Simulation Modeling of Self-Similar Teletraffic

Dimitar Radev[1], Dragan Stankovski[2], Svetla Radeva3

Abstract — **The paper consider simulation modeling of self-similar teletraffic. Steady-state simulation of self-similar queuing processes was provided with fixed-length sequence generators. A new algorithm for buffer overflow with limited relative error is developed. The implementation of suggested simulation approach for stochastic and long range dependence teletraffic source models is shown. Buffer overflow simulation for finite buffer single server model under self-similar traffic load SSM/M/1/B is considered.**

Keywords — **Communication systems, Modeling, Queuing analysis, Simulation software, Stochastic processes**

I. Introduction

Apromising method in integration of technologies and services in digital communication networks is the simulation modeling of sources, traffic and their management. The simulation modeling is an abstraction of the real interaction between sources, which leads to simplification and limited performance of the behavior of the elements in real communication network. This is realized with the help of stochastic processes where the generated traffic is based on study of the behavior of certain classes of probability processes like time series analysis, long range dependence stochastic processes, chaotic time series, discrete-time Markov state models, queuing theory [2].

Recent studies of real teletraffic data show that teletraffic exhibits self-similar (or fractal) properties over a wide range of time scales [1]. The self-similar nature of teletrafic (in sense of long-range dependent behaviour is exhibited over a range of time scales: milliseconds, seconds, minutes and hours) can have a significant impact on network performance. The properties of self-similar teletraffic are very different from properties of traditional models based on Poisson, Markov-modulated Poisson, and related processes [2]. The use of traditional models in networks characterized by self-similar processes can lead to incorrect conclusions about the performance of analyzed networks.

The traditional models can lead to over-estimation of the network performance/quality, insufficient allocation of communication and data processing resources, and difficulties in ensuring the quality of service (QoS) expected by network users. Thus, full understanding is that the self-similar nature in teletrafic is an important issue.

Self-similar teletraffic is observed in LAN and WAN, where superposition of strictly independent alternating ON/OFF traffic models whose ON- or OFF-periods have heavy-tailed distributions. In ATM network traffic self-similar traffic arriving at an ATM buffer results in a heavy-tailed buffer occupancy distribution and buffer cell loss probability decreases with the buffer size, not exponentially as in traditional Markovian models, but hyperbolically.

Other implementation of traffic self-similarity is in Internet traffic, where many characteristics of WWW can be moddeled using heavy-tailed distributions to help user requests for documents and the distribution of WWW document sizes. In TCP/IP network traffic the transfer of files or messages show that the reliable transmission and flow control mechanisms of TCP serve to mainain long range dependent structure included by heavy-tailed file size distributions. The relationship between self-similar traffic and network performane is presented in [1].

The self-similarity observed in video traffic give posibility for developing models for VBR video traffic using heavy-tailed distributions. The autocorrelation of the VBR video sequence decay hyperbolically and can be modelled using *fractional autoregressive integrated moving-average* (F-ARIMA) and *fractional Gaussian noise* (FGN) self-similar processes.

The impact of self-similar models on queueing performance is signifficant and the main trends in such findings are connected with (a) permission traffic modelling for high-speed networks, (b) efficient simulation of actual network traffic and (c) analysing queueing models and protocols under realistic traffic scenarios.

II. Self-similar processes

Self-similarity can be classified into two types: deterministic and stochastic. In the first type, deterministic self-similarity, a mathematical object is assumed to be self-similar (or fractal) if it can be decomposed into smaller copies of itself. This work is focused on stochastic self-similarity. In that case, probabilistic properties of self-similar processes remain unchanged or invariant when the process is viewed at different time scales. This is in contrast to Poisson processes that lose their burstiness and flatten out when time scales are changed. One can distinguish two types of stochastic self-similarity. A continuous-time stochastic process Y_t is strictly self-similar with a self-similarily Hurst parameter H ($1/2<H<1$), if Y_{ct} and $c^H Y_t$ (the rescaled process with time scale ct) have identical finite-dimensional probability for any positive time stretching factor c. When the weakly continuous-time self-similar process Y_t has stationary increments, i.e., the finite-dimensional probability distribution of $Y_{t_0+t} - Y_{t_0}$ do

