Generation of Simulated Fractal and Multifractal Traffic

Marco A. Alzate

Universidad Distrital FJC

Abstract

We create software tools for the generation of sample traces of self-similar traffic as fractional Brownian motion, using Mathcad. These methods have been widely used in simulation studies, but they present the difficulty of generating negative samples. So we also introduce a very efficient technique for the generation of multifractal traffic by means of wavelet synthesis, which posses the property of positiveness.

1. Introduction

Traffic modeling is an important aspect to consider when we want to optimize the use of communication resources while guaranteeing a given quality of service.

In the early years of telecommunication technology, it was enough to characterize the number of calls as a Poisson process and their duration as exponential variables. Later on, data communications introduced greater complexity due to waiting times and medium access schemes. But, with the deployment of broadband networks that carry multimedia traffic, now we need to characterize not only the number and duration of the calls but also the bandwidth variability of the information flow during the call.

Several models have been developed to this purpose, which try to capture the bursty nature of multimedia traffic. The principal modeling technique has been the multiplexing of on-off sources, which lead to Markov modulated processes.

However, recently there has been empirical evidence that traffic in Ethernet, Internet, MPEG Video, etc. is characterized by a significant correlation over a broad range of time scales. This large-scale correlation can lead to a long-range dependent behavior, i.e. one in which the autocorrelation function is not summable. The simplest long-range dependent processes are the self-similar processes, characterized by a hyperbolically decaying autocorrelation function.

There is a significant impact of long-range dependent traffic on queueing behavior, which is not predicted by classical traffic models. However, we still need much further research (including fundamental mathematical research) to develop analytical models that can predict these effects. In the meanwhile, simulation methods provide any degree of accuracy (although at very high computational costs) for both, performance analysis and analytical model validation.

In the first part of this report we review some basic concepts of fractals and create some software tools for the generation of sample traces of self-similar traffic. These traces are of the fractional Brownian motion type, which has been the most broadly applied fractal signal model because of its power and simplicity: it is statistically self similar and, still, it is subject to tractable analysis.

In the second part we introduce the multifractal wavelet model. Wavelet analysis is an adequate technique for studying fractional Brownian motion because the wavelet coefficients become independent zero-mean Gaussian random variables with power-law decaying variance. Unfortunately these fractional Brownian motion has a significant limitation for modeling actual processes which are inherently positive, like network traffic. The multifractal wavelet model overcomes this difficulty in a very promisory way.

2. Fractals, self-similarity and fractional dimension

Diverse scientific disciplines have adopted the language of fractal geometry, including the modeling of multimedia traffic in communication networks.
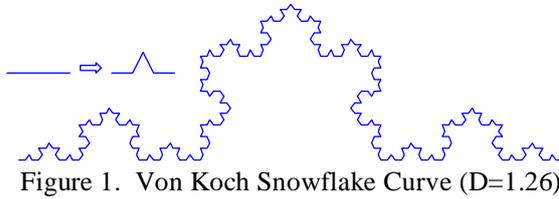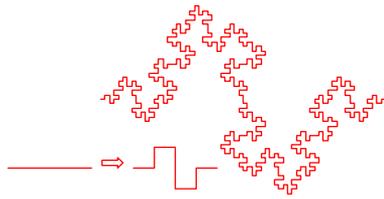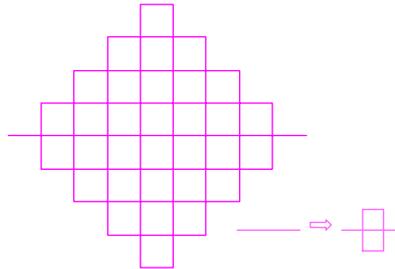

Figure 1. Von Koch Snowflake Curve (D=1.26)

A fractal is a geometric shape that exhibits invariance under changes of magnification. A good example is the Von Koch snowflake curve (Figure 1), which is constructed by dividing a simple line segment into thirds and replacing the middle segment by two equal segments as in an equilateral triangle. The procedure is repeated for each new segment. Notice that, if we magnify a small portion of the Von Koch curve, we reproduce exactly a larger portion. This property is called exact self-similarity.

Figure 2. Variation with D=1.5

Figure 3. Variation with D=2

Accordingly, a line is a one-dimensional exact self-similar shape, which can be divided in to N identical lines, each of which is scaled down by the ratio $r = N^{-1}$. A square is a two-dimensional exact self-similar shape, which can be divided in to N identical squares, each of which has a size scaled down by the ratio $r = N^{-1/2}$. A cube is a three-dimensional exact self-similar shape, which can be divided in to N identical cubes, each of which has a size scaled down by the ratio $r = N^{-1/3}$. Generalizing, a D-dimensional self-similar object can be divided into N smaller "copies" of itself, each of which is scaled by a factor $r = N^{-1/D}$. Then, given a self-similar object of N parts scaled by a ratio r from the whole, its fractal or similarity dimension is given by $D = \log(N) / \log(1/r)$

In the Von Koch curve, each segment is composed of four sub-segments, each of which is scaled by a factor of 1/3, so $D = \log(4) / \log(3) = 1.26$. So this curve fills the space more than a single line (D=1) but less than an Euclidean area of the plane (D=2). In the variation of the Von Koch algorithm shown in figure 2, each of the segments is replaced by 8 new segments, each ¼ of the original length, so the fractal dimension is $\log(8)/\log(4) = 1.5$. As a final example, the variation of figure 3 has fractal dimension $\log(9)/\log(3) = 2$, since each of the segments is replaced by 9 new segments, each 1/3 of the original length.

