A REALISTIC MODEL FOR SELF-SIMILAR ETHERNET LAN TRAFFIC IN SimATM – AN ATM NETWORK SIMULATOR: DESIGN AND PERFORMANCE IMPLICATIONS*

Ernesto L. Andrade Neto, Antônio M. Alberti, Dalton S. Arantes, Leonardo S. Mendes
DECOM-FEEC/UNICAMP
P.O. Box 6101
13081-970 - Campinas - SP – BRAZIL
phone: +55-19-788-2053 fax: +55-19-289-1395
e-mail: ernestoa, alberti, dalton, lmendes@decom.fee.unicamp.br

Abstract – This paper presents a realistic model for Ethernet LAN Traffic that takes into account the network constraints. The design of the model is presented along with a brief description of SimATM, an ATM network simulator developed at UNICAMP. The resulting self-similar model is compared with real network data from Bellcore Ethernet traffic measurements. Results of simulations with synthetic traffic traces are presented for a simple ATM network. These results clearly indicate that simple Poisson traffic models underestimate real network losses by several orders of magnitude.

1. Introduction

A number of recent empirical studies of traffic measurements from a variety of working packet networks [1] [2] have convincingly demonstrated that actual network traffic is self-similar or long range dependent in nature (i.e., bursty over a wide range of time scales). This is in sharp contrast to commonly made traffic modeling assumptions (i.e., Poisson models). Since an accurate estimation of network performance is critical for the success of broadband networks, then network simulation with self-similar models is required in order to capture the statistical characteristics of the actual traffic [2].

Several methods have been used for generating self-similar synthetic traces: asymptotically self-similar fractional Arima [3], exactly self-similar fractional Brownian motion (fBm) [4], aggregation of several heavy tailed ON/OFF sources [2], among others. A description of traditional (short range dependent) and non-traditional (long range dependent) traffic models can be found in [5].

The need to test and characterize actual network traffic in ATM network models have lead us to the development of a realistic self-similar Ethernet LAN traffic generator. Simulations with this traffic model will be particularly useful in the performance evaluation of Virtual LANs (VLANs) interconnection. As is well known, VLANs interconnection will be one of the foreseen major uses of B-ISDN. The generator is based on an interactive search for self-similar sequences synthesized from fractional Gaussian noise (fGn) power spectrum inversion with the fast Fourier transform (FFT) [4] that fits the following desired parameters: Hurst parameter (self-similarity degree), mean packet size, packet size variance and lower and upper packet size limits. Sequences with useful sizes for network simulation (order of 10^5-10^6 samples) can be synthesized in a reasonable amount of time, considerably smaller than sequence simulation time. The synthetic Ethernet sequences generated by the proposed model can be used in different ATM congestion control schemes and Connection Admission and Control (CAC) algorithms, as well as in the characterization of network behavior inside the SimATM simulation framework.

The remaining parts of this paper are organized as follows. The second section presents an overview of SimATM. The third section discusses the Ethernet traffic model and its implementation's details. The fourth section describes the model's results analysis and validation. The fifth section shows the results of model's application to a simple simulated ATM network. The sixth section draws final conclusions, comments and directions for future work.

2. SimATM Network Simulator

The starting development point of SimATM was the SimNT simulator developed in the Department of Communications at FEEC-UNICAMP, which has a general purpose simulation architecture but was initially applied to optical systems simulation [6]. The initial simulator design ideas were based on the NIST ATM simulator [7]. However, present simulator

* Partially supported by FAPESP (under grants 96/12372-2 and 97/14449-5) and CPqD-Telebrás (contracts 7500/856/97 and 7800/825/97).