[1] Dimitar Radev is Professor with the Communication Technique and Technology Department, University of Russe, 7017 Russe, e-mail: dradev@abv.bg.
[2] Dragan Stankovski is PhD student with the Communication Technique and Technology Department, University of Russe, 7017 Russe, e-mail: draganstankovski@gmail.com
[3] Svetla Radeva, is Associate Professor at of UACEG Sofia, Bulgaria e-mail: radeva.s@abv.bg.

not depend on t_0, than such process can be constructed as a stationary incremental process $X=\{X_i=Y_{i+1}-Y_i: i=0,1,2\ldots\}$. For second-order self-similar process the self-similarity is observed at the mean, variance and autocorrelation levels. The process X is asymptotically second-order self-similar with self-similarity parameter H ($0.5<H<1$) if for any block size m, the process $\{m^{1-H}X_k^{(m)}: k=1,2\ldots\}$ has the same covariance structure. Modeling and simulation of self-similar traffic can be performed with the generators of synthetic self-similar sequences, divided into two practical classes: the sequential generators and the fixed-length sequence generators.

In this work under consideration are fixed-length sequence generators, which are implemented for buffer overflow at single server models.

III. Fixed – Length Sequence Generators for Self – Similar Teletraffic

Markovian models for self-similar traffic require including several control parameters with a wide range of input values, like size of the sequence, Hurst parameter, scale parameter, vanishing moment, etc. The most appropriate controlling of these values of self-similar processes is realized with fixed-length sequences generators [3]. Here is used a generator of pseudo-random self-similar sequences based on fractional Gaussian noise and Daubechies wavelets (DW) [3], called FGN-DW approach.

Wavelet analysis transforms a sequence onto a time-scale grid, where the term scale is used instead of frequency. The wavelet transform delivers good resolution in both time and scale, as compared to the Fourier transform, which provides only good frequency resolution. The developed algorithm consists of the following steps.

Algorithm for generating of FGN-DW pseudo-random self-similar sequences**:**

Step 1: Given: Hurst parameter H. Start for $i=1$ and continue until $i=n$. Calculate a sequence of values $\{f_1,f_2,\ldots,f_n\}$ using

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

where $c_f=\sigma^2(2\pi)^{-1}\sin(\pi H)\Gamma(2H+1)$, $O(\cdot)$ represents the residual error and $f_i=\hat{f}(\pi i/n;H)$, the value of the frequencies f_i corresponds to the spectral density of an FGN process for f_i ranging between $\pi/n \div \pi$.

Step 2: Multiply $\{f_i\}$ by realizations of the independent exponential random variable with the mean of one to obtain $\{\hat{f}_i\}$, because the spectral density estimated for a given frequency is distributed asymptotically as the independent exponential random variable with the mean $f(\lambda,H)$.

Step 3: Generate a sequence $\{Y_1,Y_2,\ldots,Y_n\}$ 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 preserves the spectral density corresponding to $\{\hat{f}_i\}$. It also makes the marginal distribution of the final sequence normal and produces the requirements for FGN.

Step 4: Calculate two synthetic coefficients of orthonormal Daubechies wavelets. The output sequence $\{X_1,X_2,\ldots,X_n\}$ representing approximately self-similar FGN process is obtained by applying the inverse Daubechies wavelets transformation of the sequence $\{Y_1,Y_2,\ldots,Y_n\}$.

The normalized arrivals sequence for n=10 000 observations received after wavelet transform is shown on Fig.1 with FGN-DW algorithm. The sequence is normalized for integer number of arrivals in interval [0, 40] for following parameters: H=0,75; Scale=4 and Vanishing Moment=6. The reveived results showed that the generator based on FGN-DW method demonstrated a hidh level of accuracy, was fast and can be implemented for long sequences with long-range dependent properties.

