# PACKET LOSS PROBABILITY ESTIMATION USING ERLANG B AND M/G/1/K MODELS IN MODERN VOIP NETWORKS

Tibor MIŠUTH[1], Ivan BAROŇÁK[1]

[1]Institute of Telecommunications, Faculty of Electrical Engineering and Information Technology,
Slovak University of Technology in Bratislava, Ilkovičova 3, 812 19 Bratislava
tibor.misuth@stuba.sk, ivan.baronak@stuba.sk

***Abstract:*** *The paper presents an approach to packet loss probability estimation in modern VoIP networks. Our proposal is based on statistical characteristics of telecommunication networks and VoIp traffic packet flows. We adapted simple Erlang B model that is widely used for classic telecommunication networks dimensioning and analyze its potential applicability to modern convergent IP networks. Furthermore we derived M/G/1/K model of aggregated data flow to study influence of queue size on packet loss probability. At last we verified the model outcome and applicability using extensive simulations using NS2 software.*
***Keywords:*** *Erlang B, Packet Loss, Quality of Service, Voice over IP*.

## 1. Introduction

End of 19th and beginning of 20th century was the period of rapid growth of popularity and fast evolution of telecommunications. Together with constantly rising number of subscribers connected to the networks and therefore increase in number of executed calls also capacity requirements rose proportionally. Since construction costs, particularly of long distance lines, were quite expensive, wrong estimation of expected traffic and overdimensioning of capacity could lead to economic problems. From the other point of view underdimensioning meant loss of potential profit and degradation of customers' satisfaction level as well. Therefore analysis of dependencies of telecommunication traffic and their description became important research field.

Danish mathematician A. K. Erlang focused exactly on problematic of traffic load and its relationship to available capacity of trunk lines. The result of his effort are mainly two models for call loss or call waiting probabilities calculations also known as Erlang B and Erlang C models or 1st and 2nd Erlangs' equations [1].

These equations put together offered traffic load, number of available telecommunication lines (or handling servers from the queuing systems perspective) and probability of call not being processed immediately on its arrival. Especially the first Erlang's equation covers the problem of dimensioning of sufficient trunk lines capacity.

Character of data communication networks has changed significantly in last few years. A decade ago, most part of the traffic in such networks consisted of data belonging to file transfers between network peers and HTTP related traffic generated by users accessing web pages. These services can be recognized as non real-time and amount of traffic was rather low compared to present values. The real-time services utilized that time were mainly remote terminal access and Instant Messaging services, where amounts of data required for communication were even lower. Since majority of the traffic had non real-time character, influence of negative effects like latency and jitter on results was not so critical. Negative effects of packet loss and reordered delivery were eliminated by reliable transport protocols functions. Furthermore network infrastructure together with computers' computational power did not provide sufficient background for any wideband service [2].

This picture of steady state slowly living network has become completely obsolete nowadays. Massive growth of communication networks' capacities, throughput and dramatic increase of computational power of computers and network elements brought series of new services to light. Nowadays majority of data traffic consists of high speed multimedia streams and communication. Large amount of audio and or video traffic is exchanged between network users and servers together with content of web pages full of embedded audio and video content Since most of the services utilized today are real-time, their quality and finally level of customers' satisfaction depends on quality and parameters of the network in use [3].

The term Quality of Service (QoS) has become very frequent and popular. Negative effects like jitter and delay cannot be overlooked anymore since they play significant role in real-time communication quality. Unreliable but very simple transport protocols like UDP do not provide any method for packet loss resolution since retransmission is not an option in real-time. These negative effects therefore has to be studied, understood and addressed by utilization of right QoS methods [3].

**Received on: 29.02.2012**
**Accepted on: 06.07.2012**

Revolution in data networks caused significant changes to the world of telecommunications. Telecommunication systems were being transformed from original separate circuit switched systems to packet switched systems. One channel is used for various types of traffic and voice information was started to being transferred in form of samples or packets. The most widely deployed type of data networks are networks based on IP protocol. That's why voice information transferred in form of samples is also noted as Voice over Internet Protocol (VoIP) [4,5].

Even though capacity and construction costs of today's packet based networks are considerably different than parameters from early telecommunication era, the problematic of proper capacity estimation is still very important. This paper proposes a possible way of simple application of original Erlangs' ideas adapted to modern convergent telecommunications environment and analysis the influence of waiting queue on results using more complicated M/G/1/K queuing model. As majority of ITU codecs standardized for VoIP (G.7xx codec series) [6] works in constant bit-rate mode, we limit our study to this area.

Next sections of this paper are structured as follows. First some general principles and statistical characteristics of telecommunication networks are described. Then original Erlang B model is presented. Then some properties of VoIP networks are analyzed in comparison to classic telecommunication networks. Next section deals with adaptation of original Erlang's model to packet switched environment. Afterwards we derive characteristics of aggregated data flow to obtain an M/G/1/K model of network element in VoIP network. Finally the results of simulation with varying input parameters and their comparison to models estimations are presented.

## 2. General principles

This section deals with basic properties of telecommunication traffic that are valid since very beginning of telecommunications era back at the end of 19th century. Most of these assumptions were used by A. K. Erlang to formulate his famous Erlang equations [7]-[8].These models require following properties of processes to be guaranteed [7]:

- population of sources (requests generators) is much greater than the number of servers (agents),
- each source generates its requests randomly in time and independently from each other,
- cumulative average number of requests per time unit from all sources is invariant (constant) in time,
- serving time of each request is random variable with exponential distribution.