structure is now conforming with ITU-T and ATM Forum specifications [8-16]. As a result, all simulator source code is written in C++ and does extensive use of object modeling in its architecture. The first simulator design goal was to integrate ATM simulation in SimNT, but some important differences between the simulation approaches (one needs a *data driven kernel* for optical systems in SimNT, while the ATM simulator needs an *event driven kernel*) prevented us from integrating both simulators in the same kernel. Figure 1 shows the present structure of the SimATM simulator. The simulation *kernel* has an *event manager*, an *event queue*, a *command interpreter* and supports several instances of ATM networks running one at time. The simulator runs at cell level of ATM networks. The simulator models are divided into four categories: *equipment models*, *layer models*, *application models* and *queueing system models*. The ATM equipment models consist of broadband terminal equipment (B-TE) and ATM switch. The modeled layers consist of the AAL 5 [10], the ATM layer (one model for BTE and another for ATM switch) and the physical cell based layer common to all equipment. All BTEs run ITU-T AAL 5. The ATM application models consist of traffic sources for CBR or VBR mode and general traffic transmitter and traffic receiver for these sources. The queueing system model during simulation gathers all queueing system's statistics in a chosen equipment buffer. The main statistics provided by the simulator are: cell loss, cell delay, link utilization, number of cells transmitted and cell transit time from source to sink. Therefore, the simulator statistics allows the complete characterization of queueing systems' behavior.

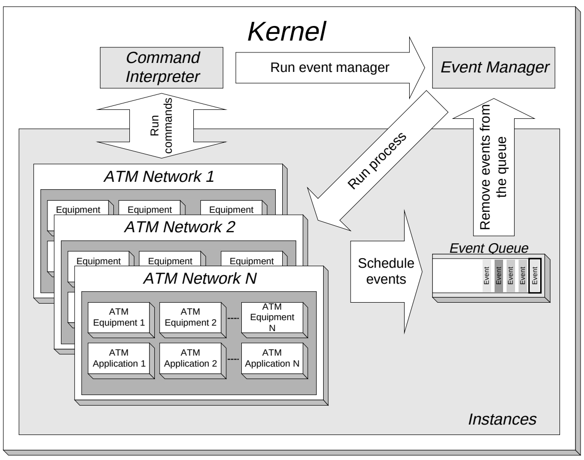

Figure 1: SimATM simulation framework.

The simulation statistics can be logged to a file in constant intervals of simulation time or each time that an event being tracked occurs (like cell arrival or cell drop), but notice that event sampling can generate huge log files. The simulation architecture allows that new equipment models be added to the simulation framework or that existing models be modified in a particular behavior to add a new feature.

3. Ethernet Traffic Model

3.1 Self-Similar Process

A sequence $\{X_1, X_2, ..., X_n\}$ of weakly stationary random variables is said to be *long range dependent* (LRD) if:

$$\sum_{n=1}^{\infty} Cov\{X_1, X_n\} = \infty \qquad [1]$$

that is, its autocovariance function $Cov\{X_1, X_n\}$ is nonsummable, otherwise it is called *short range dependent.* If $\mathrm{Var}\{X_1 + ... + X_n\}$ of a long range dependent sequence grows at rate n^{2H}, where $H \in (1/2, 1]$, then the number H is called the Hurst parameter of the sequence. The long range dependence rigorously applies only to infinite time series. The simplest models with long range dependence are self-similar processes, which are characterized by hyperbolically decaying autocorrelation functions.

A function X_t is strictly self-similar if for some H and $\forall \alpha > 0$, $X_{\alpha t} = \alpha^H X_t$. A process X_t is called *statistically self-similar* with self-similarity parameter (or Hurst Parameter) H if, for any $\alpha > 0$, the processes $X_{\alpha t}$ and $\alpha^H X_t$ have the same finite dimensional distributions. It is called asymptotically self-similar if the process $X_{\alpha t}$, suitably normalized, converges weakly to a self-similar process when $\alpha \to \infty$.

A square integrable process X_t is called second order self-similar with self-similarity parameter H if, for any $\alpha > 0$, the process $X_{\alpha t}$ and $\alpha^H X_t$ have the same second order characteristics. It is called asymptotically second order self-similar if the second order characteristics of $X_{\alpha t}$ suitably normalized, converge to those of a second order self-similar process when $\alpha \to \infty$. Self-similar and asymptotically self-similar processes are particularly attractive models mainly because long range dependence can be characterized by a single parameter, the *Hurst parameter* H, which can be estimated using many different methods, such as Whittle's procedure[4][17,18], aggregate variance plots[17,18], R/S statistics[17,18], wavelets, fractal dimension, etc.

