Riga Technical University

Faculty of Electronics and Telecommunications

Department of Transport Electronics and Telematics

Romans Jerjomins

PhD Program ”Computer Control, Information and Electronic Systems of Transport”

Research of Wireless Local Area Network in Non-stationary Mode

Summary of the doctoral thesis

Supervisor

Dr. hab. ing. sc., professor

Ernests Petersons

RTU Publishing House

Riga 2012

UDK 004.732(043.2)
Je 707 r

Jerjomins R. Research of Wireless Local Area Network in Non-stationary Mode. Summary of the doctoral thesis -R.:RTU Publishing House, Riga, 2012.-48 p.

This work has been supported by the European Social Fund within the project “Support for the implementation of doctoral studies at Riga Technical University”.

Was printed in accordance to the decision taken on 28 June, Protocol Nr. 11, by the ETF Doctorate board ”RTU P-08”.

ISBN 978-9934-10-369-8

A DISSERTATION SUBMITTED TO RIGA TECHNICAL UNIVERSITY IN FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE DOCTOR OF SCIENCE IN ENGINEERING (Dr.sc.ing.)

The presentation of the dissertation with the purpose of earning the Doctoral Degree in Engineering will take place on November 8, 2012, at the Faculty of Electronics and Telecommunications of Riga Technical University, Azenes 12, room 210.

OFFICIAL OPPONENTS:

Professor, Dr. sc. ing. Gunars Lauks
Riga Technical University, Faculty of Electronics and Telecommunications

Professor, Dr. sc. comp. Guntis Barzdins
University of Latvia, Faculty of Computing

Professor, Dr. habil.sc.ing Manfreds Sneps-Snepe
Ventspils augstskola

CONFIRMATION

I confirm that the work contained in the Dissertation submitted by me to Riga Technical University for the Doctor's Degree in Engineering is my own original work and has not previously been submitted by me for a degree at this or any other University.

Romans Jerjomins ....(Signature)

Date: ....

Thesis consists of an introduction, three chapters, conclusion, six appendices, list of references resulting in 148 pages, 108 figures, 17 tables and 141 used reference.

Abstract

Research of various network traffic types proves that it is self-similar. Therefore, recently many works and fundamental monograph is devoted to its research. Self-similar traffic influence on communication devices, such as network switchboards, concentrators and network servers, lead to sharp decrease in their functioning quality, comparing to operating mode in conditions of traditional streams of packets and queries. It reveals in throughput reduction of network nodes, insufficiency of buffer memory and results in increased of denial of service. Usually all researches of self-similar traffic influence were made in stationary operating mode of switched devices.

Author of the given work observed indicators of self-similar traffic at inclusion of communication devices in work, when load swing arise and certain time – relaxation time – was required for device to enter into a normal stationary operating mode. Therefore in the given work the attention is given to relaxation time.

As a result of research data about transient process durations, utilization of buffer memory depending on server load have been obtained.

General description

Work relevance

The global trend of Internet connection methods is that more and more clients go wireless. Due to rapid advances in wireless technology, the Internet becomes more mobile. Not only smart phones become more affordable and ubiquitous, also car manufacturers are looking into leveraging Internet connectivity in order to provide advanced applications on car maintenance - such as monitoring and diagnosis, on road assistance - such as providing route navigation, weather maps and automated toll payments, as well as on passenger entertainment, including various types of Internet applications. Most of today's network connected cars still rely on systems with low-bandwidth connectivity (e.g. GSM or satellite links), which don't correspond to needs of emerging new applications. It is expected that such situation will change quickly. Several car manufacturers are offering Internet connectivity for handful models of cars via 3G network and other manufacturers also consider offering Internet-enabled car applications or linking smart phone applications to cars. The current trend suggests that ten millions of cars will go on-line in next several years. This will emerge innovative car-based Internet applications and services, which can have major impact on both manufacturers and passenger experiences.

In the last few years we have witnessed increasing number of cars connected to the Internet. All indicators suggest that this trend will continue and drive-through vehicles will become soon first class citizens on the Internet.

The problems we face in usual wired network will become worse in drive-through network as all it's clients will be mobile and constantly switching from one access point to another. Specifically the problem of packet congestions when many clients try to get Internet access at the same time what causes bursts of traffic and packet buffering problems.

The objective

The main objective of the current work is to research transient mode in wireless network. It was divided into the following subtasks:

- Analyze self-similarity properties of wireless local area network traffic
- Explore and choose a self-similar process transient mode research method
- Research the impact of traffic parameters on system relaxation time
- Find out the possibility to predict system relaxation time taking into account currently observed traffic parameters

Research methodology

Both experimental and analytical methods are used and compared. Experimental wireless live traffic was analyzed and several simulation models used to gain the final results. Analytical method and approximation is used to model the ideal classical situation and to analyze the data of simulated systems. Simulation tool was developed to assist in research.

Results and scientific novelty

In this work the author performed experiment with wireless network which traffic was collected, analyzed and proved to have self-similar properties. Several transient process research methods were analyzed and shown that the only method to research transient processes in a system with self-similar input traffic is simulation. During the research simulation tool was developed to aid in automation of a huge number of simulations. System relaxation times and mean query number in a system with exponential arrivals and self-similar arrivals were analyzed and compared. Using the simulations data it was found that relaxation time of a system with self-similar traffic can be well approximated with Verhulst function. A new method of self-similar traffic generation was proposed.

Possible practical use of results

- System queue size prediction depending on traffic self-similarity degree and system utilization

- System transient mode duration prediction depending on traffic parameters and system utilization
- Networked device development - load planning, device performance parameters planning, memory and queue sizes
- Network planning
- Self-similar network traffic simulation

Theses

- Wireless network traffic was collected, analyzed and proved to have self-similar properties with high Hurst coefficient
- If a queuing system has exponential inter-arrivals and services it's relaxation time increases along with system utilization
- H.Kobayashi model, even modified by the author, cannot be used to model a system with self-similar traffic
- A system with self-similar input traffic relaxation time and mean query number is growing along with self-similarity degree and utilization
- A system with self-similar input traffic system relaxation time can be approximated with Verhulst function
- It is possible to generate self-similar time sequence from Verhulst equation

Results approbation

During the research the results were partly highlighted on the following conferences:

1. Mathematical methods of optimisation of telecommunication networks, Belarussia, Minsk, February 22-24, 2005
2. Electronics and Electrical Engineering, Lithuania, Kaunas, May 16-18, 2005
3. Electronics and Electrical Engineering, Lithuania, Kaunas, May 23-25, 2006
4. Mathematical methods of optimisation of telecommunication networks, Belarussia, Grodno, January 29 - February 1, 2007

5. RTU 48th International Conference, Latvia, Riga, October 11-13, 2007
6. Baltic Congress on Future Internet Communications, Latvia, Riga, February 18-20, 2011

Publications

Research results were highlighted in following publications:

1. S. Ilnickis, E. Petersons, and R. Jerjomins. Server non-stationary behaviour research at near to self-similar query stream influence. *Electronics And Electrical Engineering*, 59(3):46–51, 2005
2. R. Jerjomins and E. Petersons. Client-server model non-stationary behaviour research at near self-similar query stream influence under the condition of overloaded terminal system. *Electronics And Electrical Engineering*, 71(7):35–38, 2006
3. R. Yeryomin and E. Petersons. Self-similar traffic in wireless networks. In *Proc. of International Conference Mathematical Methods of Optimization of Telecommunication Networks.–Minsk: Belarusian State University Press*, volume 18, page 49–55, 2005
4. R. Yeryomin and E. Petersons. Server relaxation time at near to self-similar query stream influence. In *Proc. of International Conference Mathematical Methods of Optimization of Telecommunication Networks.–Minsk: Belarusian State University Press*, volume 19, page 87–91, 2007
5. R. Jerjomins and E. Petersons. Server non-stationary behaviour research in a wireless network under the condition of self-similar traffic. In *Riga Technical University 48th International Scientific Conference, Electronics and Telecommunications*, page 9–15. Riga Technical University, 2007
6. Р. Еремин и Э. Петерсонс. Исследование времени релаксации переходного процесса при обслуживании самоподобного входного трафика в беспроводных сетях. *Автоматика и вычислительная техника*, 43(3):36–46, 2009
7. R. Yeryomin and E. Petersons. Transient process relaxation time research under the condition of self-similar traffic input in wireless networks. *Automatic Control and Computer Sciences*, 43(3):138–147, 2009

8. R. Yeryomin and E. Petersons. Analytical estimation of self-similar wireless traffic relaxation time and hurst coefficient dependence. *Electronics And Electrical Engineering*, 108(2):31–34, 2011
9. R. Yeryomin and E. Petersons. Generating self-similar traffic for wireless network simulation. In *Internet Communications (BCFIC Riga), 2011 Baltic Congress on Future*, page 218–220. IEEE, 2011

Paper size and structure

Thesis consists of an introduction, three chapters, conclusion, six appendices, list of references resulting in 148 pages, 108 figures, 17 tables and 141 used reference.

1

Introduction

1.1 Wireless networks

1.1.1 IEEE 802.11 a/b/g/n networks

The global trend of Internet connection methods is that more and more clients go wireless. And this is not only our PC's and laptops - most of the handhelds and mobile phones produced nowadays are wi-fi enabled. Even most e-book readers have wi-fi chips not to mention the growing tablet market.

The major advantages of wi-fi over GPRS, 3G and 4G for a customer are:

- Connection speed is up to 600 Mbps using 802.11n
- Longer battery life - 3G/4G devices are usually very power hungry comparing to wi-fi
- Low price - a wireless router in retail could be found for as little as fifteen euros not to mention the volume price and manufacturing cost

The last one is an advantage for Internet service providers along with the fact that 802.11 standard devices can be seamlessly integrated into currently winning Ethernet infrastructure. The biggest disadvantage for now is the range and complex authorization comparing to GSM/3G. Users have to enter username and password and sometimes setup security manually. And this is a burden if we are talking about public networks. Nevertheless there is IEEE initiative called 802.11u [3] which was brought to resolve this issue.

Combination of high throughput 802.11n with seamless authorization 802.11u, mesh networking 802.11s, fast BSS transition 802.11r and seamless handover 802.21 would make a very competitive technology at lower price.

1.1.2 Mobile clients

Due to rapid advances in wireless technology, the Internet becomes more mobile. Not only smart phones become more affordable and ubiquitous, also car manufacturers are looking into leveraging Internet connectivity in order to provide advanced applications on car maintenance - such as monitoring and diagnosis, on road assistance - such as providing route navigation, weather maps and automated toll payments, as well as on passenger entertainment, including various types of Internet applications. Most of today's network connected cars still rely on systems with low-bandwidth connectivity (e.g. GSM or satellite links), which don't correspond to needs of emerging new applications. It is expected that such situation will change quickly. Several car manufacturers are offering Internet connectivity for handful models of cars via 3G network and other manufacturers also consider offering Internet-enabled car applications or linking smart phone applications to cars. The current trend suggests that ten millions of cars will go on-line in next several years. This will emerge innovative car-based Internet applications and services, which can have major impact on both manufacturers and passenger experiences.

In the last few years we have witnessed increasing number of cars connected to the Internet. All indicators suggest that this trend will continue and drive-through vehicles will become soon first class citizens on the Internet.

Recent research shows [9] that it is possible to use 802.11n based network for drive-through Internet services at the speed of mobile clients up to 100 kilometers per hour.

The problems we face in usual wired network will become worse in drive-through network as all it's clients will be mobile and constantly switching from one access point to another. Specifically the problem of packet congestions when many clients try to get Internet access at the same time what causes bursts of traffic and packet buffering problems [8].

1.2 Self-similarity and Hurst Parameter

The phenomenon having property of self-similarity, looks equally or equally behaves by its consideration with a different degree of "magnitude" or in different scale. Scaled size can be space (length, width) or time.

1.2.1 Self-similar processes

A self-similar time series has the property that when aggregated, i.e. leading to a shorter time series in which each point is the sum of multiple original points, the new series has the same autocorrelation function as the original.

Let's define a self-similar (fractal) process. Let $X = (X_t, t = 0, 1, 2 \dots N)$ be a covariance stationary (sometimes called wide-sense stationary) stochastic processes of discrete argument – time, with mean M_X, variance (or dispersion) $D_x = \sum_t X_t - M_X$ and autocorrelation function $r(k) = \sum_t (X_t - M_X) \cdot (X_{t+k} - M_X)$ which depends only on k. In particular we assume that X has an autocorrelation function of the form:

$$r(k) \sim k^{-\beta} L_1(k) \qquad , \text{as } k \to \infty \tag{1.1}$$

where $\beta \in (0, 1)$ and L_1 slowly varying at infinity. For simplicity we assume that L_1 is asymptotically constant. For each $m = 1, 2, 3 \dots$ (time scale) let $X^{(m)}$ denote the corresponding aggregated sequence with level of aggregation m, obtained by dividing the original series X into non-overlapping blocks of size m and averaging over each block, that is for each m $X^{(m)}$ is given by

$$X_k^{(m)} = \frac{1}{m}(X_{(k-1)m} + \dots + X_{km-1}) \qquad , k \geq 1 \tag{1.2}$$

Definition: a process X is called (exactly) second-order self-similar with self-similarity parameter $H = 1 - \beta/2$, $H \in (0.5, 1)$ if the corresponding aggregated processes $X_{(m)}$ have the same correlation structure as X, i.e.

$$r^{(m)}(k) = r(k) \qquad , \forall m \tag{1.3}$$

where $r^{(m)}$ denotes the autocorrelation function of $X_{(m)}$. In other words X is exactly self-similar if the aggregated processes $X_{(m)}$ are indistinguishable from X at least with respect to their second order properties.