These rules then implicate basic characteristics of requests arrival process as well as their serving process. If the first two points are fulfilled then arrival process is random Poisson process with parameter λ that is average number of requests per time unit. Then probability of exactly k calls arriving in any time unit is defined by probability density function as [9]

$$p_k = P(X = k) = \frac{\lambda}{k!} e^{-\lambda} \tag{1}$$

Furthermore, if arrival process is random Poisson process then it can be easily shown that interarrival time is random variable with exponential distribution [10] and parameter $1/\lambda$.

Since this property of telecommunication traffic does not depend on transport technology in use, we can expect the same behaviour for calls in a VoIP network with many customers hence large population of requests generators.

Once a request for a connection (new call) arrives to the telecommunication system, it's either started to being served or blocked, in case of full capacity of the system is in use at the moment. By the term served, in this paper the creation and existence of connection with called party is meant. During this time portion of the system capacity is occupied. From queuing systems theory point of view, the server (agent) in this system is the channel (line) and number of servers is determined by trunk size. Therefore the serving process is the random process that describes the duration of each call. The duration of each telephone call is again a random variable that is exponentially distributed with parameter μ that defines number of served requests per time unit. Probability distribution function is then [9]

$$f(x) = \begin{cases} \mu e^{-\mu x} & x \geq 0 \\ 0 & x < 0 \end{cases} \tag{2}$$

and average call duration is $1/\mu$.

Our further analysis is based on average number of parallel existing connections. Let's start with system where transport capacity is not limited thus having a M/M/∞/∞ queuing system. This is an ideal system where arrival process is Poisson (parameter λ), serving time is exponentially distributed (parameter μ), the system has infinite capacity (each connection has a dedicated channel). Such a system can be described by a CTMC (Continuous Time Markov Chain) showed on figure 1. State numbers represent number of existing connections (requests being served).

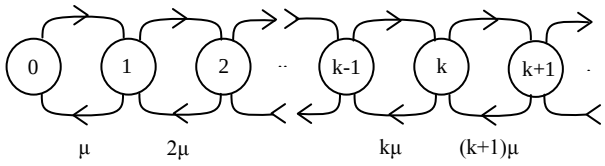



**Figure 1.** Graphical representation of CTMC of telephone calls

Probability of each state $p_k$ is defined as [11]

$$p_k = p_0 \prod_{i=0}^{k-1} \frac{\lambda}{(i+1)\mu} = \frac{\left(\frac{\lambda}{\mu}\right)^k}{k!} e^{-\frac{\lambda}{\mu}} \quad k = 0,1,\ldots,\infty \tag{3}$$

Then average number of requests being served can be calculated as

$$N = \sum_{k=0}^{\infty} k.p_k = \frac{\lambda}{\mu} \quad (4)$$

It is obvious the relative system load ρ can never exceed 1 since number of servers is infinite. However capacity (maximum number of paralel requests being served) of real system is usually limited. This phenomenon is well addressed by Erlang B model.

### 2.1 Erlang B model

Erlang B model is the basic model which does not contain the waiting queue. Incoming calls are assigned to the idle server / line directly if there is any available, otherwise they are considered blocked or lost [7]. This implies the Erlang B model is widely used to dimension the trunk capacity between Contact center and communication networks. Today, the Voice over IP (VoIP) technology is more and more important, but the basic capacity problem is only slightly modified to available data throughput of the connection. Thus Erlang B model can be used in this case as well.

The Erlang equation uses three basic parameters:

- A – the traffic load in Erlangs,
- N – number of lines / trunks (requested simultaneous connections),
- PB – probability of call blocking.

The original form of the equation allows us to find the blocking probability if A and N values are known:

$$P_B(N,A) = \frac{\frac{A^N}{N!}}{\sum_{i=0}^{N} \frac{A^i}{i!}} \quad (5)$$

If we know the rate of calls per time unit λ and the average number of served requests per the same time unit μ (so the average handling time is 1/μ, or average call duration time) then the traffic load can be easily evaluated as [12]

$$A = \frac{\lambda}{\mu} \quad (6)$$

If we substitute A in equation (5) we receive following form:

$$P_B(N,\lambda,\mu) = \frac{\left(\frac{\lambda}{\mu}\right)^N}{N!} \frac{1}{\sum_{i=0}^{N} \left[\left(\frac{\lambda}{\mu}\right)^i \frac{1}{i!}\right]} \quad (7)$$

We can see that it is the same formula as obtained for M/M/m/m queuing system [11], [12].

## 3. Model Adaptation for VoIP Environment

### 3.1 Differences between PSTN and VoIP traffic

As was stated in the previous section, original Erlang B model (formula) was defined for dimensioning of classical telecommunication trunks that interconnect e.g. two telecommunication switching centres. In such situation the trunk is considered as a queuing system with telephone calls as the requests and set of N parallel lines (figure 2) in the trunk as handling nodes. It is obvious the system cannot have a waiting queue, so calls that arrive during the period of occupation of all lines cannot be put through and are therefore lost.

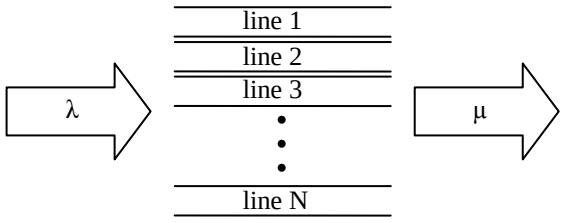



**Figure 2.** Classic telecommunications trunk of N lines

Requests arrival rate λ is simply defined as number of calls per time unit, whereas average request handling time 1/μ is the average call duration. As each line of trunk can transfer only one call at a time, the average call duration is equal to average line utilization time by one request.

To calculate the probability of call loss (blocking) the basic formula (5) can be used. The situation is slightly different in VoIP environment. The same link between two neighboring nodes of data network is shared among multiple data streams [14]. Thus the basic characteristic of connection is not the number of lines, but the throughput of the link, i.e. amount of data transferred per time unit. Data are transferred in form of packets of various lengths for various applications.