3.2 Model Implementation

The analysis of Bellcore measured Ethernet traces [1][2], confirming that actual Ethernet traffic traces are self-similar over a wide range of time scales, and results drawn from [4], about feasibility of fGn generation with fGn power spectrum synthesis and power spectrum inversion through FFT, have motivated us for the construction of a realistic Ethernet source model that represents actual Ethernet link traffic data to test simulated ATM networks. The fGn synthesis method is one of the fastest self-similar sequence generation algorithms for sequential machines [4] and is very appropriate for the present application. The Ethernet protocol forces all packets to have at least the minimum size of 64 bytes and at most the maximum size of 1518 bytes. For practical simulation purposes, the synthetic traffic source must generate a traffic stream with a specific *average value*, a *variance value* and a given *Hurst parameter*, which characterizes the desired degree of self-similarity. It is also very important that the synthetic traces be kept *constrained to the above packet size limits*.

The method that we have used for generating the traces takes into account all the above constraints. It is an heuristic that accomplishes an interactive process in order to synthesize a traffic stream with a desired self-similarity (Hurst parameter), mean packet size, packet size variance, and minimum and maximum packet size limits. A block diagram of the proposed algorithm can be seen in Figure 2. The first step in this process is to produce a pseudo-self-similar sequence with zero mean and unitary variance using an adaptation of the FFTFGN algorithm [4]. The resulting sequence is then normalized with the transformation $Y_i = 2^{X_i}$, which preserves self-similarity [4] and results in a positive valued sequence. After this, the sequence is normalized to desired mean and variance. In the normalized sequence we applied the upper and lower limits and evaluated the sequence's Hurst parameter using the variance method with equally spaced points on a logarithmic scale. If the Hurst parameter is within the desired tolerance range, the resulting sequence is saved to a file and, if necessary, the search is continued for more uncorrelated sequences using a new random generator seed. If the Hurst parameter is out of the specified range, the sequence search continues until it finds the desired number of sequences or the algorithm reaches the maximum limit of tested sequences.

When this step is concluded, the resulting sequences are uniformly distributed in time in order to achieve a desired mean link utilization for simulation purposes. The time to find a sequence with the desired parameters varies with the initial guess for the random generator's seed, and usually greater values of Hurst parameter take more iterations to produce a sequence.

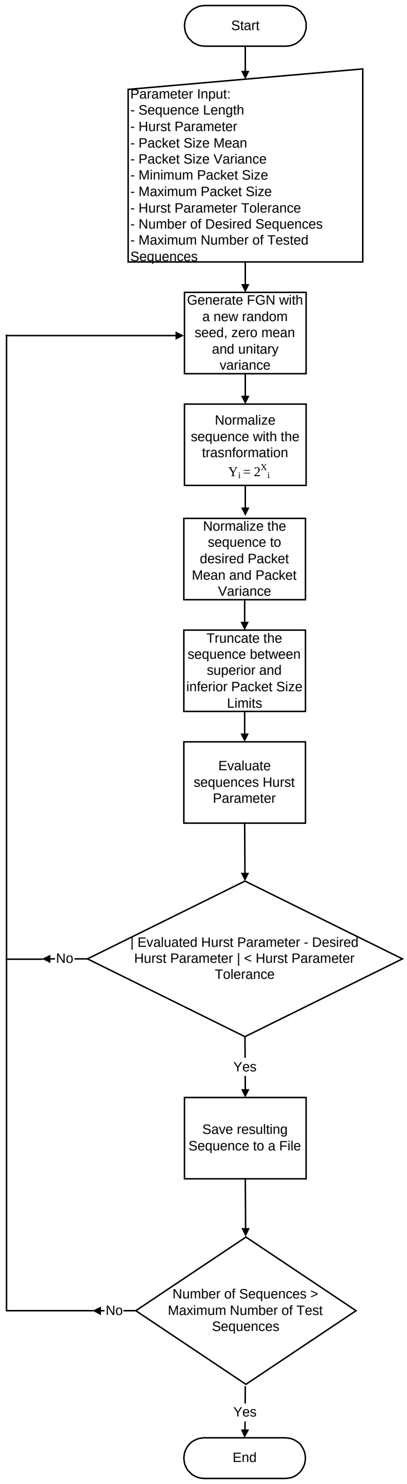

Figure 2: Self-similar model algorithm.