2.1 1/f-noise

In our traffic model project we are not interested in exact self-similarity but in statistical self-similarity. This happens when a large-scale view is insufficient to predict the exact details of a magnified view; i.e. magnified segments look like, but never exactly like, segments at different scales. In this case, the corresponding figures could be seen as noise patterns due to their unpredictability. The traces made by such noises are fractal curves where the fractal dimension is directly related to the logarithmic slope of the spectral density.

In a brownian motion the spectral density varies as $1/f^2$. In a white noise, the spectral density is a flat line (varies as $1/f^0$). But in many physical systems, the noise has an spectral density that varies as $1/f^\beta$, $0.5 < \beta < 1.5$; those noises are called, in general, 1/f-noises. So we should start this project with the generation of this kind of noises.

2.2 Fractional Brownian Motion

A fractional brownian motion $V_H(t)$ is characterized by a parameter $0 < H < 1$, which relates the typical change in V, $\Delta V = V_H(t_2) - V_H(t_1)$, to the time difference $\Delta t = t_2 - t_1$ by the simple scaling law $\Delta V \propto \Delta t^H$. More precisely, the increments $V_H(t_2) - V_H(t_1)$ have a Gaussian distribution with variance $<|V_H(t_2) - V_H(t_1)|^2> = |t_2 - t_1|^{2H}$.

We say that the increments of V(t) are statistically self-similar in the sense that V(to+t)-V(to) has the same joint distribution function of $r^{-H}(V(t_o + rt) - V(t_o))$ for any t_o and r>0. Choosing t_o=0 and V(0)=0, this means that V(t) and $r^{-H}V(rt)$ are statistically indistinguishable.

The particualr case H=1/2 corresponds to the well known Brownian motion. A simple way to generate a Brownian motion is by integrating a white noise, as the Mathcad program of figure 4 shows.
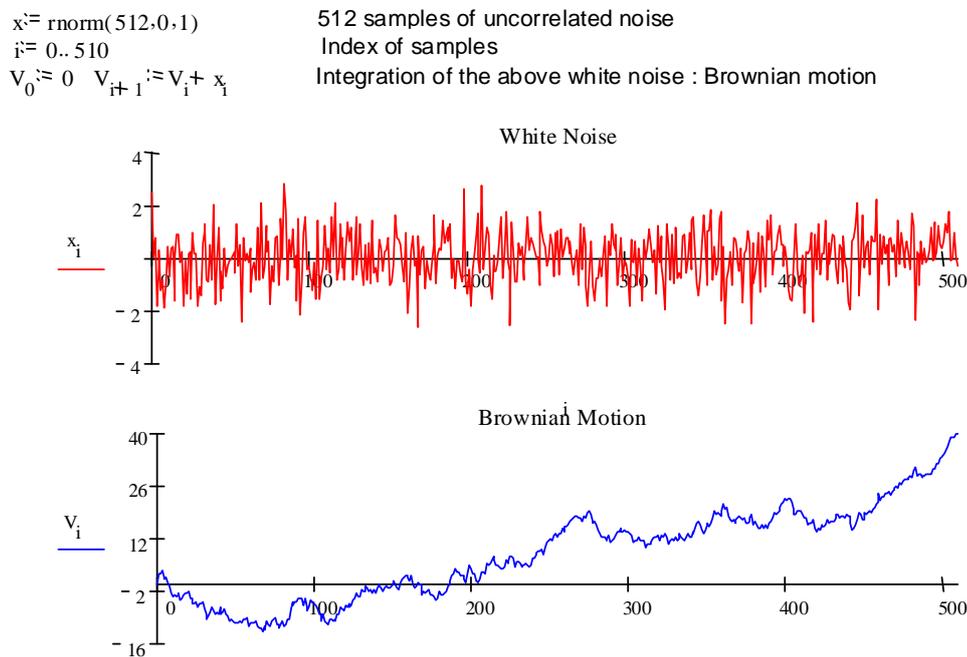
$x := rnorm(512,0,1)$ 512 samples of uncorrelated noise
$i := 0.. 510$ Index of samples
$V_0 := 0 \quad V_{i+1} := V_i + x_i$ Integration of the above white noise : Brownian motion

White Noise

x_i

Brownian Motion

V_i

Figure 4. Mathcad program for generating Brownian motion via white noise integration

From the above it is clear that E[Vi]=0 and $Cov[V_i,V_j] = |i-j|$, this is, $V(t_2)-V(t_1)$ is normal with mean 0 and variance $\sigma^2|t_2 - t_1|$. Accordingly, we could also select V(1) as a sample of a $N(0,\sigma^2)$ random variable (having V(0)=0). For $0 < t_1 < t_2 < 1$ we expect the variance of $V(t_2)-V(t_1)$ to be $\sigma^2(t_2 - t_1)$ so we can set V(0.5) such that V(0.5)-V(0) = 0.5[V(1)-V(0)] + x_1, where x_1 is a sample of a $N(0,0.25\sigma^2)$ random variable, so the total variance of V(0.5) – V(0) will be $0.5\sigma^2$ as expected. The same holds for V(1)-V(0.5).