In this paper we focus on VoIP data only, thus we can specify more details of the data streams. Basically the principle of VoIP is to convert analog voice signal to binary representation, transfer the data in form of packets to the receiving node and backward conversion to analogue form [5]. Quality of voice reproduction depends on codec used for voice encoding / decoding, packet loss during the transmission, total cumulative delay during processing and transfer and several other factors. Each codec defines the set of rules for voice packetization / depacketization, sample size, packet size, number of voice samples in one packet, packetization interval, etc.

Audio streams in VoIP can be divided into two separate groups depending on variability of generated data traffic in time. CBR (constant bit-rate) sources generate one packet of predefined size (defined by used codec) per one packetization interval (both defined by codec in use) while VBR (variabile bit-rate) sources can adjust amount of information in each packet to input signal complexity thus utilizing advanced features of particular codec. In this paper we analyze CBR audio stream generated by G.711 audio codec, however the results can be generalized on any CBR stream.

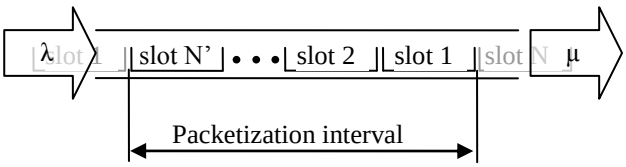



**Figure 3.** VoIP channel with virtual lines

Based on the abovementioned codec's parameters (frame (packet) size d and packetization interval τ) the required bandwidth per one VoIP connection utilizing particular codec can be derived as

$$w = \frac{d}{\tau} \quad [bit/s] \tag{8}$$

The link is shared by multiple connections utilizing a kind of time multiplex approach, thus idea of virtual telecommunication lines can be applied (figure 3).

### 3.2 Abstraction

The original Erlang B model provides a method to estimate call loss at defined traffic load A and set of lines in trunk N. For classic telecommunication networks it means only the calls above the capacity of the trunk are lost, but all other calls intact in terms of their quality. The situation is slightly different in IP world, where the link is shared. If the current transfer demands are higher than link capacity, frames are queued or lost, if the buffers are occupied. However, real time communication can tolerate only minimum level of delay. That means, if the queue is too long, packets can be discarded as their contingent transfer to the destination would occur to late and decoder would not be able to use the data. On the other hand, some level of packet loss can be accepted in VoIP communications without significant impact on communication quality since codecs in use have features to restore information of lost packet to certain degree.

If we abstract from transport technology in use, there is no difference between classic and VoIP connections from number of parallel connections in use over the trunk / link. We assume the Erlang B model can be used to estimate call loss probability in either cases. However for VoIP traffic depending on packet arrivals from sources, call loss does not necessary mean loss of all packets of the particular stream, but packets are lost randomly from all streams instead. If certain degree of lost packets is accepted, proportional increase of handled traffic load is obtainable. Therefore Erlang B model can be used to estimate the packet loss once traffic load A' and number of virtual links (link capacity) N' is known.

As described above, the data channel (link between nodes) is characterized by its capacity (link speed) W. Then theoretical link capacity expressed in terms of number of parallel connections the link can carry through can be calculated as

$$N' = \left\lfloor \frac{W}{w} \right\rfloor \tag{9}$$

The traffic load A' remains the same value as for classic telecommunication networks based on average call arrival rate from sources and average line occupation time (average call duration), thus

$$A' = \frac{\lambda}{\mu} \quad [Erl] \tag{10}$$

At this point, all input values to calculate probability of call loss or probability of packet loss $P_B$ (5) are known. Since we rounded N' towards minus infinity (took floor) of the fraction, the resulting value of $P_B$ gives us only upper theoretical bound for call loss, thus the eliminated part of capacity can positively influence the overall behaviour and for observed packet loss probability $P_B$' following can be stated

$$P'_B(N', A') < \frac{\frac{A'^{N'}}{N'!}}{\sum_{i=0}^{N'} \frac{A'^{i}}{i!}} \tag{11}$$

Furthermore the packet loss probability can be significantly influenced by buffer size that is associated to the particular data link. This phenomenon is discussed later.

## 4. Statistical characteristics of data traffic flows in VoIP

Packet flows generated by independent sources are transmitted to the network. At some point of the network, multiple flows share the same link, so the traffic is aggregated. Such an aggregated flow then enters network element, where all packets are put to corresponding buffers (queues) and later transmitted towards their destination. The network element is then a queuing system where requests are individual packets of aggregated stream and request handling time depends on outgoing link speed and packet size. In order to analyse this queuing system we have to describe the arrival process. Each existing telephone call generates flow of packets described above. If we have a look on particular source the probability of sending a packet in next short time interval Δt (shorter than packetization interval τ) is defined as

$$P_P = \begin{cases} \frac{\Delta t}{\tau} & \Delta t \in \langle 0; \tau \rangle \\ 1 & \Delta t \geq \tau \end{cases} \tag{12}$$

Combination of multiple sources brings us to random variable X denoting number of packets arrived on aggregated line during interval Δt. The distribution of X is binomial and its probability density function, or probability of generating i packets during interval Δt if n streams are aggregated is [9]

$$P_i = P[X = i] = \binom{n}{i} P_p^{\,i} (1 - P_p)^{n-i} \tag{13}$$

Then if n is sufficiently high and by shortening the interval Δt, thus lowering $P_p$ the binomial distribution can be easily substituted [10] by Poisson distribution with parameter $\lambda_p$

$$\lambda_p = nP_p = n\frac{\Delta t}{\tau} \tag{14}$$

and probability density function

$$P_i = \frac{\lambda_p}{i!}e^{-\lambda_p} \tag{15}$$

Modification of (14) so that arrival rate per time unit is expressed the formula can be altered to

$$\lambda_p = nP_p = n\frac{1}{\tau} \tag{16}$$

In previous section we showed, the average number of active calls N to be defined by (4), thus substituting n in (16) by N brings us to final form of arrival rate formula

$$\lambda_p = NP_p = \frac{\lambda}{\mu}\frac{1}{\tau} \tag{17}$$

Again, if arrival rate is Poisson then packet interarrival times are exponentially distributed.

