Delay Model of RTP Flows in Accordance with M/D/1 and M/D/2 Kendall's Notation

M. Voznak, M. Halas, B. Borowik and Z. Kocur

Abstract**—** This article deals with the mathematical model of an end-to-end delay and a delay variation in VoIP connections going through a two priority queue' serving system. We focused on delay caused by network components and its mathematical formulations. Our mathematical computational model is able to predict an estimated delay of packets in RTP flows, especially dealy variation in routers handling RTP packets in priority queue. We applied M/D/1 and M/D/2 queuing models and we expressed a probability of RTP packets waiting in the queue and all partial delay components and mechanisms. We determined the domain of validity in performed experiments and the proposed mathematical model is suitable for the approximation of voice traffic in network with priority queuing which consists of sources with the Poisson's probability distribution.

Keywords**—**Estimated delay, Jitter, Mathematical model, Queuing system, RTP flows, VoIP.

I. Introduction

The delay is one of the main issues in packet-based networks which have impact on QoS and can have various causes including propagation, handling or processing. There are several types of delays in an IP network which differ from each other as to where they are created, mechanism of their creation or some other attributes. Each delay component influences the result voice packet delay in a different way. This paper provides a detailed description of individual delay components, and explains mechanisms of their creation. Subsequently, it focuses on the creation of a mathematical model of a VoIP end-to-end delay in the network.

The delay components should be classified based on the place of their creation: Coder and packetization delay in transmitter, Queuing, serialization and propagation delay in transmission network, De-jitter, de-packetization and decompression delay in receiver.

II. Dealy Components

We can find two types of delay in the transmitter. The first is a *coder delay* and is affected by the used codec. It has two components: the *frame size delay* and the *look-ahead delay*. Their values are exactly defined for any particular coder, e.g. for the ITU-T G.711 (PCM) codec it is 0.125 ms frame size without look-ahead and for the ITU-T G.729 (CS-ACELP codec) it's the frame size value 10 ms and 5 ms look-ahead. The second type of delay in the transmitter is the *packetization delay*. This delay occurs when data blocks are encapsulated into packets and transmitted by the network. The packetization delay is set as multiples of the packetization period used by particular codec and specifies how many data blocks are transmitted in one packet [1], [2]. The estimation process is given by the following equation:

$$T_{PD} = \frac{P_S}{C_{BW}} \quad [\text{ms}] \qquad (1)$$

where
T_{PD} is packetization delay [ms],
P_S is payload size [b],
C_{BW} is codec bandwidth [kbps].

We can incur three types of delays in the receiver. The first type is the *de-jitter delay* which is closely related to the variable delay in the network when it is necessary to eliminate a variance of these variable components using supplementary buffer store in the receiver, this buffer is called a *playout* buffer. Its size is typically adjusted as a multiple of the packetization delay because of an optimization. If this value is adjusted statistically, then jitter buffer sizes are about 30-90 ms, a typical value is 60 ms. If the variable playout buffer is used, the size is adapted based on the real-time delay variation. In this case the typical maximum value is about 150 ms [3].

The second type is a *depacketization delay.* Its mechanism is very similar to that of the packetization delay mentioned above. The depacketization is a reverse packetization and therefore the size of depacketization delay of one block in the frame is in correlation with its packetization delay. In real traffic the delay of each block within the frame of one packet occurs, always only for the value of the packetization delay. This is why we count with only one constant packetization delay value.

The third type is a *decompression delay*. The decompression delay, similarly to the coder delay depends on

Manuscript received March 23, 2011. The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 218086.

M. Voznak is with the VSB – Technical University of Ostrava, 17. listopadu 15, 708 33 Ostrava Poruba, Czech Republic (phone: +420 5969991699 ; e-mail: miroslav.voznak@vsb.cz).

M . Halas is with the Slovak University of Technology, Ilkovicova 3, 812 19 Bratislava, Slovak Republic (e-mail: halas@ktl.elf.stuba.sk).

B. Borowik is with the University of Bielsko-Biala, Willowa 2, 43-309 Bielsko-Biala, Poland (e-mail: bo@borowik.info).

Z. Kocur is with the the Czech Technical University in Prague, Technicka 2, 160 00 Prague, Czech Republic (e-mail: zokl@fel.cvut.cz).

the compressing algorithm selection. On average, the decompression delay is approximately 10 % of the compressing codec delay for each voice block in the packet. But it is very dependent on the computing decoder operation and mainly on the number of voice blocks in one packet. This decompression delay might be defined by the following formula:

$$T_{DCD} = 0.1 \cdot N \cdot T_{CD} \text{ [ms]} \qquad (2)$$

where
T_{DCD} is decompression delay [ms],
N is number of the voice blocks in the packet,
T_{CD} is coder delay [ms].