In the next step we set V(0.25) – V(0) = 0.5[V(0.5) – V(0)] + x_2, where x_2 is a sample of a $N(0,0.125\sigma^2)$ random variable, so the total variance of V(0.25)-V(0) will be $0.25\sigma^2$ as expected. The same holds for V(1)-V(0.75). We can continue in this way with x_n distributes as $N(0,\sigma^2 2^{-(n+1)})$ as in the recursive algorithm of figure 5, called "midpoint displacement".

$$mxL := \frac{\ln(N)}{\ln(2)} \qquad L := 1..\,mxL \qquad\qquad \text{Number of levels, index of levels}$$

$$\sigma_0 := 1 \qquad\qquad \sigma_L := 2^{-(0.5 \cdot (L+1))} \qquad\qquad \text{Array of standard deviations}$$

$$
\begin{aligned}
\text{MidPoint}(X, i0, i2, L) := \Big|\ & i1 \leftarrow 0.5 \cdot (i0 + i2) && \text{Midpoint} \\
& X_{i1} \leftarrow 0.5 \cdot \left(X_{i0} + X_{i2}\right) + \sigma_L \cdot wn_{i1-1} && \text{Displacement} \\
& \text{if } L < mxL && \text{Recursive computation:} \\
& \quad \Big|\ X \leftarrow \text{MidPoint}(X, i0, i1, L+1) && \qquad \text{Previous midpoint} \\
& \quad \Big|\ X \leftarrow \text{MidPoint}(X, i1, i2, L+1) && \qquad \text{Next midpoint} \\
& X && \text{Update the array}
\end{aligned}
$$

$$bm_0 := 0 \qquad bm_N := wn_{N-1} \qquad\qquad \text{Initialize the endpoints}$$

$$bm := \text{MidPoint}(bm, 0, N, 1) \qquad\qquad \text{Actual recursion}$$
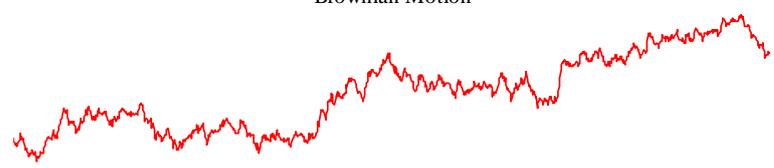
Brownian Motion

Figure 5. Mathcad program for generating Brownian motion via Midpoint displacement

Finally, we can interpret the Brownian motion as the cumulative displacement of infinite independent jumps, $V(t) = \Sigma_{n=-\infty}^{\infty} S_n U(t-t_n)$, where S_n are independent Gaussian random variables, t_n are independent Poisson random variables and $U(t)=1$ for $t>0$ and $U(t)=0$ elsewhere. This interpretation is useful for us in the sense that the traffic intensity could vary by an amount S_n at instant t_n (an scene change in a movie, an activation of deactivation of an user in a big multiplexer, etc.). Figure 6 shows this new algorithm.

$$
\begin{aligned}
\text{BM}(NS) := \Big|\ & X_{N-1} \leftarrow 0 && \text{Initialize the array} \\
& \text{for } s \in 0..\,NS-1 && \text{Number of steps} \\
& \quad \Big|\ k0 \leftarrow \text{floor}(N \cdot rnd(1)) && \text{Ends of the cutted segment,} \\
& \quad \Big|\ k1 \leftarrow k0 + \frac{N}{2} - 1 && \text{The segment has length N/2} \\
& \quad \Big|\ \text{for } k \in k0..\,k1 \\
& \quad \quad \Big|\ X_k \leftarrow X_k + wn_s \ \text{ if } k < N && \text{Displace this segment} \\
& \quad \quad \Big|\ X_{k-N} \leftarrow X_{k-N} + wn_s \ \text{ otherwise} \\
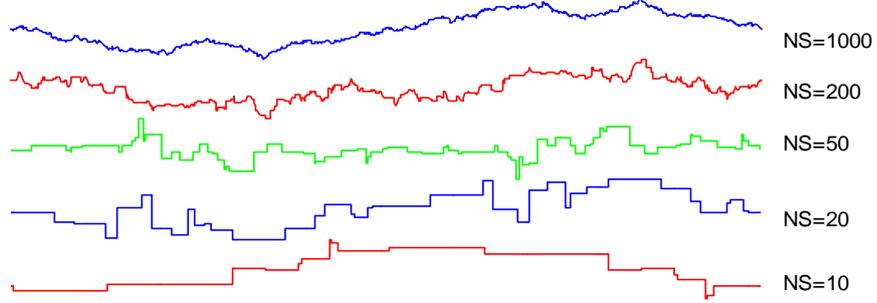& X
\end{aligned}
$$