For this paper, we consider sources to be constant bitrate. Therefore serving time for each packet is the same and depends only on packet size and link speed and is deterministic. The serving time is the interval for which the backed is being transmitted to the line so occupying the server. The serving rate then can be expressed as

$$\mu_p = \frac{W}{d} \tag{18}$$

where W is the link speed in bit/s and d is total packet size (in bits). We analyze situation with only one shared link used for transmission. Furthermore, buffer capacity of the network element is limited as well. That leads us to queuing system of M/D/1/K. K is obviously the buffer size (in packets) incremented by one.

### 4.1 Mathematical model of network element using M/G/1/K

The M/D/1/K system can be generalized to M/G/1/K system, or system with generally distributed serving time. States of this system are represented by number of requests (packets) present in the system at particular moment. To find probabilities of individual states, approach of imbedded Markov Chain can be used [15]. The chain events are the moments of the departure of a request (packet) after being served and chain states are the number of packets left by the departing request behind in the system. Therefore the states are 0 to K-1. Imbedded Markov Chain state probabilities $p_{d,i}$ are calculated in the usual fashion by solving K-1 balance equations and a normalisation condition (sum of all state probabilities to be zero). Once state probabilities of imbedded chain are known we can obtain probability of packet blocking or loss due to fully occupied queue as

$$P_L = 1 - \frac{1}{p_{d,0} + \lambda_p \overline{X}} = 1 - \frac{1}{p_{d,0} + \frac{\lambda_p}{\mu_p}} \tag{19}$$

where $\overline{X}$ is average serving time, that for our case is deterministic.

## 5. Simulation results

We decided to validate our ideas using simulation created in software Network Simulator 2. The network topology was very simple (figure 4) consisting of two nodes and a link between them.

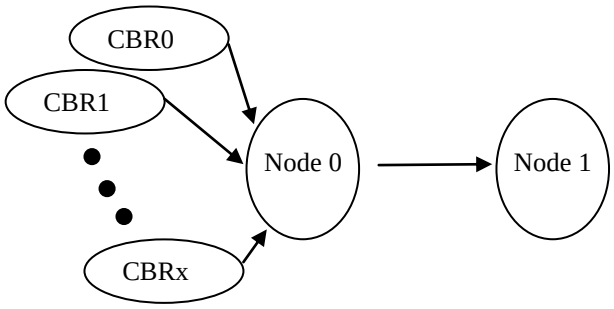



**Figure 4.** Simulated network topology

The first node worked as a source and second one was configured as a sink. The data transmission was done in one direction only, however this does not have any negative effect on results, since the link is configured as full duplex and both directions of traffic are isolated in nodes. Therefore simulation of only one direction is sufficient.

**Table 1.** G.711 characteristics

| | |
|---|---|
| Sampling frequency | 8 kHz |
| Sample size | 8 bits |
| Packetization interval | 20 ms |
| Number of samples per packet | 160 |
| Packet payload size | 160 B |
| Packet header size | 58 B |
| Nett bitrate per call (payload only) | 64 kbps |
| Gross bitrate per call (including headers) | 87.2 kbps |

Each call was simulated as separate CBR traffic source attached to node 0. The start time of transmission of a particular source was determined using Erlang distribution with parameter λ (the transformation to exponential distribution for interarrival time was used) together with call duration time (1/μ), after which the source was deactivated and stopped to send traffic. This fulfils the requirements for requests arrival distribution

and average request handling time defined by Erlang B model.

We decided to simulate G.711 codec as the basic option for all VoIP devices. The codec characteristics are following summarized in table 1 [16].

We decided to simulate G.711 codec as the basic option for all VoIP devices. Total header size was 58 bytes (L2 – L4 headers), that is 26.6% of total frame length (58 bytes header and 160 bytes payload). We decided to simulate three separate levels of traffic while maintaining the same required bandwidth to link capacity ratio (i.e. the same relative link utilization ρ). The absolute link capacity was constant for each traffic level for all relative load values. The relative link utilization ρ varied from 60% to 130%. We focused on packet loss level for each individual case. Each simulation round consisted of 10 000 simulated calls and 10 repetitions of every round were done.

**Table 2.** Packet loss probabilities for various traffic loads using modified Erlang B model

| ρ | A | N | $P_B$ |
|---|---|---|---|
| 0.6 | 6 | 10 | 4.31 |
| | 30 | 50 | 0.02 |
| | 60 | 100 | 0.00 |
| 0.7 | 7 | 10 | 7.87 |
| | 35 | 50 | 0.33 |
| | 70 | 100 | 0.01 |
| 0.8 | 8 | 10 | 12.17 |
| | 40 | 50 | 1.87 |
| | 80 | 100 | 0.40 |
| 0.9 | 9 | 10 | 16.80 |
| | 45 | 50 | 5.41 |
| | 90 | 100 | 2.70 |
| 1.0 | 10 | 10 | 21.46 |
| | 50 | 50 | 10.48 |
| | 100 | 100 | 7.57 |
| 1.1 | 11 | 10 | 25.96 |
| | 55 | 50 | 16.10 |
| | 110 | 100 | 13.61 |
| 1.2 | 12 | 10 | 30.19 |
| | 60 | 50 | 21.61 |
| | 120 | 100 | 19.63 |
| 1.3 | 13 | 10 | 34.12 |
| | 65 | 50 | 26.73 |
| | 130 | 100 | 25.16 |