The last component of our classification is a delay in the transmission network. Again, there are three types of this delay. The first one depends on the transmission rate of the used interface and it is called as a *serialization delay*. The packet sending takes some time. This time depends on the transmission medium rate and on the size of packet. Relation (3) shows estimation of the time:

$$T_{SER} = \frac{P_S + H_L}{L_S} \text{ [ms]} \qquad (3)$$

where
T_{SER} is serialization delay [ms],
L_S is line speed [kbit/s],
P_S is payload size [b],
H_L is header length [b].

The second type of a delay originated in the transmission network is the *propagation delay*. This delay relates to the signal transmission, i.e. to its physical regularities of the propagation in the surroundings. It depends on the used transmission technology, in particular on the distance over which the signal is transmitted. Today's networks are mostly built on single mode optical fibers. The speed of light in optical fiber is $2.07 \cdot 10^{-8}$ [m/s], from which the propagation delay should be defined using the following formula:

$$T_{PROP} = \frac{L}{v} \text{ [ms]} \qquad (4)$$

where
T_{Prop} is propagation delay [ms]
L is line length [km]
v is speed of light in optical fiber = $2.07 \cdot 10^{-8}$ [m/s]

The last type is the delay which occurs in active elements of the transmission network and relates to handling of RTP packets, in particular in the router queues. This delay is the most significant part of the jitter. A delay variation or a jitter is a metric that describes the level of disturbance of packet arrival times compared to the ideal arrival time. Such disturbances can be caused by queuing or by processing [1], [4].

III. Dealy Variation

This topic is discussed in many publications and queuing theory provides solution to many issues [5]. It involves mathematical analysis of processes including arrival at the input of a queue, waiting in the queue and the serving at the front of the queue and providing the appropriate performance parameters of the designed model.

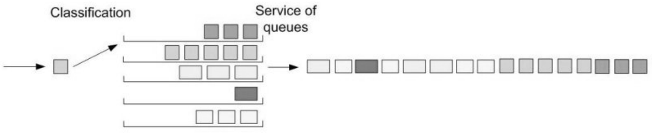

Fig. 1 Priority Queuing

It is proven that in certain circumstances the voice traffic can be modelled by a source signal the probabilistic random variable distribution of which matches Poisson's probability distribution. We can usually trace an influence of a jitter in the routers equipped with low-speed links. These routers often operate with PQ optimization (Priority Queuing). Priority queuing is mainly used for serving the voice flow and is based on apreferred packet sorting so that the selected packets are placed into priority queue [1].

A router with four FIFO queues (at least two are necessary) is shown in Figure 1. Each queue has been assigned a different priority, there is a classifier making the decision in which of the queues to place the packet and a scheduler picking the packets starting with the higher priority queue, next with lower priority etc.

A. M/D/1 Delay Model

Any packets in the high priority queue must be served first. When the queue is empty, a queue with lower priority can be served. If there is an effectively utilized packet fragmentation mechanism on the output of the line, it is possible to mitigate the influence of the serialization delay in data packets with a lower priority than that of the voice packets. In this case, for the modelling requirements of traffic loading and delay in router, it is sufficient to watch a delay only in the priority queue. Servicing requirement technique in the priority queue corresponds to the model of queuing system *M/D/1/k*, where *k* is size of buffer. The model notation used corresponds with Kendall's notation of queuing models [5].

In order to create an analytical model of the switching delay we can ignore the buffer size and count with a system of sufficient buffer size in which the loss of preferred packets doesn't occur. If this *M/D/1/k* model can be replaced by *M/D/1/∞* model, we are able to create an analytical expression of switch buffer store seizing. Consequently it is easier to gain an analytical model of the delay in the queue. The conditions for validating the designed model are described below.

The arrival process is a Poisson process with an exponentially distributed random variable, we consider that every source of a stream corresponds to the Poisson distribution and therefore their sum also corresponds to Poisson distribution [4].

The arrival rate λ(t) is a constant λ, it means we assume that only one type of the codec is used and there are *M-sources*, a service process in priority queue is FIFO (First In First Out).

The service rate μ(t) is a constant because the same codec is used, we assume that *the number of waiting positions in a priority queue is infinite*.