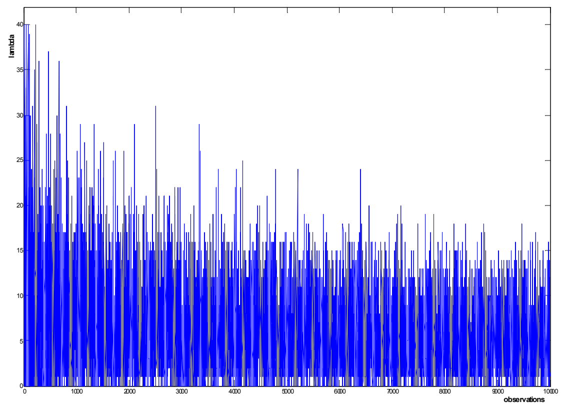

Fig. 1. Arrivals sequence after normalization with wavelet transform.

IV. Steady – State Simulation of Self – Similar Queuing Systems

There exists a significant difference in the queueing performance between traditional models of teletraffic such as Poisson processes and Markovian processes, and those exhibiting self-similar behaviour. More specifically, while tails of queue length distributions in traditional models of teletraffic decrease exponentially, those of self-similar traffic models decrease much slower. Under investigation are potential impacts of traffic characteristics, including the effects of self-similar behaviour on queuing and network performance, protocol analysis, and network congestion controls.

The steady-state simulation of self-similar queueing systems inclide generation of self-similar traffic, simulation of queing process and simulation of overflow probability. Here this is illustated on buffer overflow simulation for SSM/M/1/B queueing systems (B<∞) (i.e. queuing systems with the finite buffer capacity) at self-similar queuing processes. In this case, the difference from M/M/1/B queuing system is that the arrival rate λ_j into SSM/M/1/B queuing system is not a constant value, where SSM/M/1/B queuing system has exponential service times with constant rates $1/\mu$ as is shown on Fig. 2.

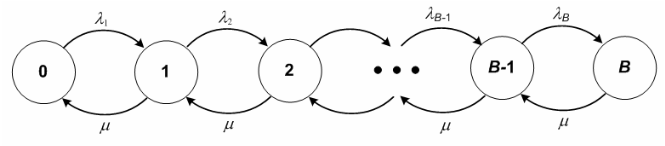

Fig. 2. State transition diagram for a SSM/M/1/B queuing system.

The flow balance equations are

$$\begin{aligned} &\lambda_j = \lambda(i,n,H); && j = 1,2,..., B \\ &\lambda_j = 0; && j \geq B+1 \\ &\mu_j = \mu; && j = 1,2,..., B+1 \end{aligned} \qquad (2)$$

This system is stable whether the throughput $\rho = \frac{\lambda(i,n,H)}{\mu} < 1$. Let consider two separated cases: ρ=1, and $\rho \neq 1$. For $j = 0,1,2,..., B$ the distribution of number of customers in system is $P_j = \rho^j P_0$, which is determined according to

$$\begin{aligned} &P_j = \frac{\rho^j(1-\rho)}{1-\rho^{B+1}}; && \rho \neq 1 \\ &P_j = \frac{1}{B+1}; && \rho = 1 \end{aligned} \qquad (3)$$

Hence, the rate at which the custumers are lost (blocked) is λP_B. The queuing process is described with the steady-state simulation algorithm, presented on Fig.3.

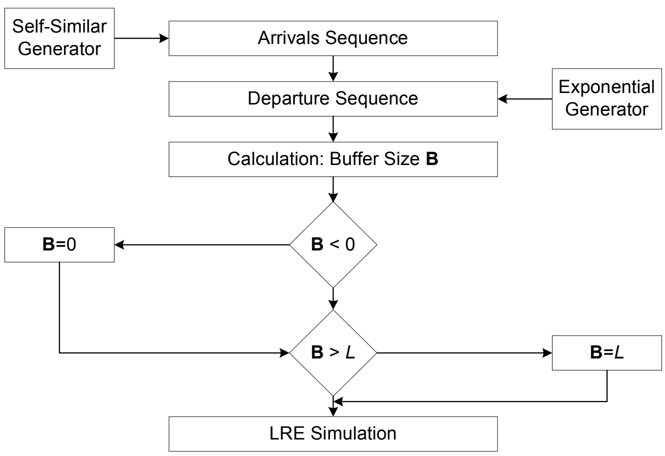

Fig. 3. Algorithm of queuing process.