Brownian Motion

Figure 6. Mathcad program for generating Brownian motion via independent cuts

The generalization $\text{Var}[V(t_2) - V(t_1)] = \sigma^2 |t_2 - t_1|^{2H}$ for 0<H<1 corresponds to a Fractional Brownian Motion (in previous simple Brownian motion, H=0.5). Now self-similarity means that $V(t_0 + t) - V(t_0)$ and $r^{-H}[V(t_0 + rt) - V(t_0)]$ have the same distribution. In the midpoint displacement algorithm, we simply redefine the sequence of variances σ_L to reflect its dependence on H. Figure 7 shows the effect of H –compare the case H=0.5 with figures 4, 5 and 6-.
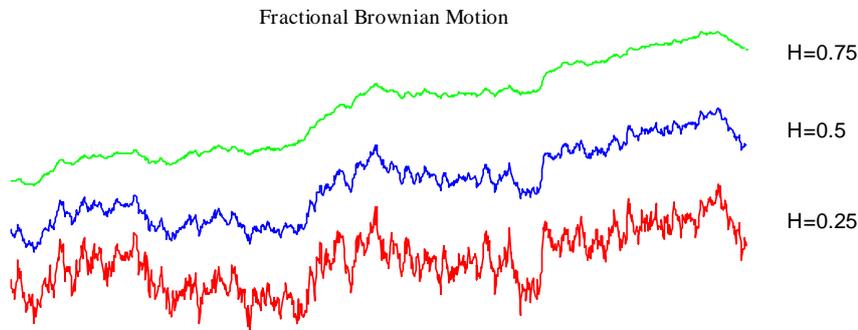

Figure 7. Fractional Brownina motion by Midpoint displacement

Whereas self-similar shapes repeat (statistically or exactly) under a magnification, fractional brownian motion repeats statistically only when t and V are magnified by different ammount : if t becomes r·t, V must become $r^H·V$ (for a random walk (H=1/2) we must take four times as many steps to go twice as far). This scaling property is known as self-affinity.

Fractals, as euclidean shapes, reduce their dimension by one under intersection with a plane. 3D sphere $\Rightarrow$ 2D circle $\Rightarrow$ 1D line $\Rightarrow$ 0D point. Similarly, the intersection of a fractal curve with fractal dimension 1 < D < 2 and a straight line is a fractal set of points of fractal dimension D-1. If the straight line eliminates one of the coordinates, a self-affine curve can be reduced to a self-similar set of points.

The zeroset of a fractional brownian motion is the intersection of $V_H(t)$ with the t-axis (all points such that $V_H(t)=0$). It's a disconnected set of points with topological dimension 0 and a fractal dimension $0 < D_0 = 1 - H < 1$, $V_H(t)$ is self-affine, but its zeroset is selfsimilar. Consequently, $D = D_0 + 1 = 2 - H$

The trace of a fractional brownian motion with H=0.8 resembles a mountainous horizon. Replace t by (x,y) to obtain $V_H(x,y)$ as the surface altitude at position (x,y). Analogous to $V_H(t)$, if a hiker travels a distance $\Delta r = \sqrt{(\Delta x^2 + \Delta y^2)}$, the typical altitud variation ΔV is proportional to Δr^H. The fractal dimension D must be greater than the topological dimension 2 of the surface : D = 3 – H. Once again, the intersection of a plane with the surface $V_H(x,y)$ is a self affine fractional brownian motion with fractal dimension $D_0 = D – 1 = 2 – H$. This generalization can go to higher dimensions : An observer moving at a constant speed along a straight line on $V_H(x,y,z)$ generate a fractional brownian motion with $\Delta V \propto \Delta r^H$, where $\Delta r = \sqrt{(\Delta x^2 + \Delta y^2 + \Delta z^2)}$. In this case, the fractal dimension is D = 4 – H. The zeroset $V_H(x,y,z)$ = constant gives a self-similar fractal with $D_0 = 3 – H$.

A statistically self-affine fractional brownian function V_H provides a good model for many natural scaling processes and shapes : $V_H(t)$ for noises and music, $V_H(x,y)$ for landscapes and surfaces, $V_H(x,y,z)$ for flakes and clouds, etc. In all cases, the scaling property may be characterized by H, D or even the "spectral exponent" β : $V(t) \Leftrightarrow S_V(f) \sim |V(f)|^2/\Delta f \propto 1/f^\beta$. (As we said before, this is why they are called 1/f-noises).

Variations on H are related with variations on β very simply : For a fractional brownian motion function of E variables (t, (x,y), (x,y,z),...) , D = E + 1 – H = E + (3 - β) / 2.

For $0 < H < 1$ we have $E < D < E+1$ and $1 < \beta < 3$. $H \sim 0.8$ is empirically a good choice for many natural phenomena, including fractal traffic.