We express the utilization of the system in equation (5) and for stability we must have $0 \leq \rho < 1$.

$$\rho = \frac{\lambda}{\mu} \tag{5}$$

where
λ is arrival rate [s^{-1}]
μ is service rate [s^{-1}]
ρ is system utilization

We can express the arrival rate by the following equation:

$$\lambda = \frac{C_{BW}}{P_S} \ [\mathrm{s}^{-1}] \tag{6}$$

and the service rate by the equation (7) below:

$$\mu = \frac{1}{T_{SER} + T_S} \ [\mathrm{s}^{-1}] \tag{7}$$

where
T_{SER} is serialization delay [s],
T_S is processing time (handling by processor) [s].

The probability that *k* attempts will be waiting in the queue is:

$$p_k = (1-\rho)\sum_{j=1}^{k}(-1)^{k-j}\left(j\rho\right)^{k-j-1}\frac{(j\rho+k-j)e^{j\rho}}{(k-j)!} \quad \text{for } k \geq 2 \tag{8}$$

$$p_k = (1-\rho)\cdot(e^{\rho}-1) \quad \text{for } k=1 \tag{9}$$

$$p_k = (1-\rho) \quad \text{for } k=0 \tag{10}$$

Equation (11) determines mean service time (1/μ is the service time of one request).

$$T = \frac{1}{\mu} + \frac{\rho}{2(1-\rho)\mu} \ [\mathrm{s}] \tag{11}$$

N in equation (12) stands for the mean number of attempts in the system:

$$N = T \cdot \lambda \tag{12}$$

We assume that there are M sources with Poisson distribution of inter-arrival times and that all RTP streams use the same codec. Then we can express the arrival rate as follows:

$$\lambda = M\,\frac{C_{BW}}{P_S} \ [\mathrm{s}^{-1}] \tag{13}$$

We know the transmission speed of the low-speed link and subsequently we can derive the equation for the calculation of the service rate in the system. We apply the relation (3) to the relation (7) and obtain the following result:

$$\mu = \frac{L_S}{P_S + H_L + L_S \cdot T_S} \quad [\mathrm{s}^{-1}] \tag{14}$$

We apply the relations (13) and (14) to (5) and obtain the following equation for the system utilization:

$$\rho = \frac{M \cdot C_{BW} \cdot (P_S + H_L + L_S \cdot T_S)}{L_S \cdot P_S} \tag{15}$$

Likewise a relation for the probability that k-attempts will be waiting in the queue can be derived from equations (13), (14) and(15) applied to (8), (9) and (10):

$$p_k = (1 - MC_{BW}\left(\frac{P_S + H_L + L_S \cdot T_S}{L_S P_S}\right) \cdot \sum_{j=1}^{k}\left[-1^{(k-j)} \cdot \left(j \cdot M \cdot C_{BW}\,\frac{P_S + H_L + L_S \cdot T_S}{P_S L_S}\right)^{k-1-j} \cdot \left(j \cdot M \cdot C_{BW}\,\frac{P_S + H_L + L_S \cdot T_S}{P_S L_S} + k - j \right) \cdot \frac{e^{jMC_{BW}\frac{P_S + H_L + L_S \cdot T_S}{P_S L_S}}}{(k-j)!} \right] \quad \text{for } k \geq 2 \tag{16}$$

$$p_k = (1 - M C_{BW}\left(\frac{P_S + H_L + L_S \cdot T_S}{L_S P_S}\right) \cdot \left(e^{jMC_{BW}\frac{P_S + H_L + L_S \cdot T_S}{P_S L_S}} - 1 \right) \quad \text{for } k = 1 \tag{17}$$

$$p_k = (1 - M C_{BW}\left(\frac{P_S + H_L + L_S \cdot T_S}{L_S P_S}\right) \quad \text{for } k=0 \tag{18}$$

The probability of waiting in the queue is expressed in the following relation (19):

$$p_{Tk} = p_k \frac{P_S + H_L + L_S \cdot T_S}{L_S P_S} \tag{19}$$

Equation (20) derived from equations (13), (14), (15) and (11) above expresses the mean service time:

$$T = \frac{1}{2} \cdot \frac{P_S + H_L + L_S T_S}{L_S} \cdot \frac{2P_S L_S - C_{BW} M(P_S + H_L + L_S T_S)}{P_S L_S - C_{BW} M(P_S + H_L + L_S T_S)} \tag{20}$$

B. M/D/1 Delay Model

Based on the analysis of the principle of service models with two priority queues, we can assume that delays in a higher priority service queue are shorter. If we opt for better system resources for the high-priority service queue, we are likely to experience poorer system resources for voice packets in the lower-priority service queue. In order to be able to express the delay in the service element with priority queues, it is necessary to monitor solely queues that are used for the voice flow service. The method applied to transmit voice packets in priority queues corresponds with the *M/D/n/k* model where *n* is the number of service queues and *k* is the size of the cache memory [4].