The simulation is realized with RESTART method [4], which is a variant of splitting where any chain is split by a fixed factor when it hits a level upward, and one of the copies is tagged as *original* for that simulation level [5]. When any of those copies hits that same level downward, if it is the original it just continues its path, otherwise it is killed immediately. This rule applies recursively, and the method is implemented in a depth-first fashion, as follows: whenever there is a split, all the non-original copies are simulated completely, one after the other, then simulation continues for the original chain. The calculation of buffer size for all sequences gives possibilities for determining the overflow probability.

V. LIMITED RELATIVE ERROR SIMULATION OF BUFFER OVERFLOW

The limited relative error (LRE) [5] gives posibility to determine the complementary cummulative function of arrivals at single server buffer queues with Markov processes. For describing the princilpes of LRE for steady-state simulation in disctere-time Markov chains consider homogenious two-node Markov chain, wich is extended to common disctere-time Markov chain, consisting of k+1 nodes with states, respectively $S_0,S_1,\ldots,S_k$. We receive the random generated sequence $x_1,x_2,...,x_t,x_{t+1},\ldots$ for x=0,1,…, k, for wich exists transition for state S_j at the time t, e.g. x_t=j and there are no constraints to the parameters of transition probabilities

$$p_{ij} = \mathrm{P}(j \mid i); \quad (i,j) = 0,1,...,k; \quad \sum_{j=1}^{k} p_{ij} = 1. \qquad (4)$$

There are no absorbing states S_i at p_{ii}=1 for all stationary probabilities P_j j=0,1,…,k, which satisfy the constraint condition

$$0 \leq \mathrm{P}_j < 1; \quad \sum_{j=0}^{k} \mathrm{P}_j = 1. \qquad (5)$$

The cummulative distribution $F(x)$ can be presented as

$$\left.\begin{aligned} &F(x) = F_i; \quad (i-1) \leq x < i; \quad i = 1,2,...,k+1; \\ &F_i = \sum_{j=0}^{i-1} \mathrm{P}_j; \quad F_0 = 0; \quad F_{k+1} = 1; \end{aligned}\right\}. \qquad (6)$$

For simulation of (k+1) nodes Markov chain more significant is complementary cummulative distribution $G(x)$=1-$F(x)$, wich togather with the local correlation coefficient $\rho(x)$ can be determined with the help of limited relative error approach. After determining of two node Markov chain, via changing of the states n times can be received an estimation of the local correlation coefficient $\hat{\rho}(x)$, which can connect the number of transitions througt dividing line $a_i \approx c_i$, with the total number of observated events at left side l_i=n-d_i $(\beta = 0,1,...,i-1)$, and at rigth side d_i $(\beta = i, i+2,...,k)$.

The value of simulated complementary cumulative distribution $\hat{G}_i$ can be determined directly via relative frequency d_i/n, if there are enough number of samples

$$n \geq 10^3; \quad (l_i, d_i \geq 10^2); (a_i, c_i, l_i - a_i, d_i - c_i) \geq 10 \qquad (7)$$

The posterior equations can be used for the complementary function $\hat{G}(x)$, for tha average number of generated values of $\hat{\beta}$, for the local correlation coefficient $\hat{\rho}(x)$, for the correlation coefficient Cor[x] and for the relative error RE[x]

$$\hat{G}(x) = \hat{G}_i = d_i / n$$
$$\hat{\beta} = \frac{1}{n}\sum_{i=1}^{k} d_i$$
$$\hat{\rho}(x) = \hat{\rho}_i = 1 - \frac{c_i / d_i}{1 - d_i / n} \qquad i-1 \le x < i \quad i = 1,\dots,k \tag{8}$$
$$\mathrm{Cor}[x] = \mathrm{Cor}_i = (1+\hat{\rho}_i)(1-\hat{\rho}_i)$$
$$\mathrm{RE}[x]^2 = \mathrm{RE}_i = \frac{1 - d_i / n}{d_i} \cdot \mathrm{Cor}_i$$

