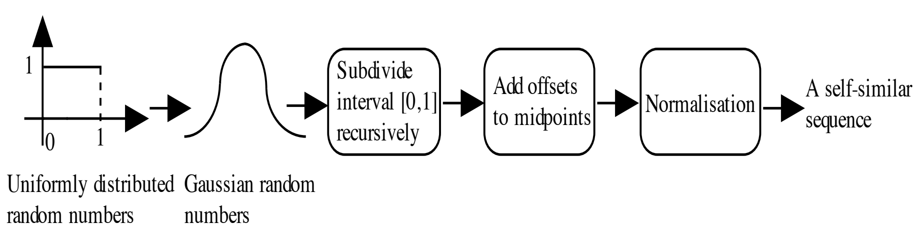

Fig. 2. RMD method

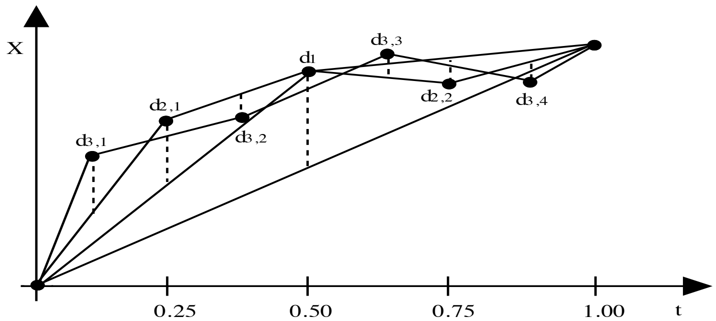

Fig. 3. The first three steps in the RMD method

Given a sequence of the approximate FBM process $\{X_t\}$ generated by the RMD method, we can transform it into the self-similar cumulative arrival process $\{Y_t\}$ [Lau et al., 1995], [Norros, 1994]: $\{Y_t\} = Mt + \sqrt{AM}\{X_t\}, \quad t \in (-\infty, +\infty)$ where M is the mean input rate and A is the peakedness factor, which is defined as the ratio of variance to the mean, $M > 0, A > 0$. The Gaussian incremental process $\{\tilde{Y}_t\}$ from time t to time $t+1$ is given as: $\{\tilde{Y}_t\} = M + \sqrt{AM}[\{X_{t+1}\} - \{X_t\}]$.

3.3 SRA method

Another alternative method for the direct generation of FBM process can be based on the *successive random addition* (SRA) algorithm [Crilly et al., 1991]. The SRA method uses the midpoints like RMD, but adds a displacement of a suitable variance to all of the points to increase stability of the generated sequence.

Figure 4 shows how the proposed SRA method generates an approximate self-similar sequence. The reason for interpolating midpoints is to construct the Gaussian increments of X, which are correlated. Adding offsets to all points makes the marginal distribution of the final result normal and produces a more precise self-similar sequence of an approximate FBM process than RMD.

The SRA method consists of the following steps:

Step.1 If the process $\{X_t\}$ is to be computed for times instances t between 0 and 1, then start out by setting $X_0 = 0$ and selecting X_1 as a pseudo-random number from a Gaussian distribution with mean 0 and variance $Var[X_1] = \sigma_0^2$. Then $Var[X_1 - X_0] = \sigma_0^2$, and for $0 \le t_1 \le t_2 \le 1$,

$$Var[X_{t_2} - X_{t_1}] = |t_2 - t_1|^{2H}\sigma_0^2. \tag{1}$$

Step.2 Next, $X_{\frac{1}{2}}$ is constructed by the interpolation of the midpoint, that is, $X_{\frac{1}{2}} = \frac{1}{2}(X_0 + X_1)$.