In order to express the mathematical model which uses two service queues for transmission of voice streams, we can substitute the *M/D/n/k* model by the *M/D/2/k* model. If we disregard the size of cache memory then this assumption enables us to replace the *M/D/2/k* model by the *M/D/2* model. The conditions for validating the *M/D/2* model are as follows:

No interruption of the priority service process. Packets in the higher priority queue are served before packets in the lower priority queue. When a packet with some priority arrives, the service is provided first.

Priority queues are served based on the FIFO (First In First Out) method and *the arrival process corresponds to the Poisson distribution*. Where every single stream matches the Poisson distribution, then the sum of such streams also matches the Poisson distribution,

The service rate is a constant because the same codec and packets of the same size are used and *the arrival rate is also a constant* since we assume a constant number of the flows with the same codec,

The size of the priority queue's *cache memory is infinite*.

The system's utilisation is express by the formula (5). The utilisation of a system with two priority queues can be express as follows:

$$\rho = \rho_1 + \rho_2 \tag{21}$$

where ρ_i is the utilisation of the system queue. So, the system utilisation is express as:

$$\rho = \frac{\lambda_1 + \lambda_2}{\mu} \tag{22}$$

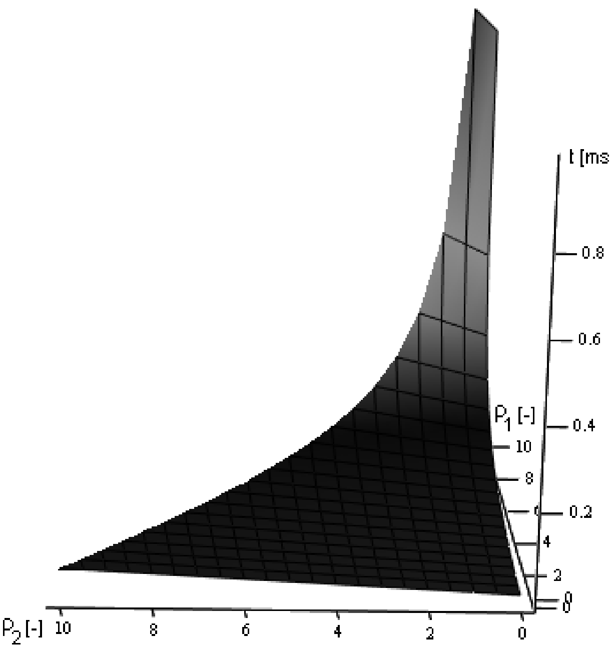

Fig. 2 The higher Priority queue: Relation between the mean service time in priority queue and system utilisation of the queue

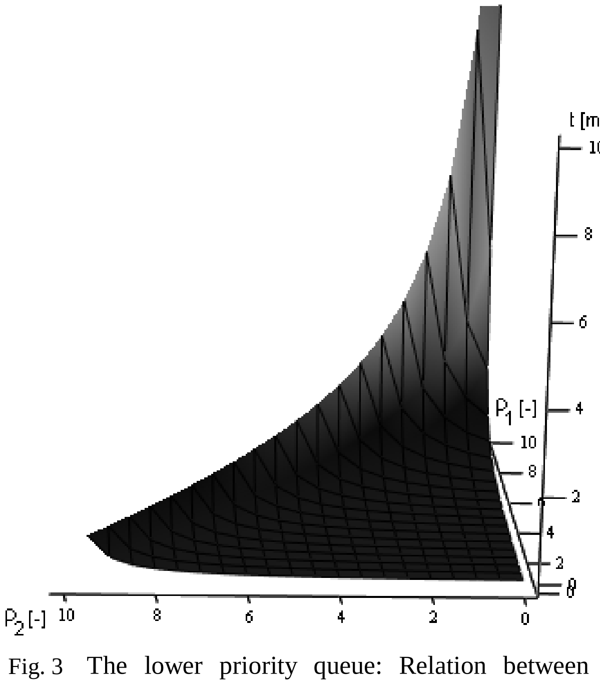

Fig. 3 The lower priority queue: Relation between the mean service time in priority queue and system utilisation of the queue