Definition: a process X is called (asymptotically) second-order self-similar with self-similarity parameter $H = 1 - \beta/2$, $H \in (0.5, 1)$ if for all k large enough

$$r^{(m)}(k) \to r(k) \qquad \text{, as } m \to \infty \tag{1.4}$$

In other words X is asymptotically second-order self-similar if the corresponding aggregated processes $X_{(m)}$ are the same as X or become indistinguishable from X at least with respect to their autocorrelation functions.

1.2.2 Hurst parameter

Hurst parameter is a convenient unit of measurements of self-similarity of stochastic process. This parameter was named by name of H.E.Hurst, devoted the life to studying of Nile and other rivers, and also problems of storage of water.

Hurst has dealt with a problem of designing of the ideal tank for regulation of a stream of Nile on the basis of available record of supervision over a level of the river. The ideal tank should provide the constant stream equal to an average entrance stream which is never overflown and never runs low. For the decision of the given problem it is required to obtain the data on variability of a stream of water. Assuming that the water level is measured once a year we can define the following variables:

X_j - an entrance stream for one year j, $(I \leq j \leq N)$; these are investigated time sequences;

$M(N)$ - the constant annual proceeding stream based on supervision during N years;

L_j - a water level in the tank by the end of year j, $(1 \leq j \leq N)$;

N - number of years of supervision.

These sizes are illustrated with Fig.1.1. Being based on record of N years, we would like to receive values of the minimal and maximal water levels in the tank $L_{min}(N)$, $L_{max}(N)$, and also range $R(N) = L_{max}(N) - L_{min}(N)$.

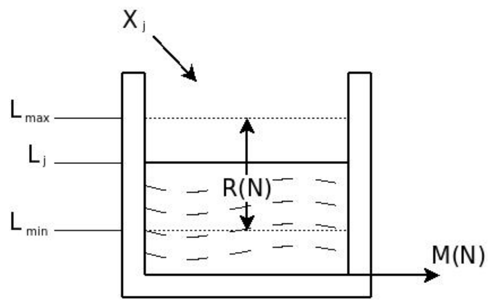


Figure 1.1: An illustration of parameters for Hurst analysis

Taking into account the data on an entrance stream for the period, parameters wanted are easy for calculating:

$$M(N) = \frac{1}{N}\sum_{j=1}^{N} X_j \tag{1.5}$$

$$L_j = \sum_{j=1}^{N} X_j - jM(N) \tag{1.6}$$

$$R(N) = \underset{1\leq j\leq N}{max} L_j - \underset{1\leq j\leq N}{min} L_j \tag{1.7}$$

Thus, $M(N)$ represents an average entrance stream for N years, L_j a total entrance stream for the first j years a minus a total target stream for the same years, $R(N)$ is a difference between maximal and minimal L_j values for these years. Obviously, the range depends on an interval of time N and represents not decreasing function N. Pay attention that parameter R is not equal to a range of time sequence X_j which has the following form:

$$Range(X, N) = \underset{1\leq j\leq N}{max} X_j - \underset{1\leq j\leq N}{min} X_j \tag{1.8}$$

Instead, we can consider L_j as accrued size on which the time sequence deviates average value in time j. Thus, R represents size which, in the certain sense, characterizes variability of casual variable X better.

Hurst investigated a number of the phenomena and has developed the normalized dimensionless size R/S describing variability where S represents selective average:

$$S = \sqrt{\frac{1}{N}\sum_{k=1}^{N}[X_k - M(N)]^2} \tag{1.9}$$

In the certain sense both R, and S measure variability of the data. Size R linearly depends on the data whereas S takes into account squares of reference values. Hurst named this attitude the rescaled range. He has found out, that for many natural phenomena, including the charge of water in the rivers and annual rings of trees, R/S attitude as function N is well described by the following empirical formula for big values of N:

$$R/S \sim (N/2)^H \qquad \text{, when } H > 0.5 \tag{1.10}$$

It is simple to be convinced of it visually if to construct the diagram of R/S dependence from N in logarithmic scale. Hurst has found out, that for many data sets points are laying on a straight line and the inclination of this line represents parameter H from the previous formula.

It is possible to show, that for any short-term process relation R/S becomes asymptotically proportional $N/2$, that is $H = 0.5$. However Hurst has found a set of the phenomena with values H varying in a range from 0.7 up to 0.9. Such big values of H parameter assume the big degree of variability of the data.

The Hurst parameter will be used as a main parameter to define if a processes or time series are self-similar throughout this work.

1.3 Problem statement

Taking into account the said above the following problems may arise in a mobile wireless system:

- System overload, causing:
- High latency and
- Packet loss
- Broken resource management (CPU, memory, channel throughput)

To help solving these problems we should research how the serving nodes (access points) process the requests - how the queue behaves, how long is the relaxation time of the system. Knowing that we have limited resources and the buffer is not unlimited, knowing the current traffic parameters and when we can accept the next client request we could develop MBAC mechanisms for serving node resource control.

Current research is covering the problem of finding system mean query number and system relaxation times under different realistic traffic parameters. But to start the transient mode research we first have to research the wireless traffic parameters and it's nature. The next chapter is devoted to wireless traffic research.

2
Wireless network traffic

2.1 Experiment

Unlike many previous studies analyzing traffic in wired networks (mostly Ethernet) in the given work local wireless intranet traffic of a company was collected and analyzed.

It is shown that the traffic in wireless computer networks is also self-similar and long-range dependent. So hardware manufacturers, software development companies and wireless ISPs should take this into account and should not rely on classical traffic models.

After reviewing all possible methods for getting network traffic statistics it was decided to write own SNMP client to monitor needed routers. For the full description and source code of the tool please refer to the full text of this thesis.

2.2 Experimental data

Fig.2.1 shows the incoming traffic captured from the wireless router during six days in packets per second.

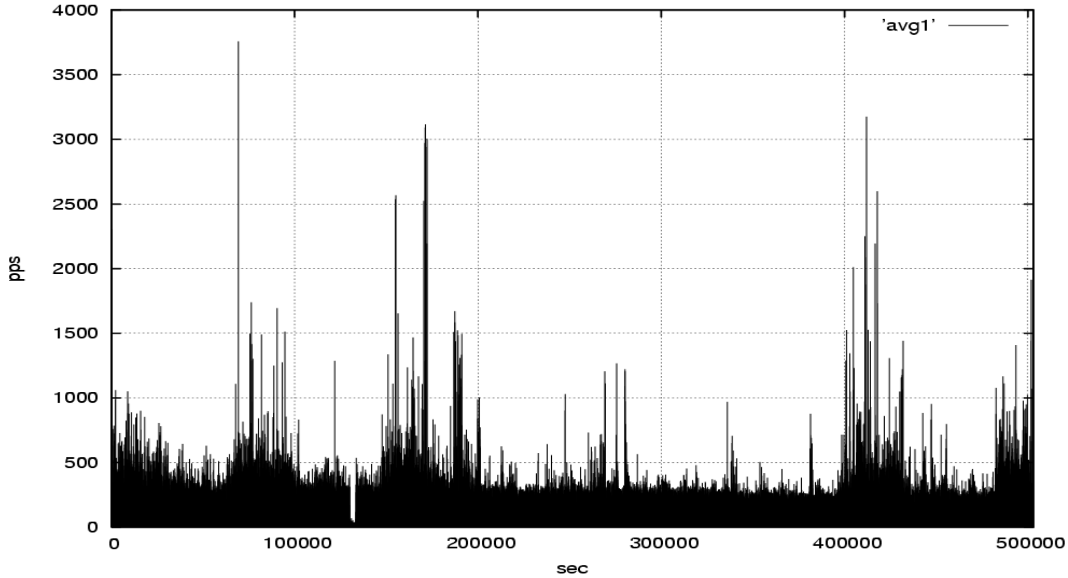


Figure 2.1: Wireless network traffic, six days period

Burstness of the traffic is clearly seen on Figure 2.1 what allows to suggest the the traffic could be self-similar. It's also seen that early in the Friday morning there was no data from the router for a particular amount of time due power line failure. Figures 2.2 through 2.5 show different magnification of the traffic which can be read from the x scale.

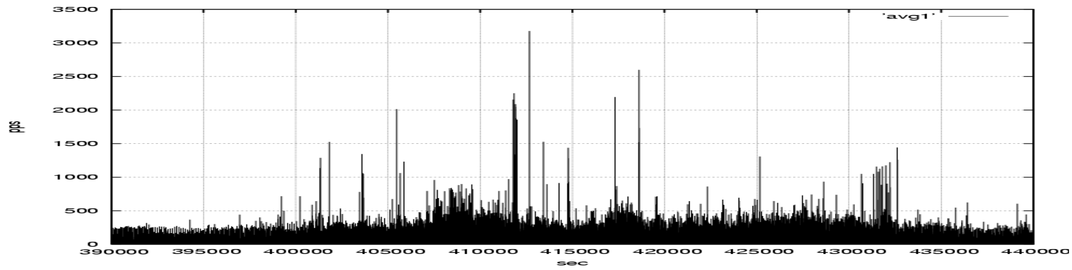


Figure 2.2: Wireless network traffic, one day period

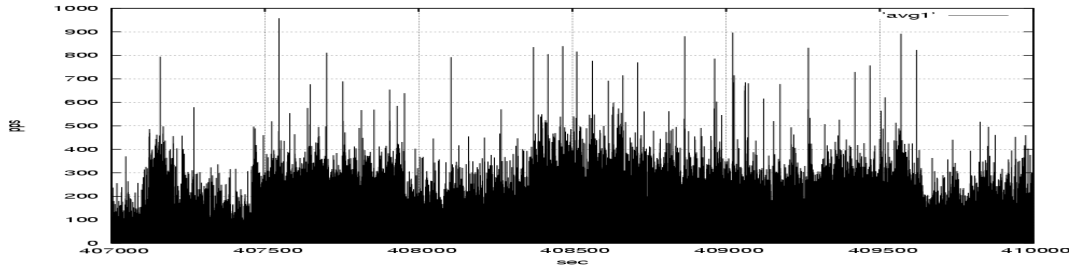


Figure 2.3: Wireless network traffic, one hour period

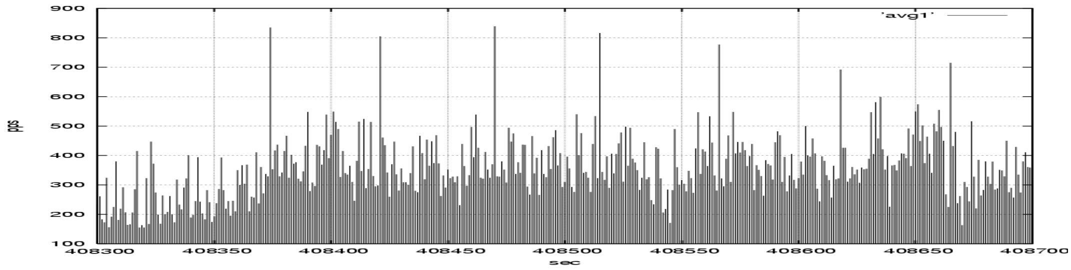


Figure 2.4: Wireless network traffic, 7 minutes period

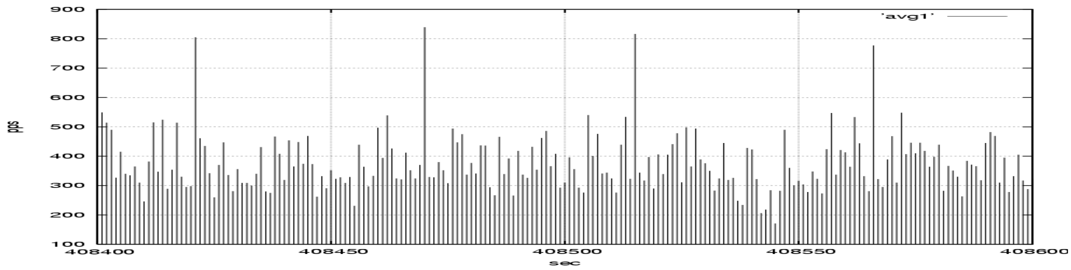


Figure 2.5: Wireless network traffic, 3 minutes period

The above figures show that the measured traffic is visually self-similar at different degree of magnitude. But to claim this we have to perform statistical computations over the dataset. For methods and tools used to perform the analysis please refer to the full text of the thesis.