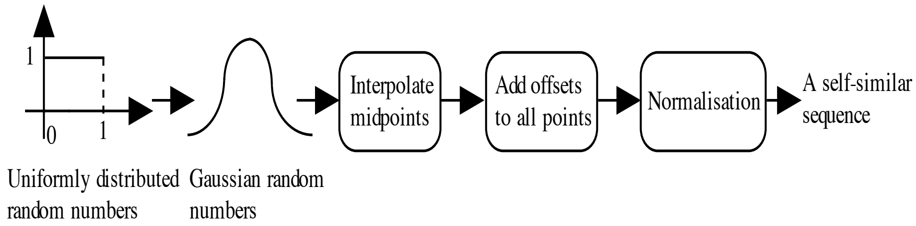

Fig. 4. SRA method

Step.3 Add a displacement of a suitable variance to all of the points, i.e., $X_0 = X_0 + d_{1,1}, X_{\frac{1}{2}} = X_{\frac{1}{2}} + d_{1,2}, X_1 = X_1 + d_{1,3}$. The offsets $d_{1,*}$ are governed by fractional Gaussian noise. For Equation (1) to be true, it is required that $Var[X_{\frac{1}{2}} - X_0] = \frac{1}{4}Var[X_1 - X_0] + 2S_1^2 = (\frac{1}{2})^{2H}\sigma_0^2$, that is, $S_1^2 = \frac{1}{2}(\frac{1}{2^1})^{2H}(1 - 2^{2H-2})\sigma_0^2$.

Step.4 Next, Step.2 and Step.3 are repeated. Therefore, $S_n^2 = \frac{1}{2}(\frac{1}{2^n})^{2H}(1-2^{2H-2})\sigma_0^2$ where σ_0^2 is an initial variance and $0 < H < 1$.

Step.5 The sequence of points $\{X_t\}$ is normalised to obtain the same sequence (cumulative arrival process) as that generated by FFT and RMD.

Using the above steps, the SRA method generates an approximate self-similar FBM process.

4 Analysis of Self-Similar Teletraffic Generators

Three generators of self-similar sequences of pseudo-random numbers described in the section 3 have been implemented on a Sun SPARCstation 4 (110 MHz, 32MB) using C. The mean times required for generating sequences of a given length were obtained by using the SunOS 5.5 `date` 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 efficiency of these methods in the sense of their accuracy and complexities, both from the theoretical and experimental point of view. For each of $H = 0.5, 0.55, 0.7, 0.9, 0.95$, each method was used to generate 100 sample sequences of 32,768 (2^{15}) numbers starting from different random seeds. Self-similarity of the generated sequences was assessed on the basis of R/S statistic. We have summarised the results of our analysis in the following:

4.1 R/S Statistic

The estimated Hurst parameter $\hat{H}$, obtained from the R/S statistic, has been used to compare the accuracy of three methods. Then, the mean inaccuracy

Table 1. Comparison of relative deviation of the mean inaccuracy (ΔH) from the required Hurst parameter value using R/S statistic plot.

H	FFT Method	RMD Method	SRA Method
0.5	+7.34 %	+8.74 %	+8.71 %
0.55	+5.32 %	+6.28 %	+6.23 %
0.7	+0.82 %	+1.28 %	+1.26 %
0.9	- 5.02 %	- 4.46 %	- 4.44 %
0.95	- 6.89 %	- 6.34 %	- 6.31 %

(ΔH) is calculated as the relative deviation from the required value by using the formula: $\Delta H = \frac{\hat{H}-H}{H} * 100\%$, where H is the input value and $\hat{H}$ is an empirical mean value.

The values of the asymptotic slope, designated by the Hurst parameter, of the R/S statistic plot in three methods are clearly between 1/2 and 1 (see Figure 5 - 8). A regression line showing the best least squares fit is also plotted, i.e., $\hat{H} = \hat{\beta}$ where $\hat{\beta}$ is the value of an asymptotic slope. The mean inaccuracy (ΔH) of the estimated Hurst parameter, obtained by R/S statistic plot, is given in Table 1. As we see, for $H = 0.5, 0.55, 0.7$, the FFT method is better than the other two, while for $H = 0.9, 0.95$, the SRA method is best. However, all three methods show that for $0.5 \leq H < 0.735$, the output Hurst parameters $\hat{H}$ are slightly larger than the required values, while for $0.735 < H < 1$, they are gradually smaller than the required values; see Table 1.