The arrival rate is expressed by equation (6), for M streams we can it modify fro particular RTP stream:

$$\lambda_i = M_i \frac{C_{BW}}{P_s} \qquad (23)$$

where
M_i – number of streams in queue i [-]
C_{BW} – codec bandwidth [b/s]
P_S – payload size [b]

Then for the service rate we adopt realtion (7). The mean service time of the process in a higher priority queue is express as follows:

$$\overline{T}_1 = \frac{1}{\mu} + \frac{\rho}{2\mu(1-\rho_1)} \qquad (24)$$

Similarly, the mean service time of the process in a lower priority queue:

$$\overline{T}_2 = \frac{1}{\mu} + \frac{\rho}{2\mu(1-\rho)(1-\rho_1)} \qquad (25)$$

The relation between the mean service time and the system utilisation of the queue is shown in the Figure 2.

In the system *without interruption* the mean service time is generally the sum of the service time, time of the remaining services, time it takes voice packets included in the same or a higher-priority queue to be transmitted and time needed to transmit voice packets of a higher priority that came while the packet was waiting to be processed by the system.

A key parameter is the time it takes to process a service element. This parameter needs to be determined individually for each service element. It is determined by hardware (processor, motherboard and network card, etc.) and software (operating system, kernel, etc.) used. The only option to determine the processing time is based on knowledge of the behaviour characteristic of the element in the increasing load.

Assuming we know both the line speed and the processing time, we express the service rate by the equation (14). We apply equation (15) to ouir situation and the system utilisation we expresse as follows:

$$\rho = \frac{C_{BW} \cdot (M_1 + M_2) \cdot (P_S + H_L + L_S \cdot T_S)}{L_S \cdot P_S} \qquad (26)$$

The mean service time of the process in a higher priority queue can be express by the following formula:

$$T_1 = \frac{1}{2} \cdot \frac{P_S + H_L + L_S \cdot T_S}{L_S} \cdot \frac{2P_S \cdot L_S - C_{BW}\left(M_1 - M_2\right)\left(P_S + H_L + L_S \cdot T_S\right)}{P_S \cdot L_S - C_{BW} \cdot M_1\left(P_S + H_L + L_S \cdot T_S\right)} \qquad (27)$$

Analogically, the mean service time of the process in a lower priority queue:

$$T_2 = \frac{1}{2} \cdot \frac{P_S + H_L + L_S \cdot T_S}{L_S} \cdot \\ \cdot \frac{2\left(P_S \cdot L_S\right)^2 - C_{BW}\left(M + 2M_1\right)\left(P_S + H_L + L_S \cdot T_S\right)}{\left(P_S \cdot L_S\right)^2 - C_{BW}\left(M + M_1\right)\left(P_S + H_L + L_S \cdot T_S\right)} \\ \frac{\left(P_S \cdot L_S\right) + 2C_{BW}{}^2 \cdot M \cdot M_1\left(P_S + H_L + L_S \cdot T_S\right)^2}{\left(P_S \cdot L_S\right) + C_{BW}{}^2 \cdot M \cdot M_1\left(P_S + H_L + L_S \cdot T_S\right)^2} \qquad (28)$$

The dependancy of the mean service time on the number of calls proceessed in a higher and a lower priority queue is depicted on Figure 4.

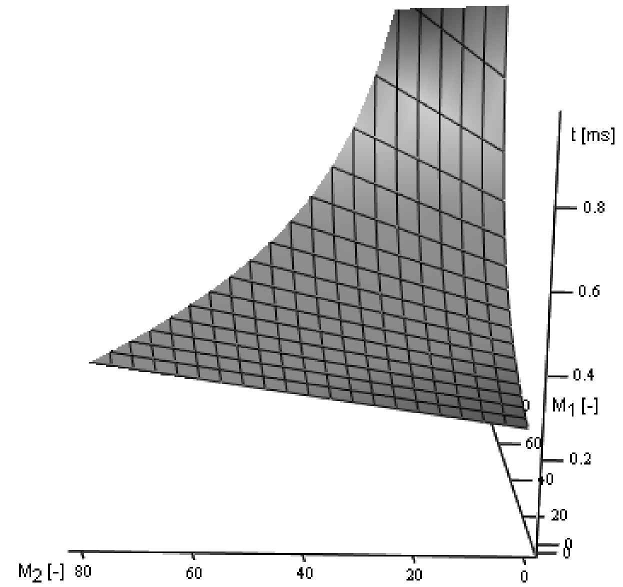

Fig. 4 Relation between the mean service time in a higher and lower priority queue and the number of calls.

