RESEARCH OF THE NETWORK SERVER IN SELF-SIMILAR TRAFFIC ENVIRONMENT

Sergejs Ilnickis[1]

Keywords: network traffic, self-similar traffic, traffic analysis.

***Abstract* - Last scientific publication shows that real network traffic is self-similar and its properties differs from exponential traffic. The goals of this paper were:**

- Analyzing real computer network traffic selecting one of the self-similarity grade determination methods and using gathered from experiments data to prove the self-similar nature of the real network traffic.

- Developing a model, which describes this input self-similar traffic with variable self-similarity grade, and compare results with model with exponential input traffic.

1. Introduction

Last scientific publication shows that real network traffic is self-similar and its properties differ from exponential traffic. The goals of this paper are:

- Analyze real computer network traffic selecting one of the self-similarity grade determination methods and using gathered from experiments data to prove the self-similar nature of the real network traffic.

- Develop a model, which describes this input self-similar traffic with variable self-similarity grade, and compare results with model with exponential input traffic.

- Compare working parameters for the M/M/1 system (with a Poison input flow) and G/M/1 system with self-similar input flow. Analyze queue average length, average waiting time in the queue and the maximal length of the queue in case of different input data flows.

2. Computer network server traffic analysis

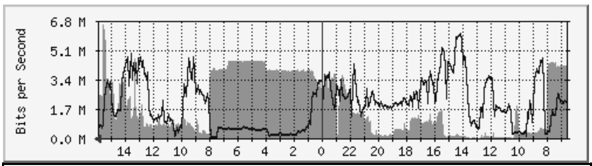
2.1. Description of the researched network

Analysis was done in the corporate network of the company “X”, which is software Development Company and widely using databases inside and outside of corporate network. Maximum network speed was 10 Mbit/s. From a wide range of available servers there was selected the one, which was highly loaded most of the time. The main functions of this server in the network were the following: WEB server, DataBase server and File and Print server.

Results of the measured traffic for this server were gathered using Multi Router Traffic Grapher (MRTG). Data was collected whole month, 24 hours a day with a 5 minutes interval. For further analysis data was divided into groups: one-day traffic, one-week traffic, one-month traffic. Daily traffic also was divided into smaller subgroups depending on network load intensity:

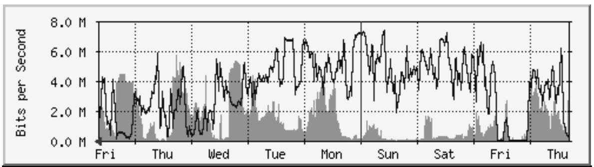
- from 02:00 until 08:00 – low traffic intensity;
- from 08:00 until 14:00 – average traffic intensity;
- from 14:00 until 20:00 – high traffic intensity;
- from 20:00 until 02:00 – maximal traffic intensity;

2.2. Results of the measurement



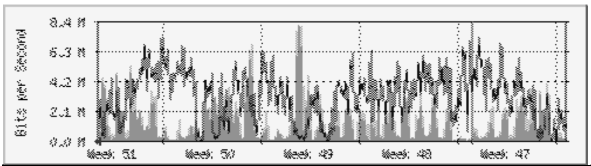
Max In:	6622.5 kb/s (66.2%)	Average In:	1966.3 kb/s (19.7%)
Max Out:	6057.7 kb/s (60.6%)	Average Out:	2010.1 kb/s (20.1%)

Figure 1: Daily traffic



Max In:	5805.9 kb/s (58.1%)	Average In:	1504.2 kb/s (15.0%)
Max Out:	7373.1 kb/s (73.7%)	Average Out:	3647.7 kb/s (36.5%)

Figure 2: Weekly traffic



Max In:	8178.1 kb/s (81.8%)	Average In:	1507.3 kb/s (15.1%)
Max Out:	7282.0 kb/s (72.8%)	Average Out:	3231.4 kb/s (32.3%)

Figure 3: Monthly traffic

2.3. The methods of researching of traffic self-similarity

We assume that traffic is self-similar. Our goal is to prove this statement. For this purpose we will use an absolute moments method.

In given method researched sequence with a length N is divided into blocks with a length of m. For each block we calculate average value $X^{(m)}$ and mean value $\overline{X}$ for whole sequence. After this for each block we find *n* sequence moment:

(1) $$AM_n^{(m)} = \frac{1}{N/m}\sum_{k=1}^{N/m}\left|X^{(m)}(k)-\overline{X}\right|^n .$$

In this equation *n=1* (absolute average value). Then let's designate our sequence as *m* and plot the graph – average values sequence moments dependency from *m*. With a help of plotted points of the graph we construct a smallest square line. The inclination of this line will be named β. From β we can find self-similarity coefficient $H = 1-|\beta|$.