Figure 7 showed the generation of a fractional Brownian motion by the midpoint algorithm. We can also use the independent cut scheme of figure 6 since, changing the step function, we can generate fractals with different H and D. Thus $U(t) = t^{H-1/2}$ for $t \geq 0$ and $U(t) = -|t|^{H-1/2}$ for $t < 0$ can be used to generate a fractional brownian motion with H different from ½ as $V(t) = \Sigma_{n=-\infty}^{\infty} S_n U(t-t_n)$. The following figures were generated this way, with $H = 0.8$:
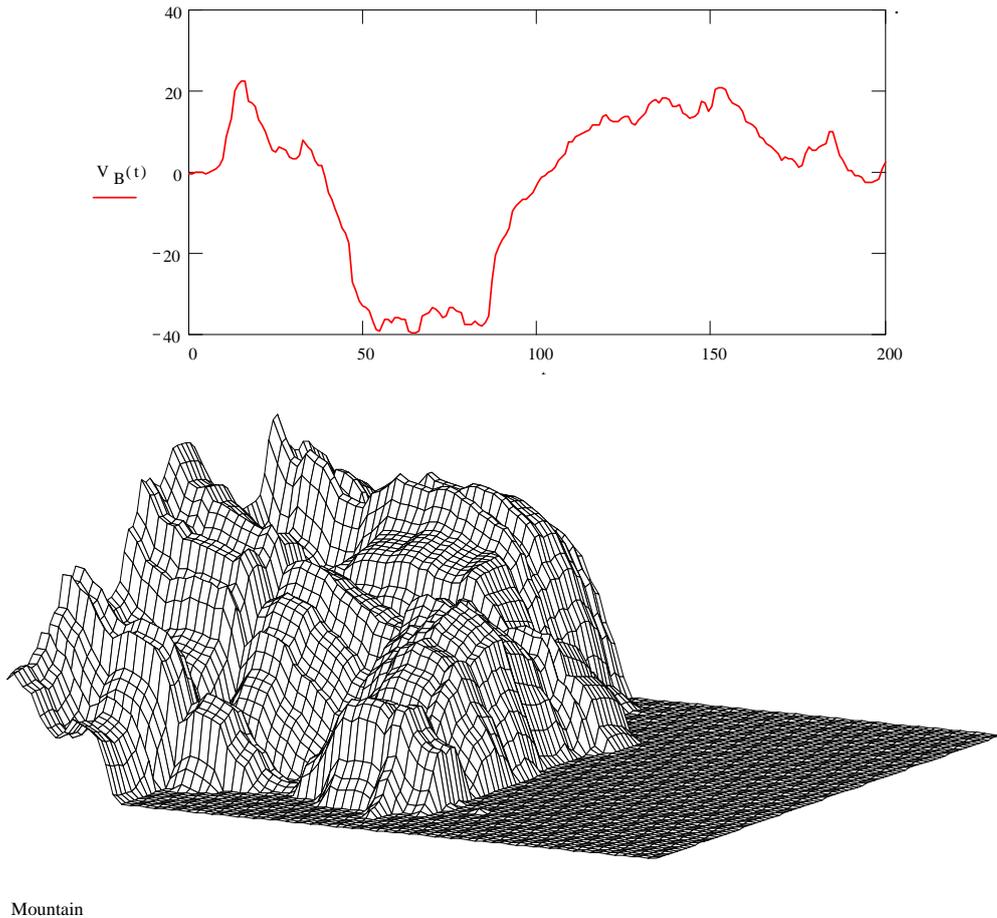