End-to-end delay can be expressed by substituting the model designed for a single service queue. The end-to-end delay in a lower priority is expressed as:

$$T_{1C} = (1 + 0.1M)T_{CD} + \frac{P_S}{C_{BW}} + \frac{1}{v}\sum_{i=1}^{n} L_i + T_{DJD} + \sum_{i=2}^{n} T_{1i} \qquad (29)$$

where:
T_{1c} – end-to-End delay [s]
N – number of voice blocks in a packet [-]
T_{CD} – total delay of the codec [s]
L_i – length of line i [m]
v – speed of signal transmission in the environment [m/s]
T_{DJD} – de-jitter delay [s]
T_{1i} – mean service time i service element i [s]

End-to-end delay in a higher priority queue can by expressed by the following formula:

$$T_{2C} = (1 + 0.1M)T_{CD} + \frac{P_S}{C_{BW}} + \frac{1}{v}\sum_{i=1}^{n} L_i + T_{DJD} + \sum_{i=2}^{n} T_{2i} \tag{30}$$

IV. Experiment

The evaluation of the first M/D/1 model was performed in simple network where two Cisco routers were interconnected via serial interface and the traffic was emulate and evaulate by *IxChariot* (tool enablig to generate and evaluate RTP streams). The second model, which was created in accordance with M/D/2, was verified in router based on Linux with *Traffic Control* tool where the prirority queues were defined and we apllied Ixchariot both for generating and for evaluating the dealy.

A. Verification of proposed M/D/1 Delay Model

A test bed for the estimation of the designed model was prepared at the department of telecommunications in Ostrava and consisted of two routers interconnected by means of a serial interface with PPP Multilink. The VoIP calls were emulated by IxChariot tester which was used for endpoints and in a console mode for evaluation of the VoIP calls. IXIA IxChariot is a test tool for simulating VoIP traffic to predict device and system performance under various conditions. This tool was used for measuring and traffic simulation. The tests were performed between pairs of network - connected computers. IxChariot endpoints created the RTP streams between pairs and the results were sent to the console and analyzed. Figure 5 illustrates the situation.

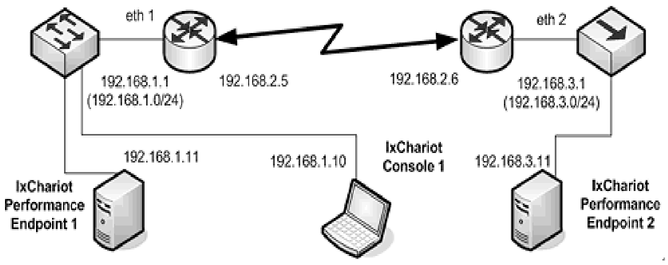

Fig. 5 Scheme of the topology used in the experiment

The configuration of the serial interface is described below. The *bandwith* value determines the bandwidth that will be used by the interface in range from 128 to 2048 Kbps.

```
interface Serial1/0
description Seriove rozhrani- DTE
bandwidth 2048
no ip address
encapsulation ppp
load-interval 30
no fair-queue
ppp multilink
ppp multilink group 1
```

The queuing mechanism used in this case was priority queuing. The highest priority queue was reserved for RTP packets with the lowest destination port 16384 and the highest port number 32767. The IP RTP Priority command shown in the following example of configuration was used to provide the highest priority to RTP packets. The last parameter is the maximum bandwidth allowed for this queue.

```
interface Multilink1
ip address 192.168.2.5 255.255.255.0
ppp multilink
ppp multilink fragment delay 20
ppp multilink interleave
ppp multilink group 1
max-reserved-bandwidth 100
ip rtp priority 16384 16383 2000
```

Other parameters such as type of codec, timing and number of the RTP sessions also had to be specified directly in the IxChariott tool. The tests ran in an environment with and without a traffic saturation which was done by a UDP generator. The tests were automatically performed by a batch file which was created for this purpose. The files stated below were used to initialise tests and the results were exported to HTML files. These files define the conditions for the performance of the tests and are executed by the following commands:

```
runtst 1024-20-1.tst
fm.exe 1024-20-1.tst 1024-20-1.tst.txt -c
fm.exe 1024-20-1.tst 1024-20-1.tst.html –h
sleep 30
```

The first line refers to the *runtst* program which runs a test that is passed as a parameter. The second line refers to the *fm* program which exports the results to a text file while the third line exports the results to an HTML file. The command *sleep 30* was inserted there because of errors in the initialization of the endpoints. Once the tests have been finished, we have identified several parameters.