Table 2 contains results obtained with modified Erlang B model (5). To correspond with all simulations carried later, values for several relative load values are presented. Relative load is the ratio between average expected traffic and link capacity. To compare results for various traffic loads, at each relative load class, calculations for three separate absolute traffic levels were done. This means, for relative load of 0.6 (i.e. 60% of link capacity occupied), instances with traffic loads 6, 30 and 60 Erl were chosen with corresponding link capacities of 10, 50 or 100 simultaneous calls. As expected, the packet loss probability is rising with higher relative load. Furthermore, for the same relative load, the packet loss probability is lower in case of higher absolute traffic. This phenomenon can be explained by better statistical utilization of the link when the amount of traffic is higher.

**Table 3.** Packet loss probability estimation using M/G/1/K model and various queue length (K=2..8)

| ρ | Queue size | | | | | | |
|---|---|---|---|---|---|---|---|
| | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 0.6 | 12.95 | 4.68 | 1.75 | 0.67 | 0.26 | 0.10 | 0.04 |
| 0.7 | 16.43 | 7.23 | 3.41 | 1.67 | 0.83 | 0.42 | 0.21 |
| 0.8 | 19.96 | 10.33 | 5.88 | 3.53 | 2.19 | 1.38 | 0.88 |
| 0.9 | 23.46 | 13.84 | 9.14 | 6.44 | 4.72 | 3.55 | 2.73 |
| 1.0 | 26.89 | 17.63 | 13.04 | 10.34 | 8.57 | 7.32 | 6.38 |
| 1.1 | 30.21 | 21.56 | 17.37 | 14.97 | 13.44 | 12.39 | 11.65 |
| 1.2 | 33.39 | 25.49 | 21.86 | 19.91 | 18.77 | 18.05 | 17.59 |
| 1.3 | 36.41 | 29.33 | 26.32 | 24.84 | 24.06 | 23.64 | 23.40 |

**Table 4.** Packet loss probability estimation using M/G/1/K model and various queue length (K=9..15)

| ρ | Queue size | | | | | | |
|---|---|---|---|---|---|---|---|
| | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| 0.6 | 0.01 | 0.01 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 0.7 | 0.11 | 0.05 | 0.03 | 0.01 | 0.01 | 0.00 | 0.00 |
| 0.8 | 0.56 | 0.36 | 0.24 | 0.15 | 0.10 | 0.06 | 0.04 |
| 0.9 | 2.12 | 1.66 | 1.32 | 1.05 | 0.84 | 0.67 | 0.54 |
| 1.0 | 5.66 | 5.08 | 4.62 | 4.23 | 3.90 | 3.61 | 3.37 |
| 1.1 | 11.10 | 10.68 | 10.36 | 10.11 | 9.92 | 9.76 | 9.64 |
| 1.2 | 17.29 | 17.09 | 16.96 | 16.86 | 16.80 | 16.76 | 16.73 |
| 1.3 | 23.26 | 23.18 | 23.14 | 23.11 | 23.10 | 23.09 | 23.08 |

Tables 3 and 4 contain results of calculations based on M/G/1/K (or in this case M/D/1/K) model (19) for various traffic load levels (same as in previous case) and various queue lengths. The queue size is expressed in number of packets that can be present in the queue at the same time. As expected, with rising relative loads the level of packet loss for the same queue size is increasing. Furthermore for any specific relative load situation, increase of queue capacity is resulting to generally lower packet loss probability. Cases where relative load is below 1 tend to converge toward zero packet loss probability as the queue size is being larger. If the relative traffic load has value higher than 1 (i.e. the link is overstressed), some portion of packets must be dropped. The loss probability however converges to a constant value as the queue size becomes larger.

Tables 5 and 6 display the complete packet loss ratios for different queue size configuration and absolute and relative traffic levels as observed during simulation. The queue worked according to FIFO principle and a packet loss occurred once the queue was fully occupied by other requests and a new packet arrived. As expected, with rising relative loads the level of packet loss for the same queue size is increasing for all three absolute traffic

amount configurations. Furthermore for any specific relative load situation, increase of queue capacity is resulting to generally lower packet loss probability.