2.3 Traffic analysis results

In this section we will take a glance on the analysis results of the same dataset as was shown in Section.2.2.

First let's consider aggregated processes. Figures 2.6 through 2.8 show aggregated traffic generated with the program described in Section.2.2 in packets per second. On every figure the scale is multiplied by two, i.e. first figure shows the traffic aggregated by two values, third by four and so on.

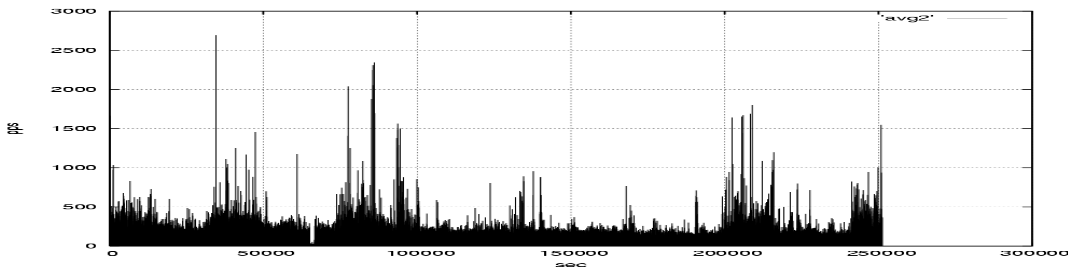


Figure 2.6: Wireless network traffic, aggregation ratio 2

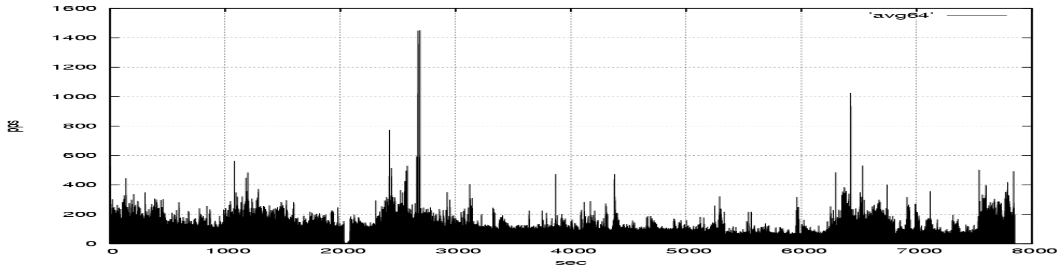


Figure 2.7: Wireless network traffic, aggregation ratio 64

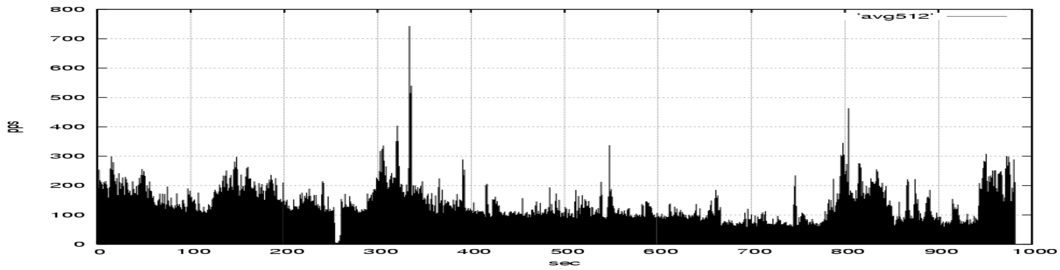
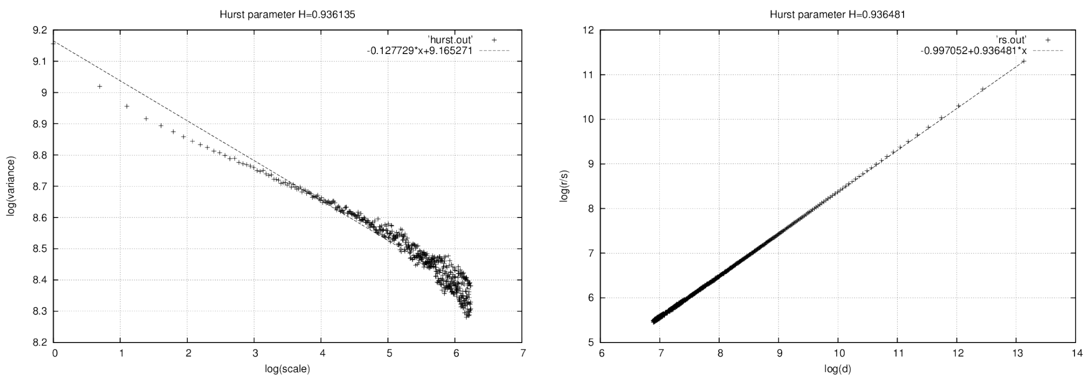


Figure 2.8: Wireless network traffic, aggregation ratio 512

Above plots show that the traffic doesn't lose it's burstness and stays visually almost the same even at high level of aggregation in contrast to non self-similar processes [20].

Now let's look at Hurst parameter estimation on Figure 2.9



(a) Variance-time plot (b) R/S statistics plot

Figure 2.9: Estimated H of wireless network traffic

Both methods show very close result - estimated Hurst parameter $H = 0.936$ indicates that the traffic which was analyzed is highly self-similar.

And last but not least - first 10000 autocorrelation values are shown on Figure 2.10.

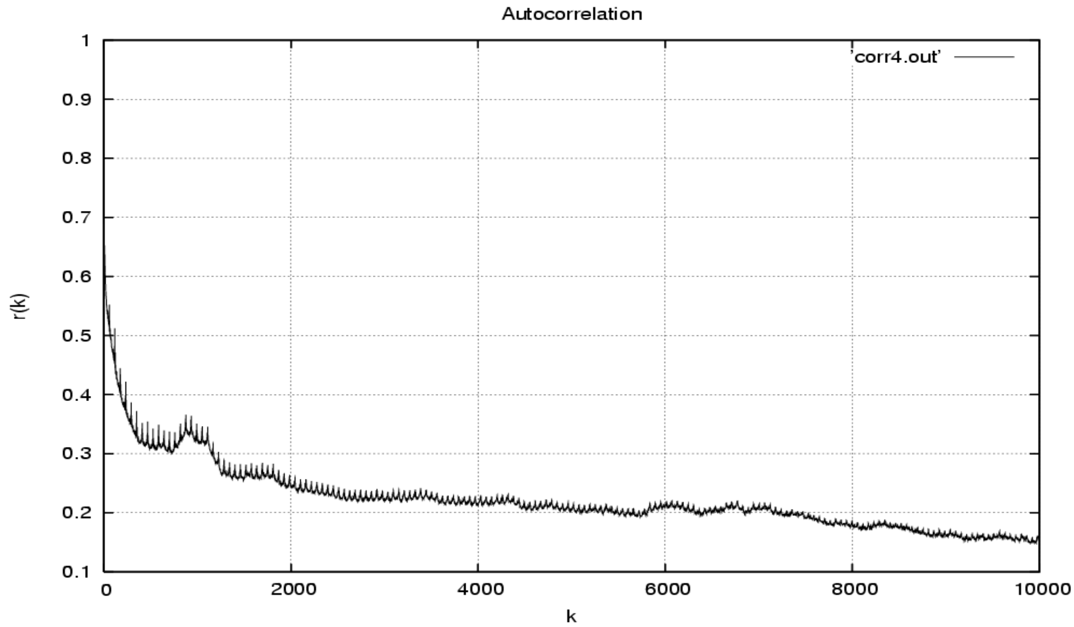


Figure 2.10: Autocorrelation of wireless network traffic, first 10000 values

It is clearly seen on the above figure that autocorrelation is not striving to zero at least for two day traffic. This means that process is long-range dependent and packets taken, for example, from sequence start and packets taken after a day are still statistically related.

2.4 Analysis summary

A number of studies (e.g. [5,17,19]) showed that traffic in wired networks is self-similar. In this chapter we have analyzed big amount of data collected from a wireless router and showed that the traffic in wireless network is also self-similar and long-range dependent. This study leads to the conclusion: traffic models which assume packets arrival process to be Poisson can not be applied for modeling wireless networks in practice. Instead, self-similar traffic model should be used for modeling to reflect the reality.

3

Transient process research methods

Unfortunately there are very few research methods for transient processes - most queuing theories suppose stationary system mode. The only one pure analytical method was developed by A.Kauffman an R.Cruon in [26] and allows to analyze mean query number in system at any arbitrary moment of time for a system with exponentially distributed mean inter-arrival time. Another method is based on diffusion approximation and was developed by H.Kobayashi in [14,15]. Apart from queue size analysis it allows to model more than one client in the system. H.Kobayashi method was modified in order to be able to simulate self-similar query stream. But, as the author shows below, the only reliable method to research transient processes is simulation, especially when dealing with self-similar traffic where pure analytical research methods are not working or the ones which would work is not developed yet.

3.1 Analytical method

By analytical method we mean there is an equation or at least a system of equations which gives a possibility to determine system mean query number at any given moment of time. However a math apparatus is very complicated even for $M/M/1$ systems not to mention systems with non Poisson distributed arrivals. As far as we know there are no analytical methods developed for non Poisson distributed arrivals as of yet and the only one for Poisson distributed arrivals is developed in [26]. It's essence is presented in 3.1.

$$\begin{aligned}\overline{n}(t) =&(\lambda-\mu)t+\mu\int_0^t e^{-(\lambda+\mu)\tau}\cdot\\ &\cdot\left[I_0(2\tau\sqrt{\lambda\mu})+\frac{1}{\sqrt{\Psi}}I_1(2\tau\sqrt{\lambda\mu})+(1-\Psi)\sum_{k=2}^{\infty}\frac{1}{\sqrt{\Psi^k}}I_k(2\tau\sqrt{\lambda\mu})\right]d\tau\end{aligned} \tag{3.1}$$

where λ is the intensity of input, μ is the intensity of service, $\Psi = \lambda/\mu$ is load coefficient and I_x are first kind, order x, modified Bessel functions. Equation 3.1 found in [26] allows us to find mean query number in system at any moment of time. But it can be used only for $M/M/1$ systems were incoming request intensity is Poisson distributed. Nevertheless let's see how relaxation time changes under different utilization factors. Figure 3.1 shows $M/M/1$ system mean query number $\overline{n}(t)$ behavior over time t.

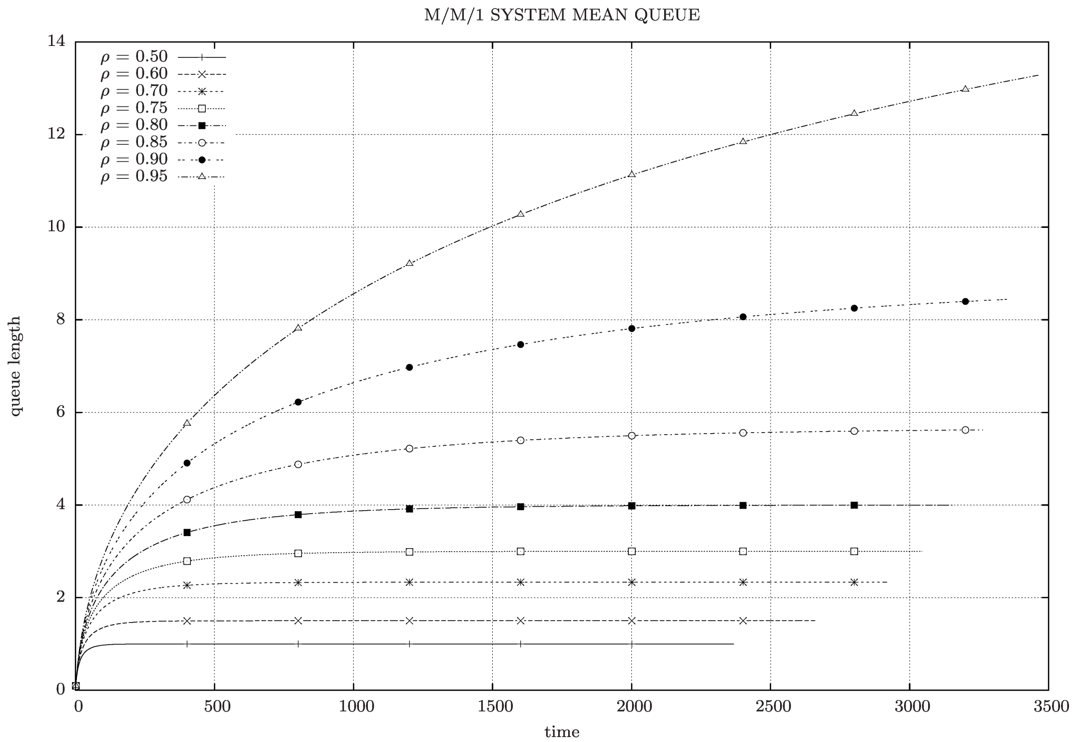


Figure 3.1: $M/M/1$ system mean query number behavior

Figure 3.1 gives very clear view - system relaxation time, i.e. the time needed for a system to enter stationary mode, increases along with utilization

ρ. As the system enters stationary mode and stabilizes (i.e. $t \to \infty$) mean query number $\overline{n}(t)$ obeys 3.2 as per [12,13]

$$\overline{n} = \frac{\rho}{1-\rho} \tag{3.2}$$

These results give us some clue on how relaxation time changes in $M/M/1$ queuing system. However we are more interested in a system with self-similar query stream, e.g. $P/M/1$ with Pareto distributed inter-arrival times. As we've already seen in Chapter 2 it would reflect the reality more closely. Thus we gradually move to the next chapter.