The results were classified in the groups as follows: MOS, R_factor, jitter and one way delay [9]. With this we could determine average values for all measured results.

It is important to say that the measures were made at different speed (128, 256, 512, 1024 and 2048 kbps) both in an environment without saturation and with saturation. The duration of the test was set to 1 minute during which all observed parameters were recorded at one-second intervals. Every test was repeated five times in order to eliminate any aberrations. We have obtained results of more than 5000 measurements.

B. Verification of proposed M/D/2 Delay Model

The workplace in which we carried out the estimation of the proposed model consisted of a service element (PC1) with *Traffic Control*, three performance endpoints and a console workstation. VoIP calls were emulated by IxChariot Performance endpoints and the IxChariot Console was used to assess VoIP calls. Experiments were carried out under different conditions. IxChariot endpoints generate voice streams between PC2 and PC3 and between PC4 and PC3. Linux distribution *OpenSuse 10.3*, with the implemented support of the QoS was used as the operating system in the core element. Two queues to process voice streams and one queue to process the rest of the traffic have been defined. The structure of the experimental workplace is illustrated in Figure 6.

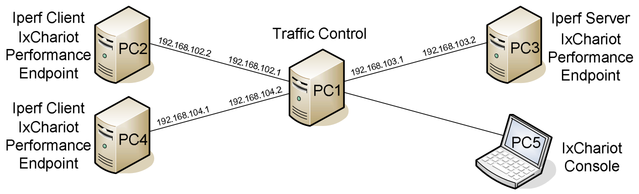

Fig. 6 Topology of the experimental workplace

The configuration of the *Traffic Control* in the core element is described below. Three priority queues were defined.

```
tc qdisc add dev eth1 root handle 1:0 prio
tc filter add dev eth1 parent 1:0 prio 1 protocol ip u32 match
ip tos 0x28 0xff flowid 1:1
tc filter add dev eth1 parent 1:0 prio 2 protocol ip u32 match
ip tos 0x48 0xff flowid 1:2
tc filter add dev eth1 parent 1:0 prio 3 protocol ip u32 match
ip tos 0x00 0xff flowid 1:3
tc qdisc add dev eth2 root handle 1:0 prio
tc filter add dev eth2 parent 1:0 prio 1 protocol ip u32 match
ip tos 0x28 0xff flowid 1:1
tc filter add dev eth2 parent 1:0 prio 2 protocol ip u32 match
ip tos 0x48 0xff flowid 1:2
tc filter add dev eth2 parent 1:0 prio 3 protocol ip u32 match
ip tos 0x00 0xff flowid 1:3
tc qdisc add dev eth3 root handle 1:0 prio
tc filter add dev eth3 parent 1:0 prio 1 protocol ip u32 match
ip tos 0x28 0xff flowid 1:1
tc filter add dev eth3 parent 1:0 prio 2 protocol ip u32 match
ip tos 0x48 0xff flowid 1:2
tc filter add dev eth3 parent 1:0 prio 3 protocol ip u32 match
ip tos 0x00 0xff flowid 1:3
```

The Network Interface Cards were configured using Ethtool. The network address was configured using standard Linux commands. An example NIC configuration for PC2 is described below.

```
ethtool –s eth1 speed 10 duplex full autoneg off
ifconfig eth1 192.168.102.2 netmask 255.255.255.0
route add default gw 192.168.102.1
```

The number of voice streams was the same. TOS 0x28 values were used in voice streams between PC2 and PC3. TOS 0x48 values were used in RTP streams between PC4 and PC3. Each RTP stream used a different communication port. For our experiment, we used G.711a and 20ms as a delay between the datagram.

The relation between the mean service time in a higher and lower priority queue and equally distributed load in the queues is shown in Figure 7.

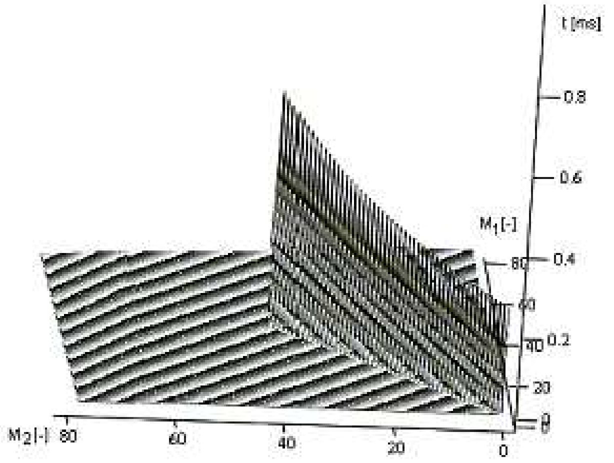

Fig. 7 Relation between the mean service time in a higher and lower priority queue and equally distributed load in the queues.