4. Model Validation

Throughout the modeling process, measurements from Bellcores's Ethernet LAN at Morrinston Research facility (file BC-pAug89.TL available in the Internet [19]) have been used in the construction and parameterization of the model. These measurements have verified the presence of self-similarity and estimates of Hurst parameter, using the aggregated variance method, have resulted in H=0.8. The mean packet size for this trace is about 434 bytes and its standard deviation is about 484 bytes.

The main methods used in the analysis of synthetic sequences are the aggregated variance method (for the Hurst parameter estimation) and the autocorrelation function plots. Figure 3 shows samples of generated sequences with H=0.50, for an uncorrelated packet sequence, and with H=0.90 for a packet sequence with high degree of self-similarity. These samples show significant visual differences in the burstiness of the uncorrelated and the highly correlated synthetic traces. Figures 4 and 5 show the autocorrelation functions and aggregated variance, respectively, for synthetic sequences and for Bellcore's Ethernet LAN data. In Figure 4 the trace with H=0.50, as expected, has only short range dependence and no significant degree of self-similarity. The synthetic trace that approximates Bellcore's data (H=0.80) has almost exactly the same long range dependence and differs only in a short range dependence that is not included in the model.

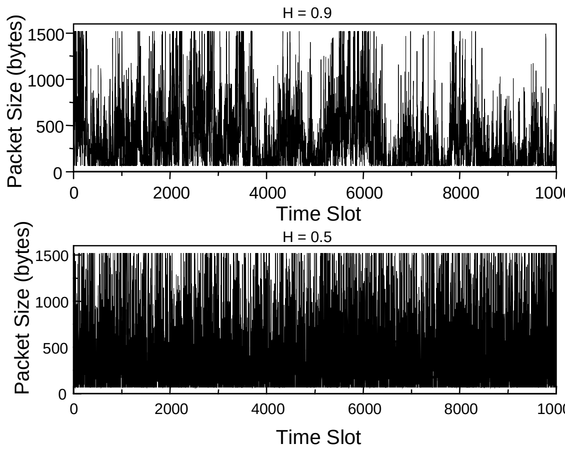

Figure 3: Sample of synthetic traces generated with the model.

In Figure 5, the aggregated variance plot for Bellcore's and synthetic data are also almost identical, differing in the aggregated variance amount mainly due to differences in packet size distribution between them. All the synthesized sequences have approximately the same mean packet size and same packet size variance as Bellcore's Ethernet trace. The trace mean packet size and packet standard deviation data can be seen in Table 1.

The differences between the target mean and standard deviation, that ought to be the same as Bellcore's trace, and the synthesized sequence mean and standard deviation, are due to the sequence's truncation during the process that was used to normalize the sequences between lower and upper packet size limits.

The negligible differences between the autocorrelation functions (Figure 4) and variance plots (Figure 5) for the synthetic traffic traces and the actual Ethernet data confirms that the presented model is an excellent source of self-similar Ethernet traces for ATM networks simulation**.** Thus the model approximates the long range behavior of the real Ethernet traffic, subject to practical network's constraints, with a high degree of fidelity.

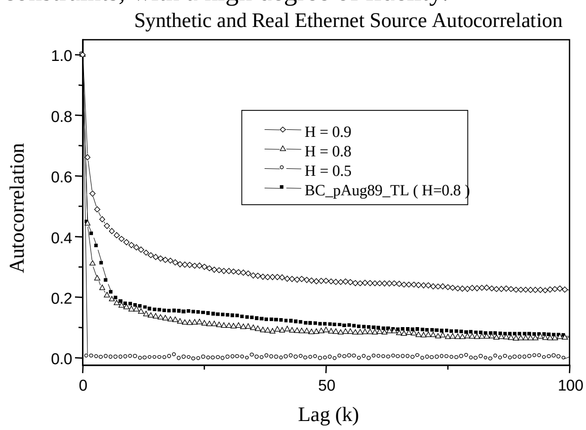

Figure 4: Autocorrelation function plot of Ethernet traffic data.