It is necessary that each block's length and blocks quantity will be high. If the sequence hasn't a process with slowly changing dependency, that *H=0.5* and approximation line's inclination will be 0.5. If the process is self-similar, then 0.5<H<1.0 and approximation line's inclination will be less than 0.5.

This method was programmed using MatLab 5.0.

With a help of this program the following graphs were plotted: log-log, dispersion dependency from coefficient *m*, and was defined Hurst's parameter for each day of the week, whole week, whole month and Hurst's parameter depending from the network load for the entire day.

One of the results are shown on the Figure 4.

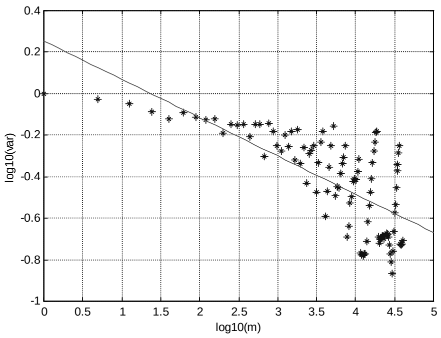


Figure 4. Dispersion-time graph for Monday's data.

Hurst's coefficients calculated with a help of absolute moments method for different time intervals are shown in the Table 1 and 2.

Name of he day	Abs(β)	Hurst's coefficient
Monday	0.1845	0.82
Tuesday	0.2206	0.78
Wednesday	0.2537	0.75
Thursday	0.1620	0.84
Friday	0.2120	0.79
Saturday	0.1518	0.85
Sunday	0.2441	0.76
Whole week traffic	0.2131	0.79

Table 1: Hurst's coefficient for the week traffic

Time interval	Abs(β)	Hurst's coefficient
02:00 - 08:00 Low intensity	0.1845	0.71
08:00 – 14:00 Average intensity	0.2206	0.73

14:00 – 20:00 High intensity	0.2537	0.76
20:00 – 02:00 Maximal intensity	0.1620	0.81
Whole day traffic	0.2184	0.75

Table 2: Hurst's coefficient for one-day traffic.

Daily traffic was specially divided into time groups. There is an opinion that the effect of the self-similarity is depending from traffic intensity and from the type of transmitted information. In the time interval 02:00 – 08:00 traffic intensity is very low and in the same time the Hurst's parameter has the least value. On the working days from 08:00 am traffic intensity are growing up, Hurst's parameter growing up too. After 20:00 till 02:00 when programmers are starting to check produced code in the automatic mode, the flow to the measured server are growing very high and Hurst's parameter is growing up to 0.81. Network traffic in case of maximal intensity (20:00 – 02:00) has very high grade of self-similarity.

After analyzing results, we can make some conclusions:

- traffic in the LAN is self-similar process.
- effect of the self-similarity appears on the wide time scale: from some hours till some months.
- self-similarity coefficient of the traffic differs $\approx 0.7 - 0.85$.
- when the intensity of the traffic is growing up, the self-similarity coefficient is growing up too.
- for the one-day traffic Hurst's coefficient is the same for each intensity;
- for the one-month traffic Hurst's coefficient is ≈ 0.81.

We can conclude, that server traffic has self-similarity in the wide range of the time and it is necessary to predict this cases when networks are projected.

2.4. Modelling of self-similar traffic in GPSS environment

Our goal is to compare working parameters for M/M/1 and G/M/1 systems with a self-similar input traffic using real network state and gathered experimental data. For self-similar input traffic modelling it is necessary to set up all the main parameters, which are characteristic for this type of the traffic.

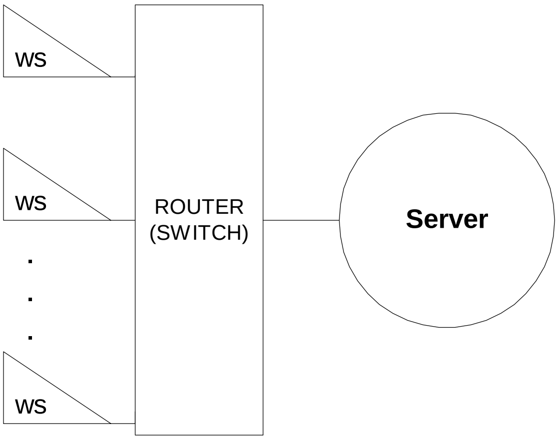
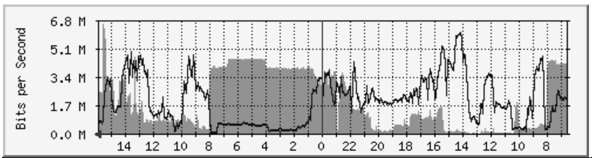


Figure 5. Simplified schema of the analyzed network.