Mountain

Figure 8. Fractional Brownian motions as functions of one and two variables

However, this method is computationally expensive. A typical approach is, then, by FFT filtering. A white noise $W(t)$ has a "constant" spectral density $S_W(f)$. If we filter it with a transfer function $T(f)$, the output $V(t)$ has a spectral density $S_V(f) = |T(f)|^2 S_W(f) \propto |T(f)|^2$. Selecting $T(f) \propto 1/f^{\beta/2}$ we obtain an $1/f^\beta$ -noise. The following canyon is a 2-dimensional white noise filtered by $T_{n,k} = 1/(n^2 + k^2)$, created with 10, 25 and 100 frequency components:
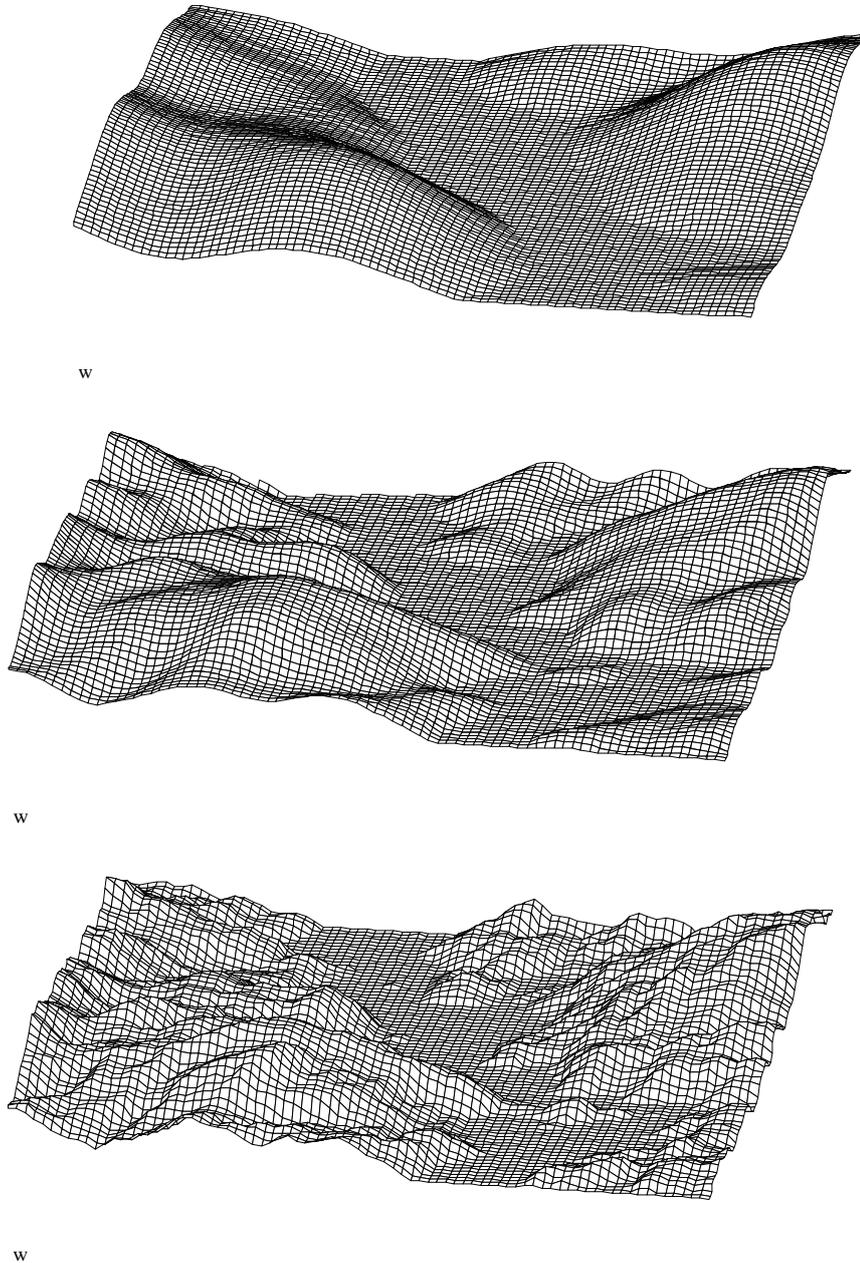
w

w

w

Figure 9. A fractal canyon created by filtered white noise

2.3 Wavelet synthesis of Fractional Brownian Motion

The wavelet transform is an analysis technique very well suited for studying multiscale phenomena, just like the signals we are considering in this report.

The wavelet transform represents a 1-dimensional signal X(t) in terms of shifted and dilated versions of a prototype bandpass wavelet function $\psi(t)$ and a lowpass scaling function $\phi(t)$. For special choices of the wavelet and scaling functions, the shifted and delayed versions $\psi_{jk}(t) = 2^{j/2}\psi(2^j t-k)$ and $\phi_{jk}(t) = 2^{j/2}\phi(2^j t-k)$ form an orthonormal basis for the real functions, with the following signal representation:

$$X(t) := \sum_k U_{j0,k} \cdot \phi_{j0,k}(t) + \sum_j \sum_k W_{j,k} \cdot \psi_{j,k}(t) \qquad \text{with} \qquad W_{j,k} := \int X(t)\,\psi_{j,k}(t)\,dt \qquad \text{and} \qquad U_{j,k} := \int X(t)\,\phi_{j,k}(t)\,dt$$

For a wavelet $\psi(t)$ centered at time zero and frequency f_0, the wavelet coefficient $W_{j,k}$ gives the amplitude of the signal at time $2^{-j}k$ and frequency $2^j f_0$, while the scaling coefficient $U_{j,k}$ gives the local average at that time. So j is an index of scale.

The Haar scaling and wavelet functions (Figure 10), is the simplest example of an orthonormal wavelet basis.
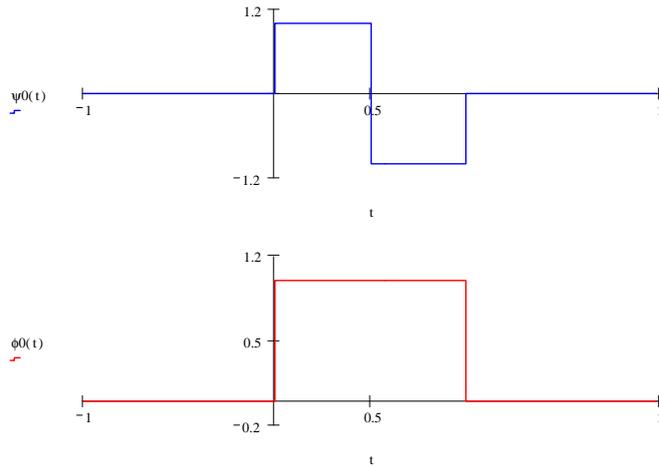

Figure 10. Haar wavelet and scaling function prototypes

Note the supports of fine-scale wavelet and scaling functions nest inside those at coarser scale. This property will be adequately exploited in next section.