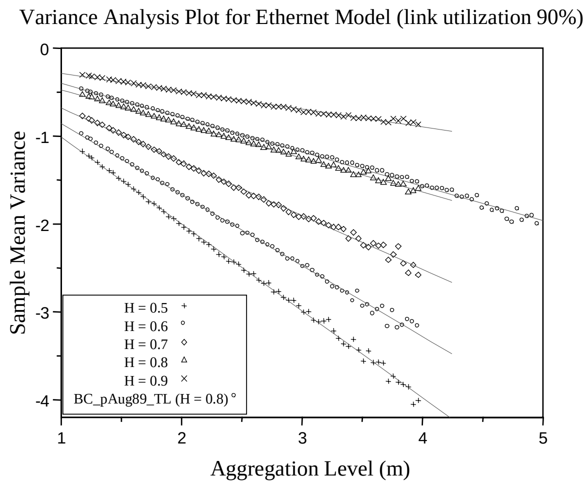

Figure 5: Aggregated variance plots for Ethernet traffic data.

H	Mean	Std. Dev.
0.5	432.33	476.91
0.6	431.95	475.96
0.7	430.75	473.42
0.8	433.44	480.09
0.9	433.58	480.28

Table 1: Packet size mean and standard deviation of synthesized sequences.

5. Performance Implications

Self-similar traffic implies that the process of aggregating “bursty” traffic still results in “bursty” traffic. Thus, cell loss in a network may be much higher than expected if the network traffic is self-similar. A simple experiment was conducted in order to assess the performance implications of self-similar traffic cell loss and buffer size requirement for a simulated ATM network. The simulation framework used for the experiment can be seen in figure 6. The Ethernet source model is connected with ATM switch using a Permanent Virtual Channel (PVC) of 155.52 Mbps. Its data passes across AAL 5, ATM and physical layers reaching a low bandwidth ATM switch output buffer (1.5 Mbps), where the simulation statistics are collected. About 1.2 millions of cells were transmitted from Ethernet source to sink for each trace with different Hurst parameter during the simulations. These cells were used to gather the system’s statistics.

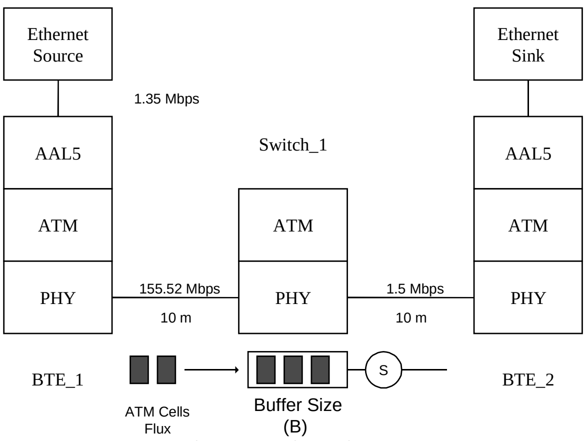

Figure 6: Simulation framework for self-similar source model test.

Self-similar traffic has serious implications on cell loss performance in the network. Figure 7 depicts average cell loss ratio experienced by the Ethernet traffic source, as a function of buffer size at the output port. These experiments are conducted at a high link utilization, the buffer traffic input rate being 90 % of output link rate. Figure 7 shows that increasing buffer size at the output always reduces cell loss ratio, as expected. However, additional buffer sizes are much less effective when traffic source has a high degree of self-similarity. For example, for H=0.5 increasing the size the of buffer from B=10 to B=100 reduces cell loss ratio from about 10^{-1} to below 10^{-3}. On the other hand, for H=0.8 a buffer size of approximately 20000 is needed in order to bring cell loss ratio below 10^{-2}.

Table 2 shows that the higher the degree of self-similarity present in the synthetic traces, the higher will be the mean number of cells stored in the switch output buffer and the mean cell delay in this buffer.

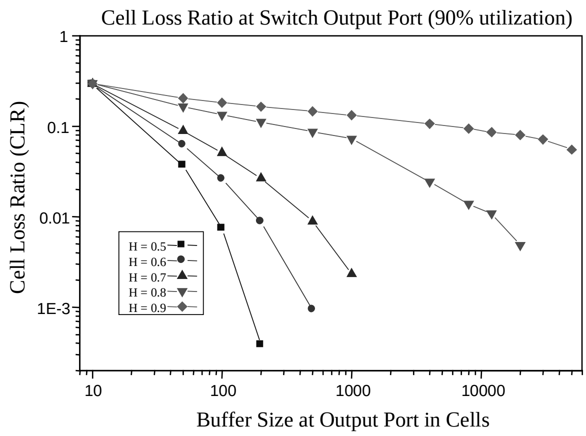

Figure 7: Results of ATM switch cell loss ratio.