As a sample we will take one-day traffic:



Max In:	6622.5 kb/s (66.2%)	Average In:	1966.3 kb/s (19.7%)
Max Out:	6057.7 kb/s (60.6%)	Average Out:	2010.1 kb/s (20.1%)

Figure 6. One-day traffic.

As we see, average output traffic is up to 2 Mb/s. That means, that channel is loaded for 20 %. We will take this value as average ON-OFF source's working intensity. That means that ON time is 5 times smaller than OFF time. GPSS integrated function allow to generate Pareto distribution with a parameter α. In our experiment parameter
α = 1.4, which corresponds with Hurst's coefficient H = 0.8. Averaging coefficient m we will change from 1 till 10000. The source code of the self-similar traffic modelling program is listed below:

Let's check that modelled flow is self-similar with a Hurst's coefficient H=0.8 (was calculated by absolute moments method).

m	log(m)	Var(x)	log[Var]
1	0	2,371	0,374932
2	0,30103	2,334	0,368101
3	0,477121	2,305	0,362671

5	0,69897	2,251	0,352375
10	1	2,115	0,32531
20	1,30103	1,843	0,265525
50	1,69897	1,297	0,11294
100	2	1,018	0,007748
200	2,30103	0,826	-0,08302
500	2,69897	0,632	-0,19928
1000	3	0,5	-0,30103
2000	3,30103	0,396	-0,4023
5000	3,69897	0,281	-0,55129
10000	4	0,222	-0,65365

Table 3: Dispersion time dependency

Using data from the Table 3 we create Diespersion – Time graph, which is shown on the Figure 7.

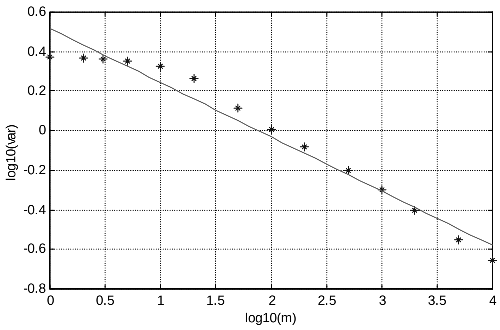


Figure 7. Modelled traffic dispersion graph, β=-0.2744, H=0.8628

The corner of a straight line is -0.2744 and this means that Hurst's parameter's value is equal by 0.8628. This statement is approximately the same, as was measured for the real server traffic in the LAN.

This gives us an opportunity to use developed model for researching the real network and compare working parameters of the flow in case, when on the input we have a self-similar traffic or Poison input flow.

References

[1] Leland W.E., Taqqu M.S., Willinger W. and Wilson D.V. On The Self-Similar Nature Of Ethernet Traffic.- Proc. ACM SIGCOMM'93, San-Fransisco, 1993, p 183-193.

[2] Walter Willinger, Murad S. Taqqu, Robert Sherman and Daniel V. Wilson, "Self-Similarity Through High-Variability: Statistical Analysis of Ethernet LAN Traffic at the

Source Level”. IEEE/ACM Transactions on Networking, Vol. 5, No. 1, 1997
[3] Adrian Popescu, Traffic Self-Similarity, Department of Telecommunications and Signal Processing, University of Karlskrona/Ronneby
[4] Paxson V., Fast Approximation of Self-Similar Network Traffic, preprint Lawrence Berkely Laboratory and EECS Disvision, University of California, 1996
[5] P.Ulanovs, E.Petersons, Modeling of Self-similar traffic in high-performance Computer Networks, RTU, report collection for 42 International Conference, Riga, 2001
[6] Цыбаков Б.С. Модель телетрафика на основе самоподобного случайного процесса. “Радиотехника”, 1999 г., № 5.

Sergejs Ilnickis, Riga Technical University, Lomonosova str. 1, Riga, LATVIA,
E-mail: Sergey_Ilnitsky@canada.com