**Table 5.** Simulation results – measured packet loss for various traffic loads and queue sizes (K=2..8)

| ρ | A | Queue size | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | **2** | **3** | **4** | **5** | **6** | **7** | **8** |
| 0.60 | 6 | 12.91 | 4.61 | 2.18 | 1.29 | 1.21 | 1.29 | 1.18 |
| | 30 | 12.94 | 4.58 | 1.66 | 0.66 | 0.26 | 0.10 | 0.04 |
| | 60 | 12.80 | 4.70 | 1.81 | 0.66 | 0.25 | 0.11 | 0.03 |
| 0.70 | 7 | 16.52 | 7.06 | 3.58 | 2.99 | 2.89 | 2.93 | 2.70 |
| | 35 | 16.55 | 7.27 | 3.31 | 1.66 | 0.93 | 0.39 | 0.23 |
| | 70 | 16.54 | 7.13 | 3.27 | 1.56 | 0.84 | 0.44 | 0.23 |
| 0.80 | 8 | 19.63 | 10.32 | 5.91 | 5.69 | 5.27 | 5.15 | 5.43 |
| | 40 | 19.82 | 10.20 | 5.76 | 3.59 | 2.13 | 1.45 | 0.95 |
| | 80 | 19.78 | 10.26 | 5.77 | 3.60 | 2.18 | 1.35 | 0.88 |
| 0.90 | 9 | 23.86 | 13.67 | 9.69 | 9.16 | 8.55 | 8.58 | 8.60 |
| | 45 | 23.40 | 13.63 | 9.09 | 6.59 | 4.53 | 3.54 | 2.83 |
| | 90 | 23.22 | 13.83 | 9.24 | 6.57 | 4.71 | 3.45 | 2.70 |
| 1.00 | 10 | 26.81 | 17.23 | 13.43 | 12.87 | 12.79 | 12.34 | 12.43 |
| | 50 | 26.86 | 17.63 | 13.29 | 10.06 | 8.76 | 7.04 | 6.23 |
| | 100 | 26.69 | 17.19 | 13.04 | 10.25 | 8.60 | 7.12 | 6.13 |
| 1.10 | 11 | 29.68 | 21.21 | 17.82 | 16.75 | 16.75 | 16.71 | 16.65 |
| | 55 | 29.96 | 21.42 | 16.89 | 15.42 | 13.37 | 12.34 | 11.25 |
| | 110 | 29.81 | 21.30 | 17.09 | 14.99 | 13.80 | 12.23 | 11.30 |
| 1.20 | 12 | 33.33 | 24.88 | 22.56 | 21.06 | 21.09 | 21.35 | 21.24 |
| | 60 | 33.39 | 25.17 | 21.81 | 19.66 | 18.74 | 18.22 | 18.01 |
| | 120 | 32.99 | 25.07 | 21.65 | 19.93 | 18.45 | 17.78 | 17.41 |
| 1.30 | 13 | 36.16 | 29.24 | 26.63 | 25.60 | 25.69 | 25.85 | 26.27 |
| | 65 | 36.59 | 29.16 | 26.19 | 24.43 | 23.85 | 23.15 | 23.20 |
| | 130 | 36.09 | 28.94 | 25.57 | 24.82 | 23.48 | 22.90 | 22.10 |

Figures 5 - 8 graphically display the results of simulations and calculations for three selected relative load values (0.6, 0.8 and 1.0). Each chart compares the packet loss probabilities measured in all three different absolute traffic load levels (with link capacity of 10, 50 or 100 parallel calls and appropriate absolute traffic load A). We can observe several interesting facts from them. First, in all cases there is a tendency to converge to stable packet loss ratio value and enlarging the queue beyond certain size does not influence the probability at all. For all cases, the critical queue size value seems to be around 10 elements. Second, for relative load levels below 1.0 (that means the link is not fully used) the packet loss probability converges towards zero, especially in situations with higher absolute traffic levels. This is however not valid for lower absolute traffic level. This brings us to the third important point that for low absolute traffic load, the simulation results deviates from M/G/1/K mathematical model expectations when the queue size is more than 4 – 5 packets. There are two key factors that contribute to it. For such a low absolute level of traffic, the flow of packet does not exactly follow Poisson distribution. It is much more probable that only one connection is active at the time that means it increases the probability of occurrence of $\tau$ interpacket times. Furthermore the changes of traffic (start of new call or end of another one) do not occur as often as with higher load. Once the overload situation occurs, it has tendency to last longer (seconds or more) and even making the queue larger cannot prevent it from filling up to 100% rapidly. Therefore for lower traffic load (20 Erl and less) it is advisable to use adapted Erlang B model to estimate packet loss or to expect higher packet loss than M/G/1K model calculation.

**Table 6.** Simulation results – measured packet loss for various traffic loads and queue sizes (K=9..15)

| ρ | A | Queue size | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | **9** | **10** | **11** | **12** | **13** | **14** | **15** |
| 0.60 | 6 | 1.30 | 1.32 | 1.29 | 1.36 | 1.22 | 1.33 | 1.29 |
| | 30 | 0.02 | 0.00 | 0.01 | 0.00 | 0.00 | 0.00 | 0.00 |
| | 60 | 0.02 | 0.01 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 0.70 | 7 | 3.05 | 2.79 | 2.84 | 2.85 | 2.85 | 2.90 | 2.92 |
| | 35 | 0.14 | 0.12 | 0.05 | 0.05 | 0.04 | 0.05 | 0.03 |
| | 70 | 0.11 | 0.08 | 0.03 | 0.01 | 0.01 | 0.00 | 0.00 |
| 0.80 | 8 | 5.16 | 5.53 | 5.43 | 5.23 | 5.37 | 5.29 | 5.43 |
| | 40 | 0.72 | 0.45 | 0.48 | 0.51 | 0.45 | 0.51 | 0.44 |
| | 80 | 0.53 | 0.38 | 0.23 | 0.18 | 0.14 | 0.08 | 0.09 |
| 0.90 | 9 | 8.48 | 8.71 | 8.69 | 8.57 | 8.88 | 8.57 | 8.53 |
| | 45 | 2.64 | 2.14 | 1.94 | 2.21 | 1.85 | 2.04 | 1.96 |
| | 90 | 2.33 | 1.76 | 1.30 | 1.06 | 1.03 | 0.91 | 0.77 |
| 1.00 | 10 | 12.60 | 12.40 | 12.41 | 12.47 | 12.41 | 12.53 | 12.82 |
| | 50 | 6.04 | 5.49 | 5.63 | 6.04 | 5.47 | 5.56 | 5.53 |
| | 100 | 5.24 | 4.74 | 4.59 | 4.04 | 3.99 | 4.22 | 3.64 |
| 1.10 | 11 | 16.85 | 16.61 | 16.81 | 16.75 | 16.95 | 16.89 | 16.87 |
| | 55 | 11.07 | 11.42 | 10.75 | 11.02 | 10.75 | 11.24 | 11.19 |
| | 110 | 10.77 | 10.38 | 10.03 | 10.12 | 10.13 | 9.81 | 9.34 |
| 1.20 | 12 | 21.40 | 21.36 | 21.06 | 21.21 | 21.56 | 21.42 | 21.27 |
| | 60 | 16.30 | 16.87 | 16.75 | 16.95 | 17.16 | 16.74 | 16.91 |
| | 120 | 16.89 | 16.73 | 16.48 | 16.41 | 16.34 | 16.08 | 16.53 |
| 1.30 | 13 | 25.80 | 26.16 | 25.77 | 25.66 | 25.67 | 25.96 | 25.58 |
| | 65 | 23.24 | 22.96 | 23.45 | 22.26 | 23.08 | 22.94 | 22.86 |
| | 130 | 23.26 | 22.94 | 22.45 | 22.71 | 22.67 | 22.87 | 22.96 |