3.2 Diffusion approximation

For system behavior analysis in transitive non-stationary operating mode author decided to try using diffusion approximation method. In this case the model for research is cyclic closed model consisting of terminal system and a network server which common solution has been developed by H.Kobayashi [14,15]. The model has been modified with the purpose of creation a stream of queries to a serving device which is coming nearer to self-similar.

3.2.1 Mathematical model

Communication system model is presented in the form of a server, which has mean query service time equal to μ_1. Terminal subsystem sends queries to server input. Queries processed by server come back to the terminal system which processes them with mean time μ_2. Overall number of queries circulating in the system is equal to N. Thus, we have closed loop system. This model can be considered as an equivalent of server model with limited N size of queries buffer memory, which accepts a stream of queries on its input.

Query stream can be arbitrary, and intervals characteristics between queries are defined only by the first moment μ_1 and dispersion σ_2^2 of returned to the terminal system queries processing.

It's known, that one of the self-similar traffic models is the stream of queries, which arrival time intervals fits power-tail distributions. One of important characteristics of these distributions is dispersion tending to infinity and one of popular laws for such intervals description is Pareto distribution. Probability density of random variable X in this law can be defined by:

$$f(x) = \frac{\alpha}{k}\left(\frac{k}{x}\right)^{\alpha+1} \tag{3.3}$$

Here $x > k$ and $k > 0$. If $\alpha \leq 2$ then $\sigma^2 \to \infty$ but when $\alpha \leq 1$ both mean and dispersion are tending to infinity.

In the work used by the author it is impossible to strictly emulate Pareto law for query intervals distribution description. However, the author of this research proposes to increase the dispersion and consequently also variation coefficient

$$C = \frac{\sigma^2}{\mu^2} \tag{3.4}$$

where μ is mean of query interval times. In research it was assumed, that mean is limited, but dispersion increases, coming nearer to very great value as it takes place to be in self-similar traffic.

In the considered model work it is believed that in terminal system $\sigma_2^2 \to \infty$ and $\mu_2 = const$. The outgoing stream of such node, and, hence, the incoming stream of server, will be close to self-similar. Input of this system is a Poisson stream with mean service time μ_1 and terminal system load coefficient ψ.

Let's imagine a cyclic system where processing times on terminal i are subordinated to distribution law with mean μ_i and variation coefficient C_i, $i = 1, 2$. System is loop-closed, therefore N is total query quantity in the system, $N = const$. Lets define diffusion process which approximates mean query number in system $n_1(t)$ through $x(t)$. Then corresponding diffusion equation will look like this:

$$(\partial/\partial t)\, p(x_0, x; t) = \frac{1}{2}\alpha \left(\partial^2/\partial x^2\right) p(x_0, x; t) - \beta \left(\partial/\partial x\right) p(x_0, x; t) \tag{3.5}$$

Where $\alpha = C_1/\mu_1 + C_2/\mu_2$ and $\beta = 1/\mu_2 - 1/\mu_1$. Solving this equation with boundary conditions $0 \leq x(t) \leq N + 1$ for all $t \geq 0$ use scaling transformation

$$\begin{aligned} y &= \frac{x}{|\alpha/\beta|} = \frac{x}{|(C_1 + C_2\rho)/(1-\rho)|} \\ \tau &= \frac{t}{|\alpha/\beta^2|} = \frac{t}{|\mu_1(C_1 + C_2\rho)/(1-\rho)^2|} \end{aligned} \tag{3.6}$$

Where $\rho = \mu_1/\mu_2$. As a result we have coordinate-free diffusion equation:

$$(\partial/\partial \tau)\, p(y_0, y; \tau) = \frac{1}{2}\left(\partial^2/\partial y^2\right) p(y_0, y; \tau) - \delta \left(\partial/\partial y\right) p(y_0, y; \tau) \tag{3.7}$$

With two reflecting barriers $y = 0$ and $y = b$:

$$\frac{1}{2}\left(\partial/\partial y\right) p(y_0, y; \tau) - \delta p(y_0, y; \tau) = 0 \qquad \text{at } y = 0 \text{ and } y = b \tag{3.8}$$

Where

$$\delta = \begin{cases} 1, & \rho < 1 \\ 0, & \rho = 1 \\ -1, & \rho > 1 \end{cases} \tag{3.9}$$

$$b = \frac{N+1}{|(C_1 + C_2\rho)/(1-\rho)|}$$

Applying the method of "eigenfunction expansion", we obtain the following solution for 3.7:

$$p(y_0, y; \tau) = \begin{cases} 2\delta e^{2\delta y}/(e^{2\delta b} - 1) + exp[\delta(y - y_0 - \delta\tau/2)]\cdot \\ \qquad \cdot \sum_{n=1}^{\infty} \phi_n(y)\phi_n(y_0) exp(-\lambda_n^2 \tau/2), & \text{when } 0 \le y \le b \\ 0, \quad \text{elsewhere} \end{cases} \tag{3.10}$$

Where $\phi_n(y)$ and $\phi_n(y_0)$ are eigenfunctions associated with eigenvalues λ_n:

$$\phi_n(y) = [2\lambda_n^2/b(\lambda_n^2 + 1)]^{\frac{1}{2}} \{\cos \lambda_n y + (\delta/\lambda_n) \sin \lambda_n y\} \tag{3.11}$$

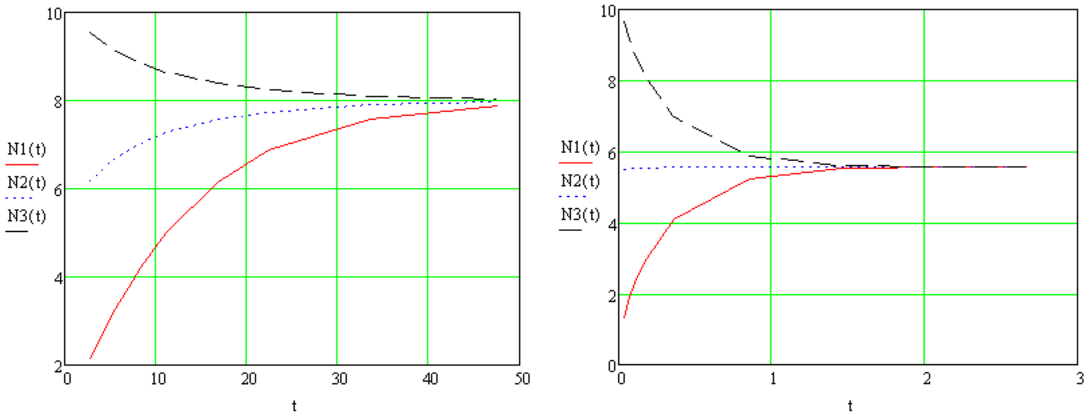
The first term of 3.10 represents the steady-state probability and the second term gives the transient part in terms of eigenfunction expansion. Note that 3.10 satisfies the initial condition $y = y_0$, i.e. $p(y_o, y; \tau) = \delta(y - y_0)$, since the delta function is expressed in terms of the eigenfunctions. The second term of 3.10 is an infinite series, but can be well approximated by finite terms, since the factor $exp(-\frac{1}{2}\lambda_n^2\tau)$ approaches zero as n increases.

3.2.2 Modeling results

By increasing C_2 from one to one hundred we try to simulate self-similar traffic stream. At the same time we also change server utilization factor $\rho = 0.75, 0.95$ to see how the system behaves under different loads. Number of queries in the system N was set to 10 and remained constant.

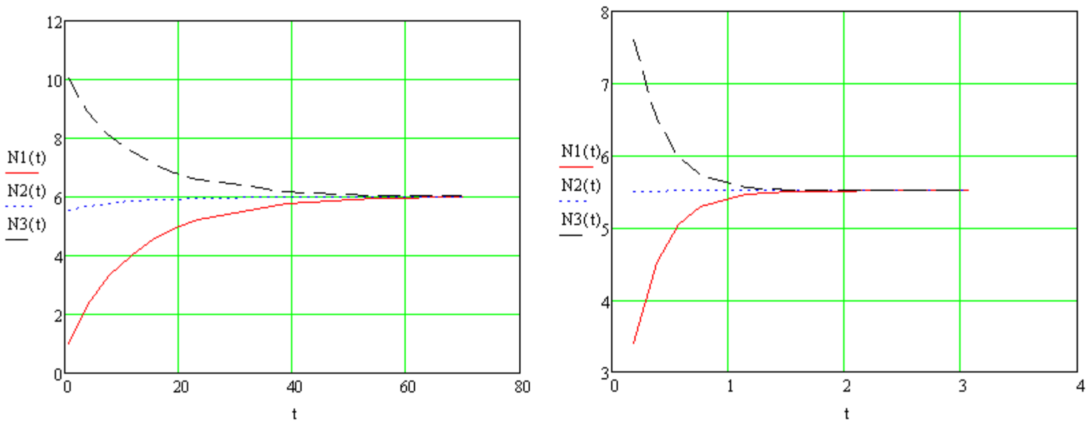
Here we will show only few main figures which will show the tendency of device queue behavior. For complete set of figures please refer to the full text.

Figure.3.2 and Figure.3.3 show the results of using H.Kobayashi modified model with different initial conditions when initial query number n is zero, 5 and 10, which correspond to $N1(t)$, $N2(t)$ and $N3(t)$ curves respectively.



(a) $C_2 = 1$ (b) $C_2 = 100$

Figure 3.2: Query number in system, $\rho = 0.75$



(a) $C_2 = 1$ (b) $C_2 = 100$

Figure 3.3: Query number in system, $\rho = 0.95$

Tables 3.1, 3.2 and 3.3 show the results of performed approximation. Mean query number in system n is listed for stationary mode, when it has finished the transition.

C_2	1	5	50	100
n, $\rho = 0.95$	6.013	5.675	5.521	5.511
n, $\rho = 0.75$	7.969	6.533	5.631	5.566

Table 3.1: System mean query number n

C_2	1	5	50	100
P_{rej}, $\rho = 0.95$	0.116	0.099	0.092	0.091
P_{rej}, $\rho = 0.75$	0.258	0.145	0.097	0.094

Table 3.2: denial of service probability P_{rej}

C_2	1	5	50	100
T_{relax}, $\rho = 0.95$	39.0	18.4	2.328	1.152
T_{relax}, $\rho = 0.75$	47.6	18.24	2.464	1.459

Table 3.3: relaxation times T_{relax}

As one can see from the tables and figures adopted Kobayashi model gives very ambiguous results. On one hand we don't know how serving device mean query number will behave supposing there is self-similar traffic and query number n decrease, while variation C_2 increases, should not frighten us. But on the other hand we know that both the query number and query loss probability should increase along with utilization factor ρ. Modeling results suppose the opposite. This fact made the author stumble at the considered model application for self-similar traffic modeling too. Also system relaxation time was expected to grow while increasing C_2. As further research will show the doubts were justified and H.Kobayashi model, even modified, cannot be used to model a system with self-similar traffic.

3.3 Simulation

As we've seen in previous section diffusion approximation results are not very convincing. The last resort for transient process study is simulation. In this case simulation involves $M/M/1/K$ and $P/M/1/K$ queuing models implementation, self-similar traffic generation for $P/M/1/K$ case and multiple client access methods simulation.

Current research conclusions are fully backed by the simulation results. Therefore we dedicate a separate chapter to consider this method.

4

Transient process simulation

First we have to simulate $M/M/1/K$ system and compare it with analytical model - they should not differ. If $M/M/1/K$ simulation works flawlessly we can proceed to $P/M/1/K$ simulation, believing it gives trustful results for this kind of queuing system also.

Prior to developing author's own simulation tool few others were considered. Among them was Omnet++, NS3 and other full blown simulation suites. All of them are good and trustworthy tools widely used by universities and other research institutions. But at the same time they require skills, and thus time to acquire them, to be able to control the simulator to get needed results.

However one, rather old but also good tool, provides simple enough scripting language for simulation. That's why it was decided to try it out at the first place.

4.1 GPSS

GPSS (General Purpose Simulation System) was one of the very first simulation systems. It was designed for discrete event modeling in 1960s and was very popular. The popularity of GPSS is due, in part, to its power of expression. A short, easily understood GPSS model would require many lines of coding in other programming language to accomplish a similar goal. GPSS user is free to concentrate on the important issues in the model being developed since the language itself collects statistics, produces tabulated results and performs a host of mundane tasks one would prefer not to deal with. Even nowadays GPSS is used by college and university students to explore the basics of simulation world. The author took one of the GPSS implementations [2] which allows free usage for students as it's limits are enough to simulate $M/M/1/K$ and $P/M/1/K$ queuing systems.