The main advantage of this approach is that the relationships between transitions c_i is reached with routine statistical calculations. The necessary total number of simulation trails n is determined by the maximal relative error $\mathrm{RE}_{\max}[x]^2$ and by the less value of the function $G(x)$, presented as $G_{\min} = \hat{G}_k$ in approximation equation

$$n = \frac{(1 - G_{\min})}{G_{\min}.\mathrm{RE}_{\max}[x]^2} \approx \frac{\mathrm{Cor}_k}{\hat{G}_k.\mathrm{RE}_{\max}[x]^2}; \qquad \mathrm{Cor}_k = \frac{1+\hat{\rho}_k}{1-\hat{\rho}_k} \tag{9}$$

This procedure is realized with common version of limmited relarive error algorithm for random discrete sequences.

VI. Experimental Results

To show the effect of self-similarity on probability of buffer overflow, the received experimental results for SSM/M/1/B queuing system are compared with complementary cumulative distribution for classical single server finite buffer queue M/M/1/B as shown on Fig. 4.

For receiving representative and stable results were used sequences of 10 000 observations. With the suggested LRE algorithm were calculated the values of complementary cumulative function $G(x)$ for different buffer size. The calculations were provided with step m=4. The increasing of Hurst parameter leads to insignificant decreasing of overflow probability. For example, for the value of Hurst parameter H=0,6 the overflow probability is $G(L)$=10,45.10^{-2}, and for H=0,9 –G(L)=5,6.10^{-2}.

VII. Conclusion

Steady-state simulation of self-similar queuing processes was provided with fixed-length sequence generator. The simulation is realized with RESTART method. A new algorithm for simulation of buffer overflow probability was developed for self-similar traffic generation with limited relative error. Buffer overflow simulation for finite buffer single server model under self-similar traffic load with suggested algorithm is considered.

An implementation of suggested simulation approach for the stochastic and long range dependence teletraffic source models is shown. Models of SSM/M/1/40 queuing system with different characteristics of self-similarly process of arrivals and different buffer size are presented.

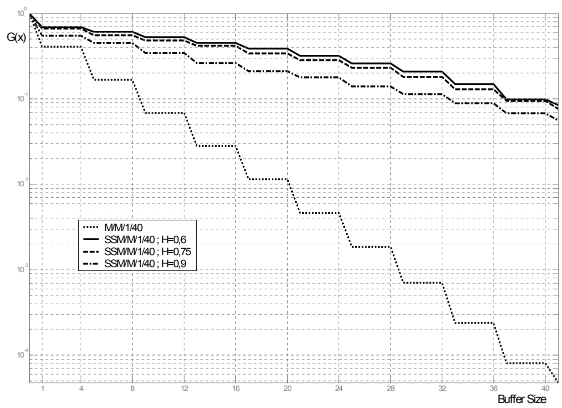

Fig. 4. Simulation of buffer overflow probability (L=41) for SSM/M/1/40 queuing system.

References

[1] J. J. Hae-Duck, "Modelling of Self-Similar Teletraffic for Simulation," PhD Thesises, University of Centerbury, UK, 2002. p. 269.

[2] D. Radev, "Stochastic process generation for traffic flow simulation." *Proc. of the IEEE Conference on Telecommunications in Modern Satellite, Cable and Broadcasting Services TELSIKS'2005*, vol. 2, Nis, Serbia, 2005, pp. 559-562.

[3] P. L'Ecuyer, "Random Number Generators. - Encyclopedia of Operations Research and Management Science." Kluwer Academic Publishers, 2001. pp. 695-702.

[4] M. Villen-Altamirano and J. Villen-Altamirano, "On the efficiency of RESTART for multidimensional systems." *ACM Transactions on Modeling and Computer Simulation*, vol. 16 (3), 2006, pp. 251–279.

[5] C. Gorg and O. Fuss, "Simulating rare event details of ATM delay time distributions with restart/lre". *Proceedings of the IEEE International Teletrafic Congress, ITC16*, 1999, Elsevier, pp. 777–786.