The wavelet transform is an approximation of the Karhunen-Loéve expansion for fractional Gaussian noise. In effect, the wavelet coefficients become independent zero-mean Gaussian random variables with power-law decaying variance, $\mathrm{var}(W_{j,k}) \propto 2^{-jr}$, where r is 2H-1 for fractional Gaussian noise. This suggest the procedure of figure 11 for generating traces of fractional Brownian motion.

$$\psi 0(t) := (t \geq 0) - 2 \cdot (t \geq 0.5) + (t \geq 1) \qquad \psi(j,k,t) := 2^{0.5 \cdot j} \cdot \psi 0\left(2^j \cdot t - k\right) \qquad \text{Haar Wavelet Function}$$

$$X(t) := \sum_{j=0}^{5} \sum_{k=0}^{2^j - 1} \mathrm{rnorm}\left(1, 0, 2^{\frac{-j}{3}}\right)_0 \cdot \psi(j,k,t) \qquad \text{Wavelet Synthesis}$$

$$i := 0 .. 255 \qquad t_i := \frac{i}{256} \qquad Y_i := X\left(t_i\right) \qquad \text{Fractional Gaussian Noise}$$

$$Z_0 := 0 \qquad Z_{i+1} := Z_i + Y_i \qquad \text{Fractional Brownian Motion}$$
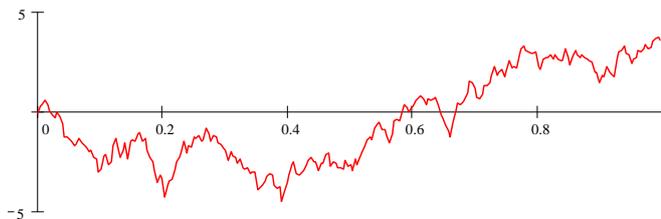

Figure 11. Fractional Brownian Motion by Wavelet Synthesis

3. The Multifractal Wavelet Model

Despite their great simplicity, fractal models based on fractional Brownian motion and fractional Gaussian noise have a significant limitation: Since they are Gaussian, there will be always the possibility of generating negative samples. Of course, traffic data are inherently positive and, besides, its distribution is spiky, far away from Gaussian smoothness. Furthermore, many real traffic traces exhibit Long Range Dependence but their short-term correlation and scaling behavior do not agree with the strict self similarity of these models.

Fortunately, for the Haar wavelet there is a simple condition the ensure the positiveness of the synthesized signal: $|W_{j,k}| \leq U_{j,k}$ for every j,k. In the multifractal wavelet model this characteristic is used by modeling the wavelet coefficients as $W_{j,k} = A_{j,k}U_{j,k}$, where the multipliers $A_{j,k}$ are independent random variables taking values between -1 and 1.

A multifractal is a fractal random process for which the parameter H depends on time on an erratic way, so the scaling behavior of the moments, as the signals are aggregated, is a nonlinear function of the moment order (like several fractals interwoven together: multi-fractals).

The multifractal wavelet model can capture closely the power spectrum (and consequently the long range dependence) of a set of training data, by adjusting the variances of the multipliers. And, unlike fractional Gaussian noise, it can also match positiveness and higher order statistics.

Taking advantage of the Haar wavelet properties, the algorithm for multifractal traffic generation is as follows:

$$\text{data}(N) := \left\| \begin{array}{l} U_{0,0} \leftarrow 1 \\[4pt] \text{for } j \in 0.. N - 1 \\[4pt] \quad \text{for } k \in 0.. 2^j - 1 \\[4pt] \qquad \left| \begin{array}{l} A_{j,k} \leftarrow 1.2 \cdot (\text{rnd}(1) - 0.5) \\[4pt] W_{j,k} \leftarrow a \cdot U_{j,k} \\[4pt] U_{j+1,2\cdot k} \leftarrow 2^{-0.5} \cdot \left(U_{j,k} + W_{j,k} \right) \\[4pt] U_{j+1,2\cdot k+1} \leftarrow 2^{-0.5} \cdot \left(U_{j,k} - W_{j,k} \right) \end{array} \right. \\[4pt] \left(U^T \right)^{<N>} \end{array} \right.$$

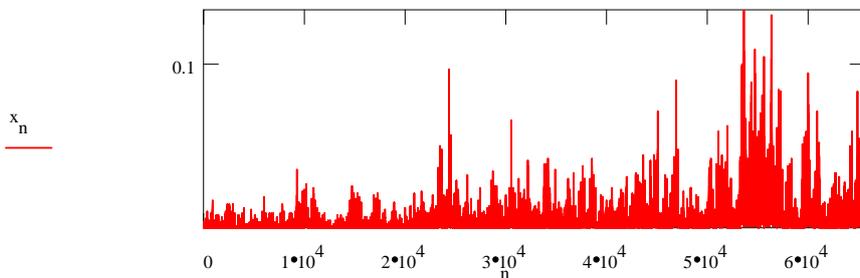$$x := \text{data}(16) \qquad\qquad n := 0.. \text{rows}(x) - 1$$

Figure 12. Multiplicative cascade for generating multifractal traces (notice we really do not need to store the multipliers, nor the wavelet coefficients. Further, we only need two consecutive scaling coefficient.)