4.2 Computational Complexity

- *FFT method*
  The FFT method is the slowest of the three analysed methods for generating self-similar sequences. This is caused by relatively high complexity of the inverse FFT algorithm. Table 2 shows its time complexity and the mean time of generation. It took 5 seconds to generate a sequence of 32,768 (2^{15}) numbers, while generation of a sequence with 1,048,576 (2^{20}) numbers took 3 minutes and 47 seconds. FFT method requires $O(n\log n)$ computations to generate n numbers [Press et al., 1986].
- *RMD method*
  The RMD method is faster and simpler than FFT. Table 2 shows its time complexity and the mean time of generation. Generation of a sequence with 32,768 (2^{15}) numbers took 3 seconds. It also took 1 minute and 33 seconds to generate a sequence of 1,048,576 (2^{20}) numbers. The theoretical algorithmic complexity is $O(n)$ [Peitgen and Saupe, 1988].

Table 2. Comparison of complexity and time of generation. Time of generation was obtained by using the SunOS 5.5 `date` command on a Sun SPARCstation 4 (110 MHz, 32MB); each mean is averaged over 30 iterations.

Method	Complexity	Sequence of 32,768 Numbers	131,072 Numbers	262,144 Numbers	524,288 Numbers	1,048,576 Numbers
		Mean generation time (*minute:second*)				
FFT	$O(n\log n)$	0:5	0:20	0:35	1:12	3:47
RMD	$O(n)$	0:3	0:11	0:29	0:40	1:33
SRA	$O(n)$	0:3	0:10	0:20	0:40	1:31

– *SRA method*
  The SRA method appears to be the fastest of the three methods considered. Table 2 shows its time complexity and the mean time of generation. It only took 3 seconds to generate a sequence with 32,768 (2^{15}) numbers. Generation of a sequence with 1,048,576 (2^{20}) numbers took 1 minute and 31 seconds. The theoretical algorithmic complexity is $O(n)$ [Peitgen and Saupe, 1988].

Our results show that the RMD and SRA methods were more efficient in practical applications than FFT, when long self-similar sequences of numbers are needed.

5 Summary

This study has indicated that the SRA method is faster in the sense of computation time than the FFT and RMD methods proposed for generating self-similar sequences. The self-similar sequences generated by the SRA method also gave more precise values of the Hurst parameter than the RMD method.

Recently, many studies of self-similar traffic have tried to accurately and quickly generate self-similar sequences. However, all existing methods for generating synthetic self-similar sequences have drawbacks of either being computationally expensive, or generating approximate self-similar sequences with inaccurate values of Hurst parameter.

Our search for more efficient and accurate generator of self-similar sequences of pseudo-random numbers will be continued. We are also working on developing new tools for investigating self-similarity, including those based on periodogram and variance-time plots and maximum likelihood estimation (MLE).

References

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

Beran, J. (1994). *Statistics for Long-Memory Processes.* Chapman and Hall, An International Thomson Publishing Company.

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

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

Crilly, A.J., Earnshaw, R.A., and Jones, H. (1991). *Fractals and Chaos.* Springer-Verlag.

Garrett, M.W. and Willinger, W. (1994). 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.

Granger, C.W.J. (1980). Long Memory Relationships and the Aggregation of Dynamic Models. *Journal of Econometrics*, 14:227–238.

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

Jeong, H.-D.J., McNickle, D., and Pawlikowski, K. (1998). A Comparative Study of Generators of Synthetic Self-Similar Teletraffic. Technical Report TR-COSC 03/98, Department of Computer Science, University of Canterbury, Christchurch, New Zealand.

Krunz, M. and Makowski, A. (1997). A Source Model for VBR Video Traffic Based on $M/G/\infty$ Input Processes. Technical report, Department of Electrical and Computer Engineering, University of Arizona, Tucson.

Lau, W-C., Erramilli, A., Wang, J.L., and Willinger, W. (1995). Self-Similar Traffic Generation: the Random Midpoint Displacement Algorithm and Its Properties. In *Proceedings of IEEE ICC'95*, pages 466–472, Seattle, WA.

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

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

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

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

Norros, I. (1994). A Storage Model with Self-Similar Input. *Queueing Systems*, 16:387–396.

Paxson, V. (1995). Fast Approximation of Self-Similar Network Traffic. Technical Report LBL-36750, Lawrence Berkeley Laboratory and EECS Division,University of California, Berkeley.

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

Peitgen, H.-O., Jurgens, H., and Saupe, D. (1992). *Chaos and Fractals: New Frontiers of Science*. Springer-Verlag.

Peitgen, H.-O. and Saupe, D. (1988). *The Science of Fractal Images*. Springer-Verlag.