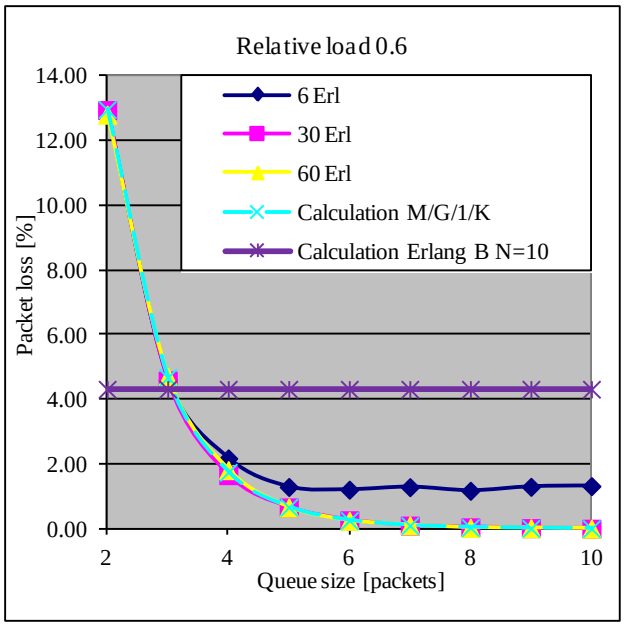



**Figure 5.** Packet loss probability using various queue sizes (relative load 0.6)

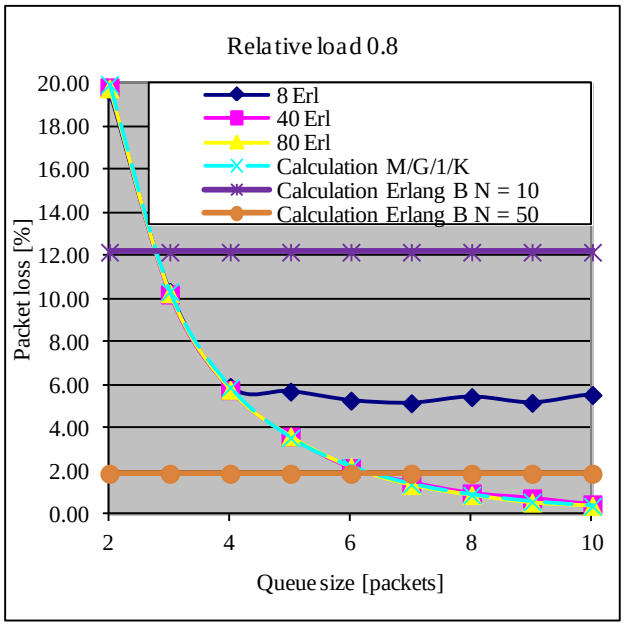



**Figure 6.** Packet loss probability using various queue sizes (relative load 0.8)

If we compare the results of adapted Erlang B model with measured values, we clearly some differences. Since Erlang B model does not take queue size as input parameter, it provides us with one output value for absolute level only. For all case however this value tends to be above measured values. This confirms the theory that Erlang B model can be used to estimate an upper bound for packet loss ratio except some extreme case with only very short queue size of 1 – 3 elements, where, especially for higher absolute traffic levels the results of the model can significantly deviate from reality. In this case, usage of M/G/1/K gives better results.

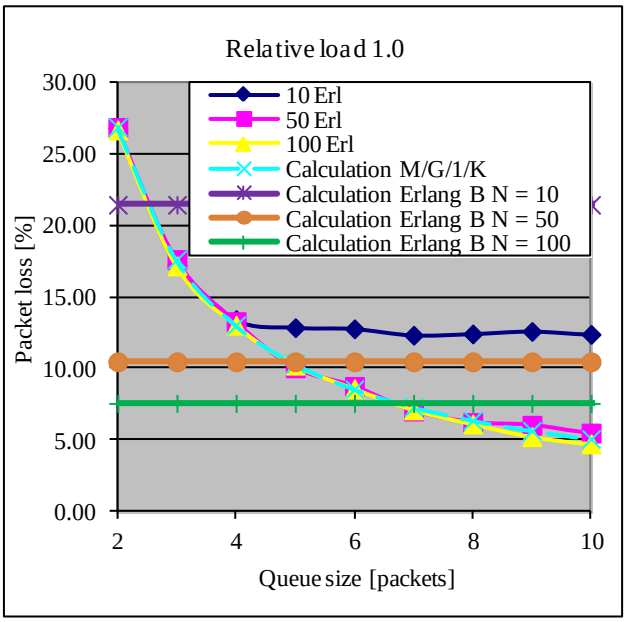



**Figure 7.** Packet loss probability using various queue sizes (relative load 1.0)

## 6. Conclusion

This paper discusses the topic of packet loss in a simple VoIP network by a defining two relatively simple mathematical models and execution of extensive simulations to compare the results for various input conditions. Simulation results showed that usability of both models is quite good but also shows some deviations of results for specific cases. Erlang B model provides good upper bound estimation for instances where absolute traffic load is relatively small (up till 20 Erl), while M/G/1/K model gives better results for cases with higher absolute traffic load and has tendency to underestimate situations with low overall traffic and longer queue sizes. For practical applications, results of both models should be obtained and compared. The higher of these values is guaranteed to be upper bound for packet loss ratio.