First of all let's see how GPSS simulation works out and if it's applicable for both $M/M/1$ and $P/M/1$. Figures 4.1 and 4.2 show utilization coefficient changes in time.

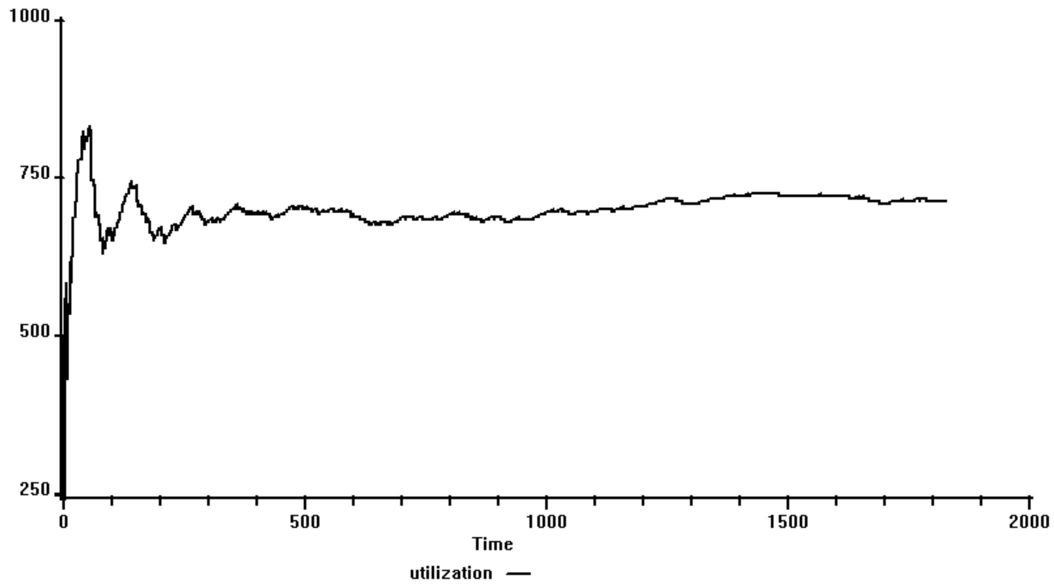


Figure 4.1: $M/M/1$ system utilization, $\lambda = 0.7$, $\mu = 1$

As we can see on Figure 4.1 GPSS gives correct stationary mode utilization result of $\rho \approx 0.7$, which makes us believe that $P/M/1$ simulation can be done using this tool also. We consider more detailed comparison of simulation and analytical expectations in Table 4.1, where $\rho = \lambda/\mu$.

λ	2	1.82	1.58	1.43	1.33	1.17	1.01
ρ, GPSS	0.503	0.554	0.64	0.707	0.757	0.86	0.993
ρ, theory	0.5	0.549	0.633	0.699	0.752	0.855	0.99
error, %	0.6	0.91	1.11	1.14	0.66	0.58	0.30

Table 4.1: GPSS simulation vs. analytical expectations

As one can see from the table above GPSS simulation gives very close results to what one would expect from the theory. But simulation, as seen from the figures already gives one big advantage - it allows to see how system behaves in transitive mode, showing all bursts and anomalies. Now let us see how $P/M/1$ simulation behaves.

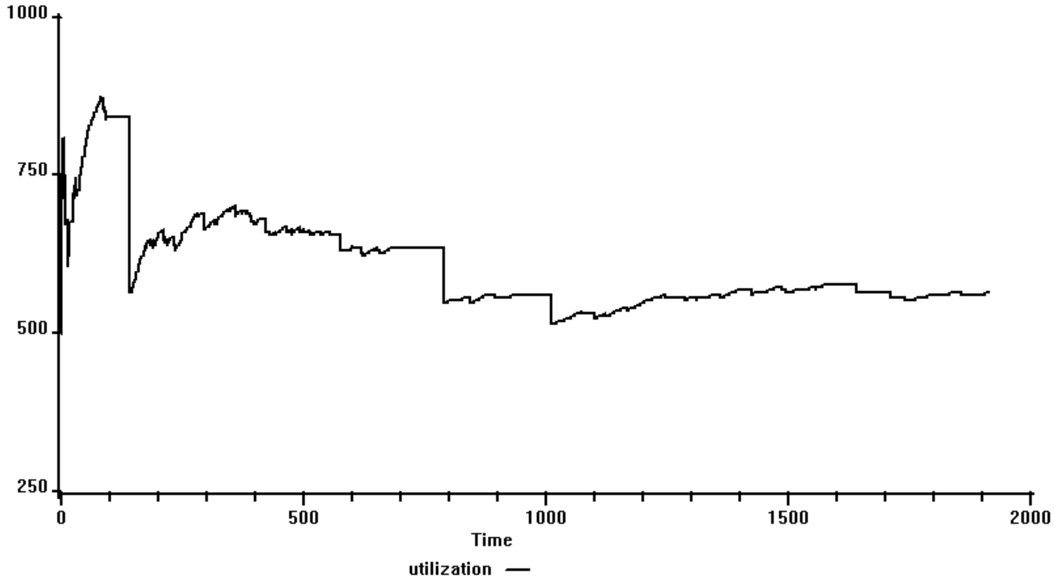


Figure 4.2: $P/M/1$ system utilization, $\lambda = 0.5$, $\mu = 1$

Figure 4.2 shows that even with lower intensity of arrivals $P/M/1$ system utilization behavior is much more complicated, as was expected. This suggests the opposite conclusion from that made by diffusion approximation - relaxation time should grow.

Although GPSS simulation shows impressive results in the sense of very good error rate comparing to the theory and the ability to simulate the system under self-similar traffic load the author stumbled upon the following GPSS tool problems:

- Very poor graphing abilities
- No way to save generated process data for later analysis
- With all respect to developers we don't know how GPSS works internally thus making it impossible to debug the simulation

Despite of the problems described above we have seen that GPSS simulation gives correct results at least for $M/M/1$ system and which completely differ from the diffusion approximation results shown in previous chapter. GPSS problems and good simulation trial results inspired the author to write it's own simulation program which will be described in the next section.

4.2 Transient process simulation

4.2.1 The model

As already noted we are simulating a pair of open queuing systems with one serving device or server. First one has a query stream with exponentially distributed mean inter-arrival times T_a and exponentially distributed mean service time T_s. In the queuing theory such systems are called $M/M/1$. Second system has self-similar query stream. Self-similarity is achieved with the help of Pareto distribution, i.e. mean query inter-arrival times are Pareto distributed. Mean query service time in the second system is also exponentially distributed and equals to one from the first system - T_s. The author called such a system $P/M/1$.

In order to be able to correctly compare $M/M/1$ and $P/M/1$ simulation results we have to equate mathematical expectations of query inter-arrival times thus obtaining 4.1.

$$T_a = \frac{kx_m}{k-1} \tag{4.1}$$

Where k and x_m are Pareto distribution parameters - shape and location respectively, which are obtained using a given Hurst coefficient H and query arrival intensity $\lambda = 1/T_a$ of $M/M/1$ system. From a well known Hurst parameter formula [20] we can easily obtain k:

$$k = 3 - 2H \tag{4.2}$$

Further, using 4.1 and 4.2 we obtain x_m in 4.3 thus obtaining all the needed parameters to simulate a stream of queries with a given Hurst parameter H.

$$x_m = \frac{kT_a - T_a}{k} = T_a - \frac{T_a}{k} = T_a - \frac{T_a}{3-2H} \tag{4.3}$$

Let us denote the importance of equality of query arrival intensity for simulation results comparison. Also 4.3 can be used only when $k > 1$ and thus $0 \leq H < 1$. However we are interested in H coefficient values at which the incoming traffic will be self-similar, i.e. $0.5 < H < 1$ [20].

Below it will be shown that $M/M/1$ system simulation modeling results and the ones obtained by analytical method are very close, what gives us reason to rely on $P/M/1$ simulation too, but now let us consider simulation tool more closely.

4.2.2 Simulation tool

During this study a number of programs were written in C, Perl and Bash programming languages. The basic idea is approximately the same as in GPSS World - the simulation time is advanced by steps, generated arrival and service time lengths T_a and T_s respectively. Depending on current simulation time either the query is added to the queue or is deleted from it. Thus giving us the data about every change in simulated system. See Figure 4.3 for timing example. As we don't have a synchronized simulation clock there is a problem of getting current mean query number in system which is solved by calculating the area under the queue "curve" and dividing it by elapsed time on every step. Buffer size limits maximum queue size - if the queue size is bigger than the given buffer size then the arrived query is discarded.

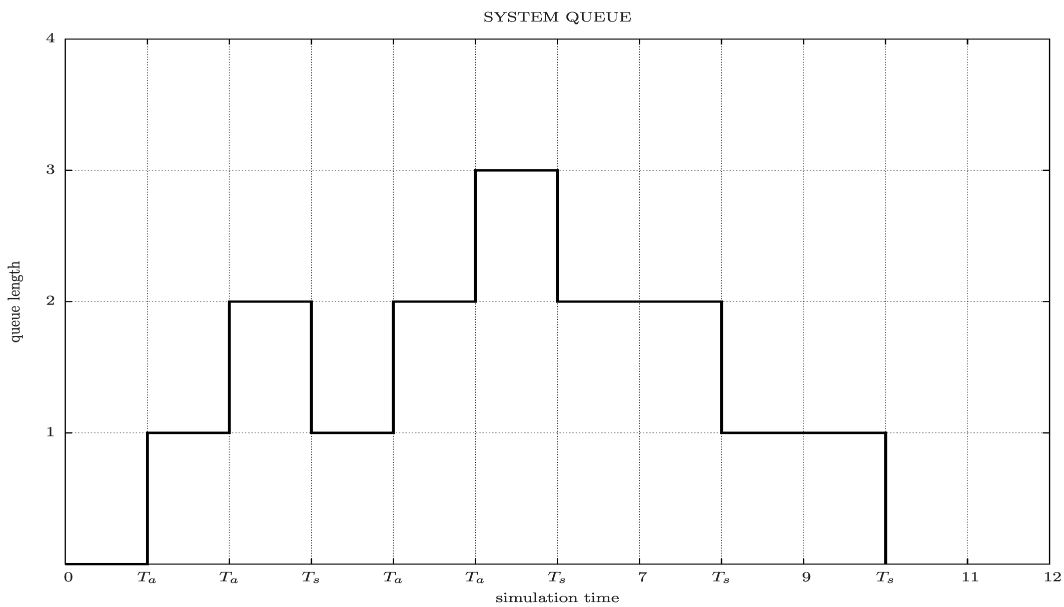


Figure 4.3: Simulated system timing example

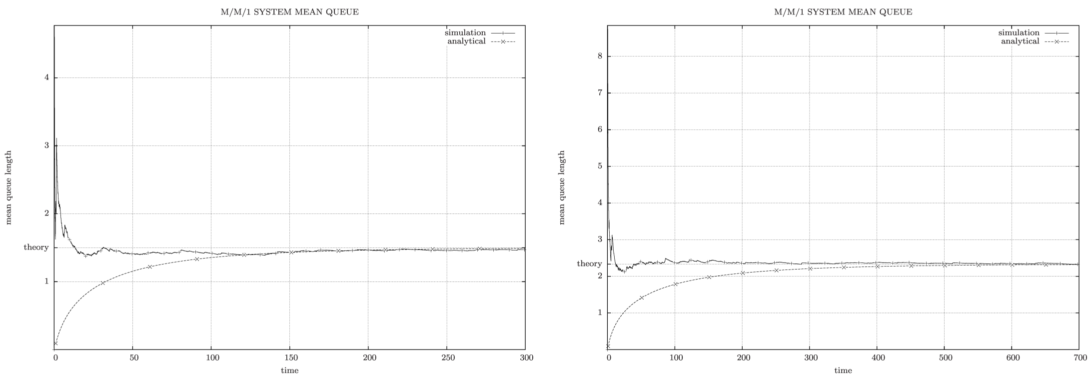
For the the full description of developed tool please refer to the full text of the thesis.