In figure 12 we used $U_{0,0}=1$ and an uniform distribution between -0.6 and 0.6 for the multipliers, independently of the scale. Of course, the model should be completed by specifying probability density functions for those random variables. The correlations and long range dependence of the output signal are controlled through the wavelet energy decay, while higher order moments and marginal pdf are controlled through the scaling coefficient moments.

Actually, the high quality of the matching among real and simulated traffic traces suggests that some of the mechanisms shaping the traffic flow might have an inherent multiplicative structure. Normally, self-similar additive schemes model traffic arrivals as a mean rate with superimposed fractional Gaussian noise fluctuations, which agrees with the conception of traffic as the superposition of individual components and is accurate on large time scales. The multifractal wavelet model represent traffic arrivals as the product of random multipliers, which looks like the partition of total traffic throughput into parts and becomes appealing when considering small time scales.

4. Conclusions

Fractional Brownian motion and fractional Gaussian noise have been useful models in the study of the long-range dependence phenomena in network traffic, especially because of its tractability.

On the other hand, the multiplicative wavelet model combines the power of multifractals with the efficiency of the wavelet transform in characterizing and synthesizing positive long-range dependent data, particularly network traffic loads.

We have presented several methods for generating simulated traces of this kind of traffic. Hopefully, the use of simulation studies will help us discover the effects of these new traffic characteristics on the general performance of the communication networks.

References

[1]   B. Tsybakov and N. Georganas, "Self-Similar Processes in Communications Networks", IEEE Transactions on Information Theory, Vol. 44, No. 5, September 1998.

[2]   B. Tsybakov and N. Georganas, "On Self-Similar Traffic in ATM Queues" IEEE/ACM Transactions on Networking, Vol. 5, No. 3, june 1997.

[3]   B. Tsybakov and N. Georganas, "Analysis of an ATM Buffer with Self-Similar ('Fractal') Input traffic", Proceedings of INFOCOM 95 (?)

[4]   V. Paxson and S. Floyd, "Wide Area Traffic: The Failure of Poisson Modeling", IEEE/ACM Transactions on Networking, Vol. 3, No. 3, june 1995.

[5]   P. Abrey and D. Veitch, "Wavelet Analysis of Long-Range-Dependent Traffic", IEEE Transactions on Information Theory, Vol. 44, No. 1, January 1998.

[6]   H. Michiel and K. Laevens, "Teletrafic Engineering in a Broadband Era", Proceedings of the IEEE, Vol. 85, No. 12, December 1997.

[7]   J. Beran, R. Sherman and M. Taqqu, "Long-Range Dependence in a Variable-Bit-Rate Video Traffic", IEEE Transactions on Communications, Vol.43, No 2/3/4, February/March/April 1995.

[8]   D. Heyman and T. Lakshman, "What are the implications of Long-Range Dependence for VBR-Video Traffic Engineering?", IEEE/ACM Transactions on Networking, Vol. 4, No. 3, june 1996.

[9] M. Parulekar and A. Makowski, "Tail probabilities for a multiplexer with self-similar traffic", Proceedings of INFOCOM 96, San Francisco, April 1996.

[10] M. Krunz and A. Makowski, "A source model for VBR Video Traffic Based on M/G/∞ Input Processes", Proceedings of INFOCOM 98, San Francisco, April 1998.

[11] S. K. Agrawal, C. F. Barry, V. Bannai and L. Kazovsky, "Characterization, Adaptive Traffic Shaping and Multiplexing of Real-time MPEG-II video", Proceedings of the SPIE, Vol. 2915, 1997.

[12] C. Weinstein, "Fractional Speech Loss and Talker Activity Model for TASI and for Packet Switched Speech", IEEE Transactions on Communications, August 1978.

[13] Alzate, M. "Modelos de Tráfico para Simulación de Redes de Comunicaciones", Reporte de Investigación Número 5, Maestría en Teleinformática, Universidad Distrital FJC, 1995.

[14] Peitgen, H. and Saupe D. (editors) "The Science of Fractal Images", Springer-Verlag, New York, 1988.

[15] Riedi, R.H. Crouse, M. S., Ribeiro, V.J., and Baraniuk, R.G., "A multifractal wavelet model with application to Network traffic", IEEE Trans. on Info.Theory, Vol.45, N.3, April 1999.