## 7. Acknowledgement

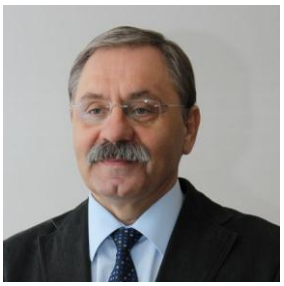
This work is a part of research activities conducted at Slovak University of Technology Bratislava, Faculty of Electrical Engineering and Information Technology, Institute of Telecommunications, within the scope of the projects VEGA No. 1/0186/12 „Modeling of Multimedia Traffic Parameters in IMS Networks" and „Support of Center of Excellence for SMART Technologies, Systems and Services II., ITMS 26240120029, co-funded by the ERDF".

## 8. References

[1] A. K. Erlang, "The Theory of Probabilities and Telephone Conversations", In *Nyt Tidsskrift for Matematik*, 1909, vol. 20, no. B, p. 33 – 39, Available at: <http://oldwww.com.dtu.dk/teletraffic/erlangbook/pps131-137.pdf>.

[2] Ch. Grimm, G. Schlüchtermann, "IP Traffic Theory and Performance", Springer, Berlin, Heidelberg, Germany, 2008, ISBN 978-3-540-70603-8.
[3] M. Kováčik, "Hodnotenie kvality služieb VoIP", Slovak VoIP Telephony, Banská Bystrica, Slovakia, 2010-09-29/30, Avaliable at: <http://www.voip-forum.sk/archiv/Kovacik_hodnotenieVoIP.pdf>.
[4] L1 ASSOCIATES, "An Introduction to VoIP" L1 Assocates, [s.l], 2003-12-18, Available at: <http://www.l1associates.com/Introduction%20to%20VoIP.pdf >.
[5] S. Kľúčik, A. Tisovský, "Queuing Systems in Multimedia Networks", In "*Elektrorevue*", vol. 15, art. no 99, Nov 2010, ISSN 1213-1539.
[6] H. Schulzrinne, "Audio codecs", Columbia University, New York, USA, 2008-10-01, Available at: <http://www.cs.columbia.edu/~hgs/audio/codecs.html>.
[7] Diagnostic Strategies, "Traffic Modeling and Resource Allocation in Call Centers". Diagnostic Strategies, Needham, Mass., USA, 2003.
[8] E. Chromy, et. al.., "Markov Models and Their Use for Calculations of Important Traffic Parameters of Contact Center", In *WSEAS TRANSACTIONS on COMMUNICATIONS*, issue 11, vol. 10, November 2011, ISSN: 1109 2742, pp. 341-350.
[9] G. Bolch, et al., "Queuing Networks and Markov Chains," 2nd ed, John Wiley, Hoboken, New Jersey, USA, c2006, 878 p, ISBN 0-471-56525-3.
[10] J. F Hayes, T. V. J Ganesh Babu, "Modeling and Analysis of Telecommunications Networks," 1st. ed., John Wiley, Hoboken, New Jersey, USA, c2004, 399 p, ISBN 0-471-34845-7.
[11] J. Polec, T. Karlubíková, "Stochastic models in telecommunications 1", 1st ed., FABER, Bratislava, Slovakia, 1999.
[12] G. Koole, "Call Center Mathematics: A scientific method for understanding and improving contact centers", Vrije Universiteit , Amsterdam, Netherland, Available at: <http://www.math.vu.nl/~koole/ccmath>.
[13] L. Unčovský, "Stochastic models of operational analysis, 1st ed., ALFA, Bratislava, Slovakia, 1980.
[14] E. Chromý, M. Kavacký, "Asynchronous Networks and Erlang Formulas", In *International Journal of Communication Networks and Information Security*, vol. 2, no. 2, 2010, ISSN 2073-607X, p. 85-89
[15] S. K. Bose, "Analysis of a M/G/1/K Queue without Vacations", Indian Institute of Technology, Guwahati, India.
[16] H. Schulzrinne, "RFC 3551, RTP Profile for Audio and Video Conferences with Minimal Control", Columbia University, [s.l.], 2003, Available at: <http://tools.ietf.org/html/rfc3551>.

**Tibor MIŠUTH, MSc.** was born in Žilina, Slovakia on November 1984. He received Master of Science degree (electrical engineering) from Slovak Technical University Bratislava in 2009. Since then he continues as postgraduate student at Institute of Telecommunications STU Bratislava. In May 2012 he submitted his dissertaion "Application of queuing systems in IP networks." He focuses on problems of digital switching systems, NGN, VoIP, QoS and application of queuing theory to modern telecommunication environment.



**Ivan BAROŇÁK, prof. MSc., Ph.D.** was born in Žilina, Slovakia, on July 1955. He received Master of Science degree (electrical engineering) from Slovak Technical University Bratislava in 1980. Since 1981 he has been a lecturer at Department of Telecommunications STU Bratislava. Nowadays he works as a professor at Institute of Telecommunications of FEI STU Bratislava. Scientifically, professionally and pedagogically he focuses on problems of digital switching systems, ATM, Telecommunication management (TMN), NGN, VoIP, QoS, problem of optimal modeling of private telecommunication networks and services