4.3 Simulation modeling results

4.3.1 Simulation vs. analytics

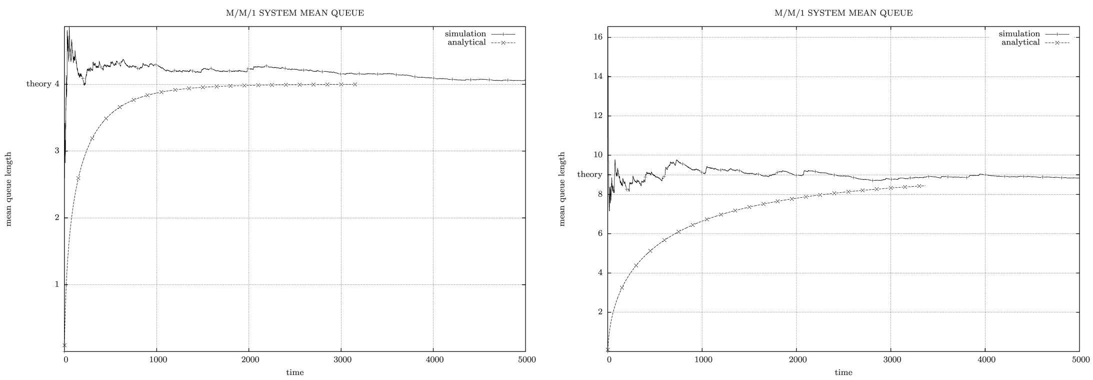
To make sure that our simulation tool works correctly let's first see how the results differ from those obtained by using analytical method and how close mean query number in system in stationary mode is to the theoretical $\overline{n}$ =

$\rho/(1-\rho)$. Figures 4.4 and 4.5 reflect the results of this test simulation.



(a) $M/M/1$ simulation, $\rho = 0.6$ (b) $M/M/1$ simulation, $\rho = 0.7$

Figure 4.4: $M/M/1$ simulation vs. analytics



(a) $M/M/1$ simulation, $\rho = 0.8$ (b) $M/M/1$ simulation, $\rho = 0.9$

Figure 4.5: $M/M/1$ simulation vs. analytics

By eye it seems that simulated resulting mean query number is very close to the theory and relaxation times are close to the analytical method. But let's look at the numbers. Table 4.2 summarizes the results from above figures.

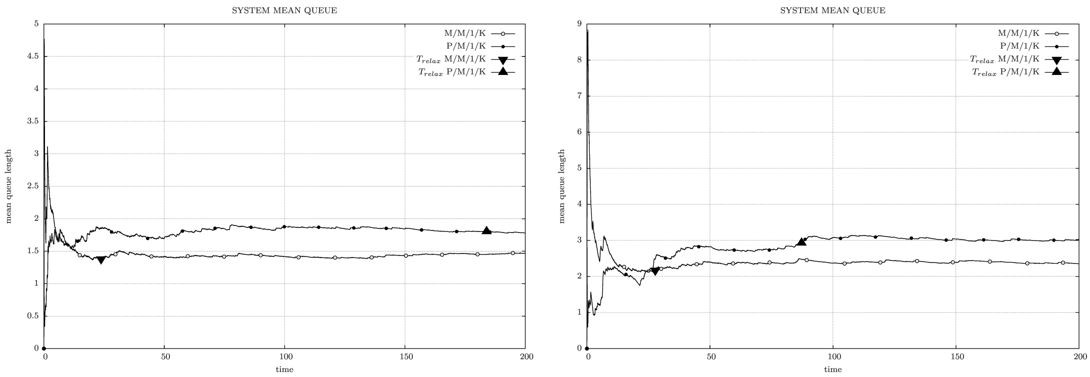
ρ	0.60	0.70	0.75	0.80	0.85	0.90	0.95
$\overline{n}$, theory	1.5	2.333	3	4	5.667	9	19
$\overline{n}$, analytical	1.5	2.333	2.999	3.998	5.623	8.447	13.282
$\overline{n}$, simulation	1.487	2.340	2.962	4.061	5.585	8.898	18.921

Table 4.2: Simulation vs. queuing theory

Although analytical results are exactly the same as expected in theory on lower loads, they quite differ in last two columns. This is due to the calculating errors - Octave functions could not process further and returned error, so the data was cut. On the other hand we see that our simulation tool gives very good output with low error, comparing to the expected theoretical numbers. Sure enough that our simulation tool is working we leave the theoretical expectations and proceed to comparing $M/M/1/K$ and $P/M/1/K$ simulations.

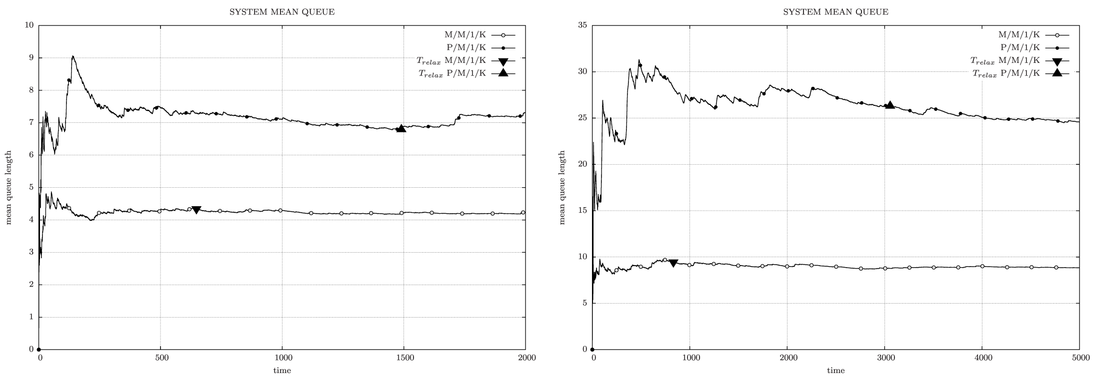
4.3.2 $M/M/1/K$ vs. $P/M/1/K$

Here we already are much closer to our main interest - transient processes in queuing systems and, especially, in the systems with self-similar incoming query stream (in our specific case it will be $P/M/1/K$). Figures 4.6 and 4.7 reflect the results of this comparison of $M/M/1/K$ and $P/M/1/K$ simulations.



(a) $\rho = 0.6$ (b) $\rho = 0.7$

Figure 4.6: $M/M/1/K$ vs. $P/M/1/K$, $H = 0.7$



(a) $\rho = 0.8$ (b) $\rho = 0.9$

Figure 4.7: $M/M/1/K$ vs. $P/M/1/K$, $H = 0.7$

Figures above show how $M/M/1/K$ and $P/M/1/K$ systems queue behavior differ from each other under various utilization factor ρ. Although we took quite low self-similarity grade $H = 0.7$ the difference is quite noticeable. Tables 4.3 and 4.4 summarize the comparison.

ρ	0.60	0.70	0.75	0.80	0.85	0.90	0.95
$\overline{n}$, $M/M/1/K$	1.487	2.340	2.962	4.061	5.585	8.898	18.921
$\overline{n}$, $P/M/1/K$	1.685	3.151	4.508	7.299	11.672	24.567	77.703

Table 4.3: $M/M/1/K$ vs. $P/M/1/K$ mean query number

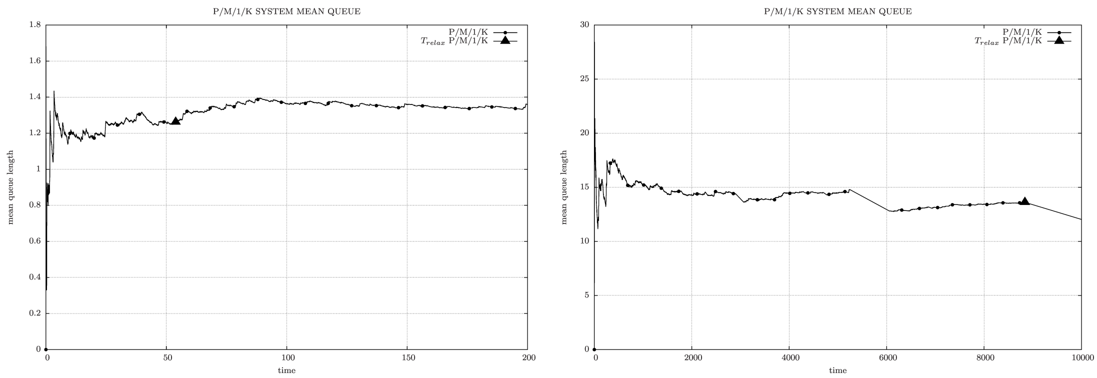
ρ	0.60	0.70	0.75	0.80	0.85	0.90	0.95
T_{relax}, $M/M/1/K$	23.75	27.79	134.27	647.64	265.85	832.06	2928.59
T_{relax}, $P/M/1/K$	183.92	87.26	519.55	1489.79	1471.84	3056.63	4286.55

Table 4.4: $M/M/1/K$ vs. $P/M/1/K$ relaxation times

From the tables and figures above we see that both mean query numbers and relaxation times are much higher for $P/M/1/K$ than for $M/M/1/K$. Here we must note that utilization ρ for Pareto incoming traffic is not quite correct - it differs significantly from $M/M/1/K$ computations, especially when self-similarity parameter H is high. We will consider this fact in the next section. As it was already known, and we have proved it with current simulation results, $M/M/1/K$ system is legging far behind in the sense of reflecting a system with self-similar incoming traffic, i.e. real wold queuing system. Therefore we switch our attention to simulating $P/M/1/K$ system only.

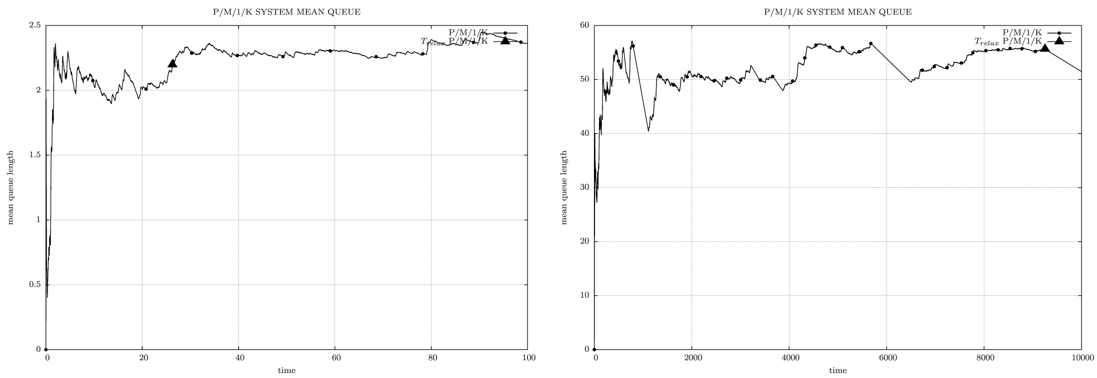
4.3.3 $P/M/1/K$

In this simulation series we change ρ from 0.6 to 0.95 and see what happens if we change self-similarity grade H from 0.6 to 0.9 for each ρ. Here and below we will address to ρ as to a relation of mean service time T_s to mean inter-arrival time T_a, i.e. T_s/T_a or λ/μ, as it would be in a $M/M/1$ system, but the actual utilization factor of $P/M/1$ system, computed during simulation, we will designate as U_p. Figures 4.8 through 4.11 represent this simulation series results.



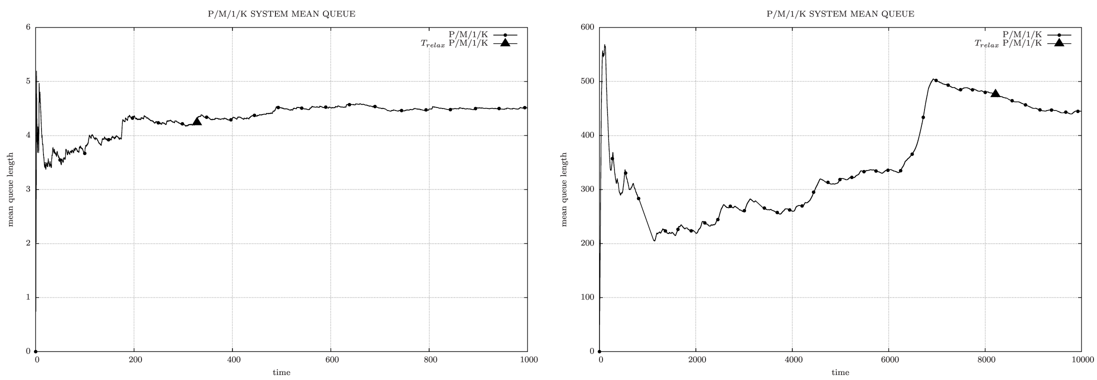
(a) $H = 0.6$ (b) $H = 0.9$

Figure 4.8: $P/M/1/K$ simulation, $\rho = 0.6$



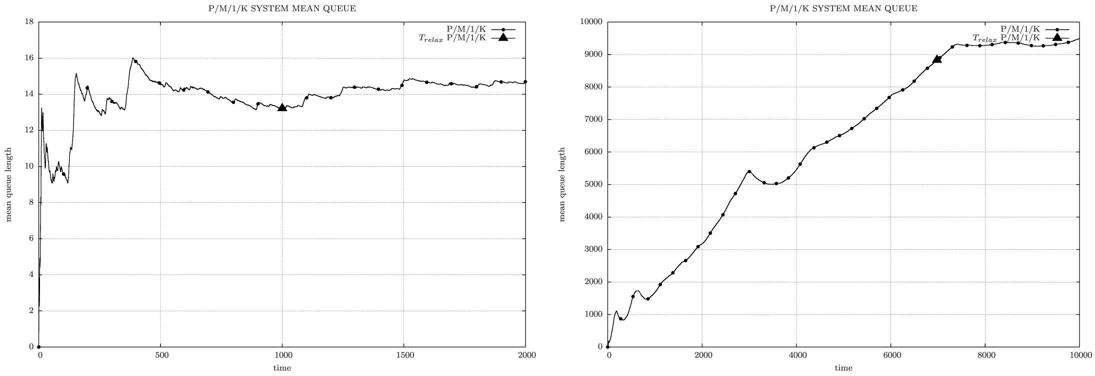
(a) $H = 0.6$ (b) $H = 0.9$

Figure 4.9: $P/M/1/K$ simulation, $\rho = 0.7$



(a) $H = 0.6$ (b) $H = 0.9$

Figure 4.10: $P/M/1/K$ simulation, $\rho = 0.8$



(a) $H = 0.6$ (b) $H = 0.9$

Figure 4.11: $P/M/1/K$ simulation, $\rho = 0.9$

Tables 4.5, 4.6 and 4.7 below summarize the results of these simulation series.