Mean Number of Buffer Cells and Mean Cell Delay (Buffer Size B = 100)				
H	Mean N. of Buffer Cells	Buffer Cells Std. Dev.	Mean Cell Delay (s)	Cell Delay Std. Dev.
0.5	28.3182	24.9330	0.0086	0.0007
0.6	32.9631	29.5197	0.0099	0.0011
0.7	32.6862	32.2219	0.0098	0.0012
0.8	34.5327	35.5483	0.0103	0.0020
0.9	36.9848	39.5718	0.0110	0.0059

Table 2: Mean number of cells and cell delay in the ATM output buffer as a function of Hurst parameter (H).

6. Conclusions

This paper has presented a synthetic workload model that is capable of modeling self-similar Ethernet traffic. The model captures the autocorrelation and long range dependence that has been observed in recent Ethernet LAN measurements [1] and [2]. Notice that the generated self-similar process are always constrained by the Ethernet Network parameters.

The simulation results in this paper confirm that self-similar traffic has potentially serious implications on ATM networks. Cell delay, cell jitter and cell loss may be significantly higher than predicted by simple traffic models. Poisson traffic models, as opposed to self-similar traffic models, may underestimate ATM cell loss by two or three orders of magnitude. Self-similar traffic models are thus a valuable tool as an actual stress test to ATM networks.

Acknowledgments

We would like to thank FAPESP and CPqD-Telebrás for their financial support to this work.

References

[1] W. Leland et al., "On the Self-Similar Nature of Ethernet Traffic (Extend Version)", IEEE/ACM Transactions on Networking, pp 1-15, february, 1994.
[2] W. Willinger et al., "Self-Similarity Through High-Variability: Statistical Analysis of Ethernet LAN Traffic at the Source Level", IEEE/ACM Transactions on Networking, vol 5, no 1, pp 71-86, 1997.
[3] P. S. Koskoszka et al., "Parameter Estimation for Infinite Variance Fractional Arima", The Annals of Statistics, vol 24, pp 1880-1913, 1996.
[4] V. Paxson, "Fast Approximation of Self-Similar Network Traffic", technical report Lawrence Berkeley Laboratory, 1995.
[5] A. Adas et al., "Traffic Models in Broadband Network", IEEE Communications, pp 82-89, june, 1997.
[6] M.R.N. Ribeiro, H. Waldman, J. Klein, L.S. Mendes, "Error Rate Parameters For Modeling of Optically Amplified Transmission Systems", IEEE Journal on Selected Areas in Communication, Vol 15, No. 4, pp 707-716, may, 1997.
[7] N. Golmie, A. Koening, D. Su, "The NIST ATM Network Simulator Operation and Programming Manual", NIST Internal Report 5703, august, 1995.
[8] ITU-T, "I.321 - B-ISDN Protocol Reference Model and Its Application", 1995.
[9] ITU-T, "I.361 – B-ISDN ATM Layer Specification", 1995.
[10] ITU-T, "I.363 – B-ISDN User-Network ATM Adaptation Layer Specification", 1993.
[11] ITU-T, "I.371 – Traffic Control and Congestion Control", 1995.
[12] ITU-T, "I.432 – B-ISDN User-Network Interface – Physical Layer Specification", 1993.
[13] ITU-T, "I.610 – B-ISDN Operation and Maintenance, Principles and Functions", 1995.
[14] ITU-T, "I.731 – Types and General Characteristics of ATM Equipment", 1996.
[15] ITU-T, "I.732 – *Functional Characteristics of ATM Equipment*", 1996.
[16] ATM Forum UNI 3.1, "ATM Forum User-to-Network Interface Specification, Version 3.1", The ATM Forum Technical Committee, 1994.
[17] M. Taqqu et al., "Estimators for Long-Range dependence: an empirical study", Fractals, vol 3, no 4, pp 785-788, 1995.
[18] J. Roberts et al eds., "Broadband Network Teletraffic – Final Report of Action Cost 242", Springer Verlag, 1996.
[19] ftp://ita.ee.lbl.gov/traces/BC-pAug89.TL.Z