Press, W.H., Flannery, B.P., Teukolsky, S.A., and Vetterling, W.T. (1986). *Numerical Recipes: The Art of Scientific Computing*. Cambridge University Press.

Rose, O. (1997). *Traffic Modeling of Variable Bit Rate MPEG Video and Its Impacts on ATM Networks*. PhD thesis, Bayerische Julus-Maximilians-Universitat Wurzburg.

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

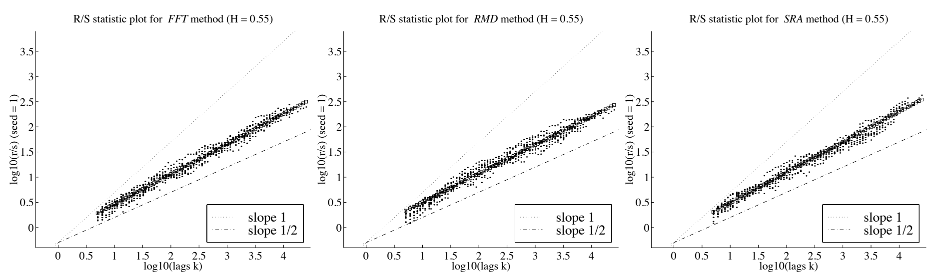

Fig. 5. R/S statistic plots for FFT, RMD and SRA method (H = 0.55).

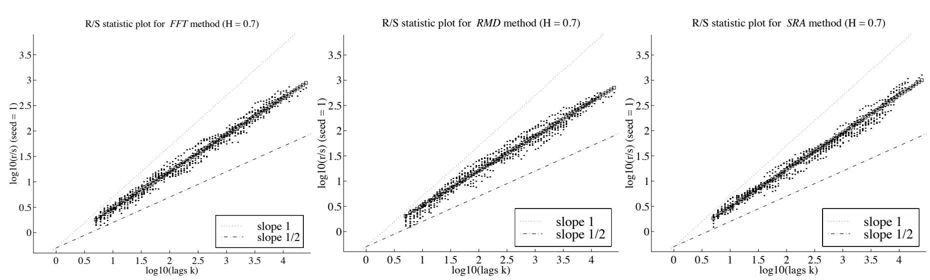

Fig. 6. R/S statistic plots for FFT, RMD and SRA method (H = 0.7).

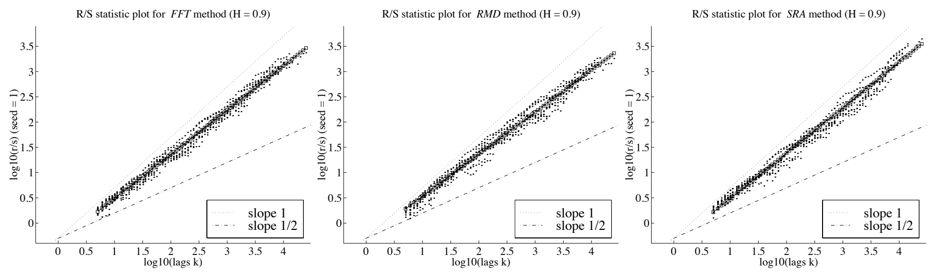

Fig. 7. R/S statistic plots for FFT, RMD and SRA method (H = 0.9).

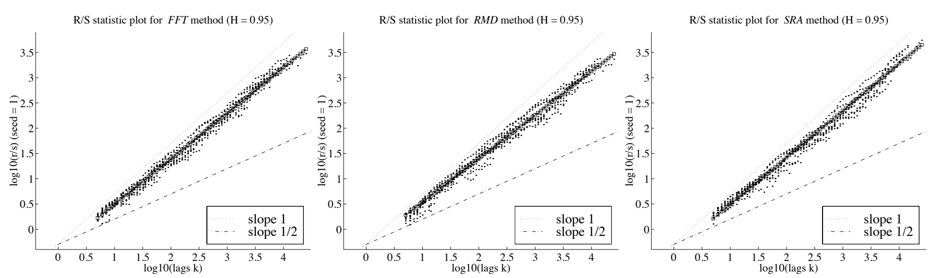

Fig. 8. R/S statistic plots for FFT, RMD and SRA method (H = 0.95).