ρ	0.60	0.70	0.75	0.80	0.85	0.90
$H = 0.6$	1.358	2.372	3.194	4.558	7.376	14.203
$H = 0.7$	1.682	3.109	4.526	7.086	11.825	24.598
$H = 0.8$	2.694	5.896	9.493	17.162	33.462	95.155
$H = 0.9$	12.030	51.453	137.776	444.868	1276.517	9489.977

Table 4.5: $P/M/1/K$ mean query number $\overline{n}$ under different parameters

ρ	0.60	0.70	0.75	0.80	0.85	0.90
$H = 0.6$	0.598	0.702	0.750	0.796	0.848	0.901
$H = 0.7$	0.602	0.696	0.754	0.801	0.845	0.885
$H = 0.8$	0.611	0.692	0.742	0.779	0.821	0.868
$H = 0.9$	0.653	0.701	0.694	0.872	0.848	0.992

Table 4.6: $P/M/1/K$ system utilization U_p under different parameters

ρ	0.60	0.70	0.75	0.80	0.85	0.90
$H = 0.6$	53.888	26.272	354.372	327.967	87.822	1000.364
$H = 0.7$	184.353	85.493	436.840	236.371	1480.721	3051.891
$H = 0.8$	113.705	1144.486	1361.760	5586.714	6872.655	8921.949
$H = 0.9$	8838.261	9249.404	8300.975	8215.723	8150.110	6978.514

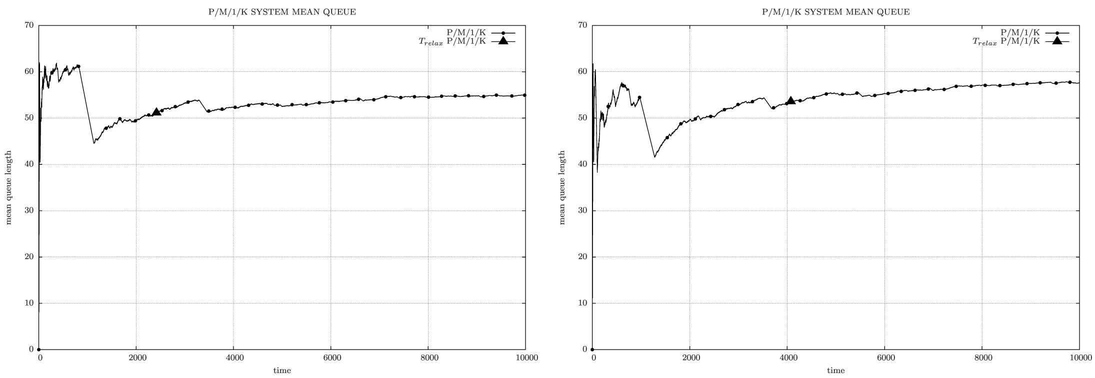
Table 4.7: $P/M/1/K$ system relaxation time T_{relax} under different parameters

As one would notice there are inadequate numbers in last rows and columns. Also after looking at the corresponding figures it's clear that the system is unstable during all simulated time - it accumulates more and more packets in the buffer and is unable to process them. This is quite typical situation and this is why limited buffers are used in all network equipment. Let's try to introduce a limited buffer to our simulation. As an example we will take Linux kernel [4] buffer sizes. Packet queue buffer size varies from driver to driver but in average the buffers are about 100 packets. We also take only $H = 0.8$ and $H = 0.9$ columns for this simulation as most affected.

The results are combined in Tables 4.8 through 4.11 and Figure 4.12 show two most affected systems (for others please see the full text).

ρ	0.60	0.70	0.75	0.80	0.85	0.90
$H = 0.8$	2.694	5.944	9.411	16.375	27.296	42.345
$H = 0.9$	12.573	28.601	36.518	49.335	54.963	57.590

Table 4.8: $P/M/1/K$ mean query number $\overline{n}$, $K = 100$



(a) $\rho = 0.85$ (b) $\rho = 0.9$

Figure 4.12: $P/M/1/K$ simulation, $H = 0.9, K = 100$

ρ	0.60	0.70	0.75	0.80	0.85	0.90
$H = 0.8$	0.611	0.692	0.742	0.779	0.823	0.878
$H = 0.9$	0.576	0.624	0.794	0.782	0.806	0.796

Table 4.9: $P/M/1/K$ system utilization U_p, $K = 100$

ρ	0.60	0.70	0.75	0.80	0.85	0.90
$H = 0.8$	113.705	1183.656	1319.189	1949.432	3484.568	1343.704
$H = 0.9$	7920.085	8470.755	8560.601	4176.040	2417.903	4075.193

Table 4.10: $P/M/1/K$ system relaxation time T_{relax}, $K = 100$

ρ	0.60	0.70	0.75	0.80	0.85	0.90
$H = 0.8$	0	0	0	0.001	0.006	0.024
$H = 0.9$	0.002	0.028	0.063	0.098	0.135	0.176

Table 4.11: $P/M/1/K$ packet loss probability P_{loss}, $K = 100$

Above figures and tables show significant improvement in system relaxation times and mean query numbers (e.g. compare Tables 4.7 and 4.10) but this was a trade-off - up to 17% packet loss appeared in considered systems. It is obvious that increasing packet buffer size will help to lower packet loss probability but it is always a compromise between the memory size and the packet loss probability especially in resource constrained and embedded systems where RAM size is very limited.

Nevertheless increasing the buffer size is not always a good deed. J.Gettys quite recently discovered a problem now called “bufferbloat” [7]. The problem lies in extensive buffering on slow and high latency lines such as DSL. It completely breaks TCP congestion control if a router or switch discards large buffer causing TCP retransmissions and, as a result, much lower overall speed and higher latency. This is one more problem a solution to which could be a dynamic buffer management using the same techniques as MBAC systems [16]. See also [1] for reference on bufferbloat problem.

4.4 Analysis of obtained results

The previous section shows the results of our simulation but that’s not all. The simulation parameters spectrum was much broader and in this section we will try to analyze them.

4.4.1 Relaxation time

As we have stated earlier we are interested in transient process relaxation time T_{relax} which main known parameter is self-similarity degree H. The data we have after conducting about 1200 simulations allows us to construct the graphs of relaxation time T_{relax} dependence on self-similarity parameter H.

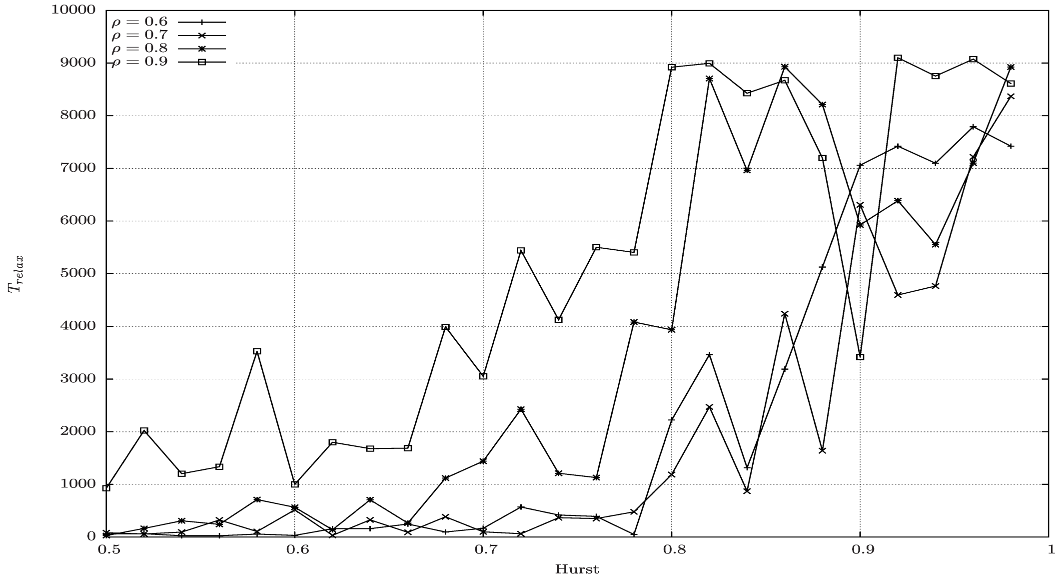


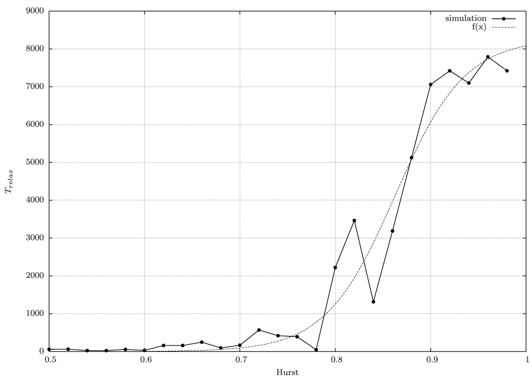
Figure 4.13: T_{relax} dependence on H, different ρ

Figure 4.13 shows how simulated system relaxation time grows along with self-similarity degree. Although having a real world or simulation data is good there is always a certain interest in approximation this data. Having analytical estimation can ease the further use of these results. By analytical estimation we mean finding an equation which will fit best to our simulation results. Such estimation could be called regression or non linear fit. We performed this analysis using Gnuplot software which gives the ability to define any formula for experimental data fitting. As there are countless number of different formulas to try it was decided to stop on several simple well known formulas which would allow fast and easy computation of relaxation time. The chose fell on linear as the most simple, exponential, logarithmic and Verhulst equations. The later was chosen because it was determined by eye that it could probably fit well our simulation data. Goodness of fit was determined by smallest chi square parameter which basically is squared sum of differences between the formula and the data allowing to see how close the formula approximates the data. Comparative results of chi square values we have put in 4.12.

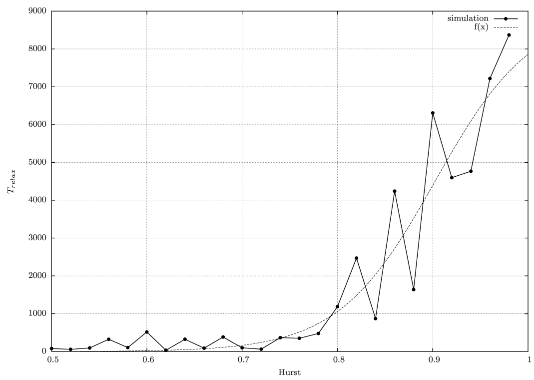
ρ	0.6	0.7	0.8	0.9
Linear, $f(x) = ax + b$	$1.2e+08$	$8.0e+07$	$2.3e+07$	$5.3e+07$
Exponential, $f(x) = a \cdot e^{-bx} + c$	$3.4e+07$	$2.4e+07$	$2.4e+07$	$6.0e+07$
Logarithmic, $f(x) = a + b \cdot ln(x)$	$2.9e+08$	$3.8e+08$	$2.8e+08$	$1.4e+08$
Verhulst, $f(x) = \frac{ab \cdot e^{cx}}{a+b(\exp(cx)-1)}$	$8.7e+06$	$4.7e+06$	$1.9e+07$	$3.3e+07$

Table 4.12: $T_{relax}(H)$ regression errors

From Table 4.12 we see that Verhulst equation gives the closest approximation to out simulation data. Here a, b and c are equation parameters which were estimated. Verhulst curve can adopt very close to our data in most cases what is seen on Figures 4.14 and 4.15 below.

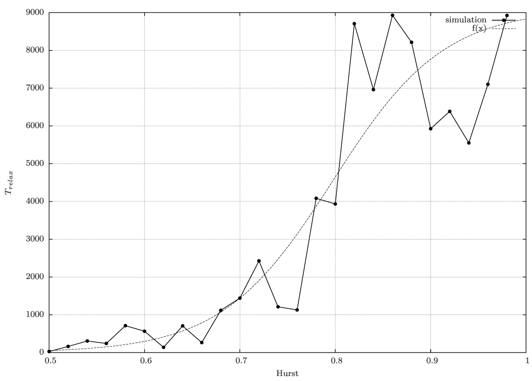


(a) $\rho = 0.6$

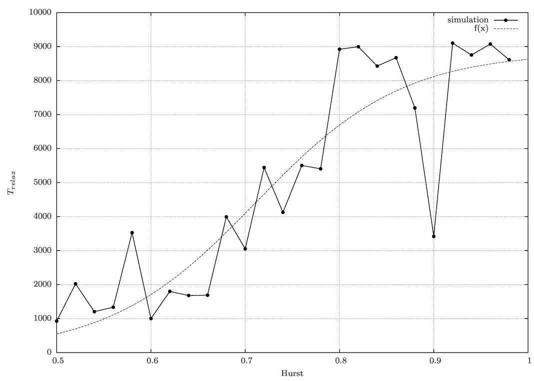


(b) $\rho = 0.7$

Figure 4.14: $T_{relax}(H)$ Verhulst regression



(a) $\rho = 0.8$



(b) $\rho = 0.9$

Figure 4.15: $T_{relax}(H)$ Verhulst regression

Estimated parameters of Verhulst equation for above figures are given in Table 4.13:

ρ	0.6	0.7	0.8	0.9
a	8270.48 +/- 634.6	9015.42 +/- 2311	9105.39 +/- 1176	8838.93 +/- 998.8
b	4.53e-07 +/- 2.16e-06	0.00018 +/- 0.0009	0.0105 +/- 0.0419	0.96 +/- 2.46
c	27.38 +/- 5.65	19.66 +/- 6.19	17.14 +/- 5.229	12.83 +/- 3.80

Table 4.13: $T_{relax}(H)$ regression estimated parameters

Although Table 4.13 gives us estimated parameters for the fitting formula these are not a common solution to the problem. Finding common solution would involve solving a system of three differential equations for each ρ under the test and then finding relations and correlations between a, b and c of each ρ. Solving this task would probably result in doctoral thesis in mathematics and is actually out of scope of this work. Nevertheless one could use the tables of already calculated parameters for practical purposes.

4.4.2 Self-similar traffic generation

While the above analysis results could be used in an MBAC system the author have found another very interesting property of Verhulst equation which could be used for traffic generation.

Let us go a little back. Verhulst equation 4.6 we used earlier (also called logistic function) is a common solution to differential equation 4.4 used by P-F.Verhulst himself for population growth modeling and, later, by R.May and M.Feigenbaum.

$$\frac{dP}{dt} = rP\left(1 - \frac{P}{K}\right) \tag{4.4}$$

The where P is population size, r is growth rate, K is capacity and t is time. Dividing both sides of the equation by K and setting $x = P/K$ gives the differential equation:

$$\frac{dx}{dt} = rx(1-x) \tag{4.5}$$

Common solution to this equation with P_0 being the initial population is:

$$P(t) = \frac{KP_0e^{rt}}{K + P_0(e^{rt} - 1)} \tag{4.6}$$

There is also a discrete representation of 4.5 called logistic map:

$$x_{n+1} = rx_n(1 - x_n) \tag{4.7}$$

R.May [18] and, later, M.Feigenbaum [6] have studied 4.7 in their work and discovered such interesting property as bifurcation. Plotting many generations of x against r gives us bifurcation diagram. This diagram is also called Feigenbaum fractal (or tree) because each next bifurcation point graphically looks like previous only in smaller scale and plotted resembles a tree.

So, it appears that a system, serving self-similar (or it also could be called fractal) traffic, relaxation time can be well approximated by logistic curve on one hand and the same differential equation produces recurrent relation, which produces one of the most well known fractal, on the other hand. Thus the author expected that some useful results could be obtained using 4.7 for self-similar time sequence generation, which could be used for traffic generation and, further, for wireless network systems relaxation time modeling.

First let's see if fractal orbits can give some self-similarity. By orbit author mean recording at least 100 thousands of x value iterations with fixed r and given x_0 as initial condition. In [6, 18] it was stated that x destabilizes starting from $r = 3.57$. So, we took $r = 3.50$ as first orbit and proceeded till $r = 3.99$ with step 0.01, thus making 50 tests. Unfortunately none of the test gave self-similar sequence regardless of initial value of $0.1 < x_0 < 0.5$. The highest achieved Hurst parameter was around $H = 0.53$ at $r = 3.96$, while for self-similar sequence it should be $0.5 < H < 1$ and as we have already seen in Chapter 2 for real wireless network traffic H is higher than 0.9.

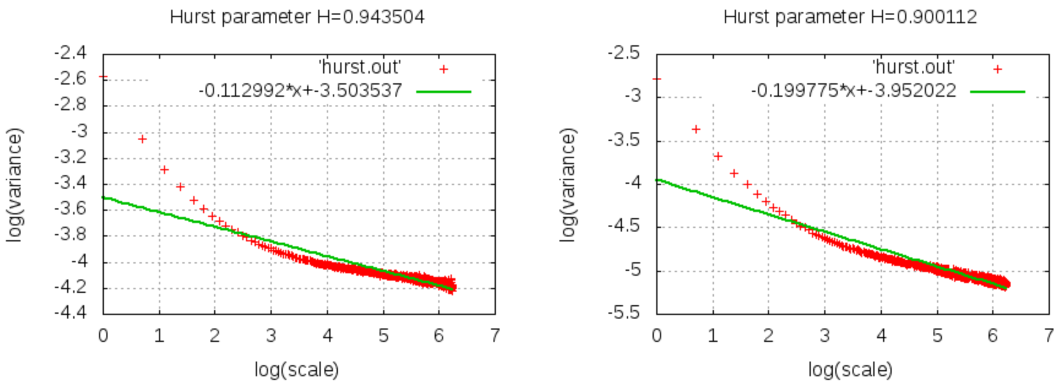
Nevertheless the author tried to use 4.7 in another way - track a given iteration of x while changing r. This method gave very interesting results. The author have performed a set of 16575 tests with dynamic diapason of r starting from $3.50 < r < 3.88$ and ending at $r = 4$ and tracking from 15th to 100th iteration of x, thus conducting 3315 tests for each given $0.1 < x_0 < 0.5$. An overview of results we have put in Table 4.14. which represents a number of obtained self-similar sequences and obtained Hurst parameter values where both variance-time and R/S test methods gave similar results.

Hurst parameter H	Initial x_0				
	0.1	0.2	0.3	0.4	0.5
0.5 - 0.59	0	2	1	1	0
0.6 - 0.69	134	106	179	142	21
0.7 - 0.79	433	374	743	427	316
0.8 - 0.89	1568	1099	1092	1340	586
0.9 - 0.99	904	1319	776	1090	2206

Table 4.14: Generator test overview

From Table 4.14 is clearly seen that more than 70% from overall tests produce highly self-similar sequence and more than 50% from those have Hurst coefficient $H > 0.9$. Almost non of the x iteration tracing method tests gave Hurst parameter $H < 0.6$ what author considers as a very good result.

Figure 4.16 show variance-time plots for estimated Hurst parameter for tracked 90th iteration of x and r starting from 3.6 with initial conditions $x_0 = 0.1$ and $x_0 = 0.5$ correspondingly.



(a) $x_0 = 0.1$, $H = 0.94$ (b) $x_0 = 0.5$, $H = 0.90$

Figure 4.16: Estimated H for 90th iteration of x, $r = 3.6$

It seems natural to the author to generate self-similarity from something that is already self-similar, or fractal, in it's base. Though it's still unclear how to control Hurst parameter to get predicted results directly from using 4.7 but at least already tested parameters can be used to generate self-similar traffic and use it for wireless and other network modeling.

5
Conclusions

5.1 Research summary and contributions

During the work defined objectives were achieved and the following points summarize author's research results:

- Wireless network traffic was collected, analyzed and proved to have self-similar properties with high Hurst coefficient H
- $M/M/1/K$ system relaxation time increases along with system utilization
- H.Kobayashi model, even modified by the author, cannot be used to model a system with self-similar traffic
- $P/M/1/K$ system relaxation time T_{relax} and mean query number $\overline{n}$ is growing along with self-similarity degree H and utilization ρ
- $P/M/1/K$ system relaxation time T_{relax} can be approximated with Verhulst function
- It is possible to generate self-similar time sequence from Verhulst equation

Every point was discussed in international conferences and highlighted in author's nine publications [8, 10, 11, 21–25, 27]

5.2 Possible practical use

Practical use of the current work results could involve:

- System queue size prediction depending on traffic self-similarity degree and system utilization
- System transient mode duration prediction depending on traffic parameters and system utilization
- Networked device development - load planning, device performance parameters planning, memory and queue sizes
- Network planning
- Self-similar network traffic simulation

References

[1] The bufferbloat projects. `http://www.bufferbloat.net/`.

[2] Gpss world. `http://www.minutemansoftware.com/simulation.htm`.

[3] Ieee 802.11u. `http://standards.ieee.org/findstds/standard/802.11u-2011.html`.

[4] Linux kernel archives. `http://kernel.org/`.

[5] M.E. Crovella and A. Bestavros. Explaining world wide web traffic self-similarity. Technical report, Boston University Computer Science Department, 1995.

[6] M.J. Feigenbaum. Quantitative universality for a class of nonlinear transformations. *Journal of statistical physics*, 19(1):25–52, 1978.

[7] Jim Gettys. Bufferbloat: Dark buffers in the internet. *IEEE Internet Computing*, 15:96, 95, 2011.

[8] S. Ilnickis, E. Petersons, and R. Jerjomins. Server non-stationary behaviour research at near to self-similar query stream influence. *Electronics And Electrical Engineering*, 59(3):46–51, 2005.

[9] A. Ipatovs. *Experimental and Analytical Goodput Evaluation of Drive-thru Internet Systems*. PhD thesis, Riga Technical University, 2012.

[10] R. Jerjomins and E. Petersons. Client-server model non-stationary behaviour research at near self-similar query stream influence under the condition of overloaded terminal system. *Electronics And Electrical Engineering*, 71(7):35–38, 2006.

[11] R. Jerjomins and E. Petersons. Server non-stationary behaviour research in a wireless network under the condition of self-similar traffic. In *Riga Technical University 48th International Scientific Conference, Electronics and Telecommunications*, page 9–15. Riga Technical University, 2007.

[12] L. Kleinrock. *Queueing systems: Computer applications*. Number vol. 2 in Wiley-Interscience Publication. Wiley, 1976.

[13] L. Kleinrock. *Queueing Systems: Theory*. Number vol. 1 in A Wiley-Interscience publication. Wiley, 1976.

[14] Hisashi Kobayashi. Application of the diffusion approximation to queueing networks i: Equilibrium queue distributions. *J. ACM*, 21:316–328, April 1974.

[15] Hisashi Kobayashi. Application of the diffusion approximation to queueing networks ii: Nonequilibrium distributions and applications to computer modeling. *J. ACM*, 21:459–469, July 1974.

[16] M. Kulikovs. *Research and development of effective managing algorithms in admission control systems for telecommunication networks*. PhD thesis, Riga Technical University, 2010.

[17] W.E. Leland, M.S. Taqqu, W. Willinger, and D.V. Wilson. On the self-similar nature of ethernet traffic. *ACM SIGCOMM Computer Communication Review*, 23(4):183–193, 1993.

[18] R.M. May et al. Simple mathematical models with very complicated dynamics. *Nature*, 261(5560):459–467, 1976.

[19] O. Rose. Estimation of the hurst parameter of long-range dependent time series. *University of Wurzburg, Institute of Computer Science Research Report Series.—February*, 1996.

[20] W. Stallings. *High-speed networks and internets: performance and quality of service*. William Stallings books on computer and data communications technology. Prentice Hall, 2002.

[21] R. Yeryomin and E. Petersons. Self-similar traffic in wireless networks. In *Proc. of International Conference Mathematical Methods of Optimization of Telecommunication Networks.–Minsk: Belarusian State University Press*, volume 18, page 49–55, 2005.

[22] R. Yeryomin and E. Petersons. Server relaxation time at near to self-similar query stream influence. In *Proc. of International Conference Mathematical Methods of Optimization of Telecommunication Networks.–Minsk: Belarusian State University Press*, volume 19, page 87–91, 2007.

[23] R. Yeryomin and E. Petersons. Transient process relaxation time research under the condition of self-similar traffic input in wireless networks. *Automatic Control and Computer Sciences*, 43(3):138–147, 2009.

[24] R. Yeryomin and E. Petersons. Analytical estimation of self-similar wireless traffic relaxation time and hurst coefficient dependence. *Electronics And Electrical Engineering*, 108(2):31–34, 2011.

[25] R. Yeryomin and E. Petersons. Generating self-similar traffic for wireless network simulation. In *Internet Communications (BCFIC Riga), 2011 Baltic Congress on Future*, page 218–220. IEEE, 2011.

[26] А. Кофман и Р. Крюон. *Массовое обслуживание. Теория и приложения*. Мир, 1965.

[27] Р. Еремин и Э. Петерсонс. Исследование времени релаксации переходного процесса при обслуживании самоподобного входного трафика в беспроводных сетях. *Автоматика и вычислительная техника*, 43(3):36–46, 2009.