The mathematical model uses values characteristic for the G.711 codec. The length of transmission lines was set to 50 meters and the De-jitter buffer size was set to 1ms. The accuracy of the model depends on what T_S parameter is chosen. In order to compare the conformity of real and theoretical values, T_S = 0,11 ms was applied. A comparison of theoretical values and the results of the experiment is shown in the diagrams below. In experimental workplace we have been theoretically able to run approximately 110 calls. In real terms we performed only 75 calls without other influences that we were not able to reflect in the model, such as an unpredictable processing of the call and loss of the information.

V. Conclusion

The designed mathematical model works with a voice traffic approximation supported by a traffic source with Poisson's probability distribution. The described way does not exactly imitate real characteristics of voice traffic, in particular a certain tendency to form clusters. Therefore it was assumed that with the increasing line load the mathematical model will not return absolutely exact information.

In the first case of the M/D/1 model, the measurements showed that in most cases the designed mathematical model returns data with ±6 % accuracy up to the 80 % line load. With the increasing number of simultaneous calls and with the decreasing line load the accuracy of gained data increases. Even though individual voice flows do not match the model of signal source with the Poisson's probability distribution, their sum approximates to this model, in particular with the growing number of calls. Where 10 simultaneous calls do not load the output line by more than 40 %, the exactness of the model reaches ±1.5 %. As most of designed VoIP networks operate with a much higher number of simultaneous connections, we can assume that the model will return sufficiently exact assessment of an average delay in the network.

In the second case of the M/D/2 model, the measurements have shown that the mathematical model strongly depends on selection of T_S value. T_S has proved to be significant between 50 % and 70 % of the line load, due of the emergence of the processing delay. Because of the use of the processing time in the mathematical model, we are able to get data with accuracy below ± 3 % up to the 70 % of line load. Furthermore (over 70% of line load), the tests did not reproduce due to the unpredictable behaviour of call processing and loss. T_S is a key parameter is the time it takes to process a service element. This parameter needs to be determined individually for each service element. It is determined by hardware (processor, motherboard and network card, etc.) and software (operating system, kernel, etc.) used. The only option to determine the processing time is based on knowledge of the behaviour characteristic of the element in the increasing load. Up to the 70 % of line load, the maximum deviation between the theoretical model and real values was 0,75 ms. The delay incurred in the queuing element with delay below 1ms can not be considered as sufficiently precise since the absolute measurement error of the method using IxChariot equals 1ms. As regards the end-to-end delay, the relative error measured during the experiment is less than 3 % when compared to the theoretical values obtained through the application of the mathematical model.

If we apply these models to describe VoIP networks that process a greater number of simultaneous voice connections, we can assume that the proposed models will return sufficiently exact assessment of an average delay in the network.

Acknowledgment

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 218086.

References

[1] M. Voznak, *Voice over IP*, VSB-Technical University of Ostrava, 1st ed., College book, Ostrava, 2008.

[2] A. E. Mahdi and D. Picovici, ”Advances in voice quality measurement in modern telecommunications, ” *Digital Signal Processing*, Volume 19, Issue 1, January 2009, pp.79-103.

[3] K. Fujimoto and S. Ata, M. Murata, ”Adaptive playout buffer algorithm for enhancing perceived quality of streaming applications, ” *Telecommunication Systems,* vol . 25, Issue (3-4): 259-271, 2004.

[4] I. Pravda and J. Vodrazka, ”Voice quality planning for NGN including mobile networks, ” *In the 12th International Conference on Personal Wireless Communications (PWC 2007)*, 2007, Prague.

[5] D. Gross and C. Harris, *Fundamentals of Queuing Theory*. New York: John Wiley & Sons, 1998.

[6] M. Voznak, F. Rezac and M. Halas, ”Speech Quality Evaluation in IPsec Environment, ” *WSEAS 12th International Conference on Networking, VLSI and Signal Processing (ICNVS '10)*, FEB 20-22, 2010, Cambridge, England, pp. 49-53.

[7] M. Halas, B. Kyrbashov and M. Voznak, ”Factors influencing voice quality in VoIP technology, ” *In 9th International Conference on Informatics,* 2007, pp. 32-35, Bratislava, 2007, p. 32-35.

[8] M. Voznak, ”Speech bandwith requirements in IPsec and TLS environment, ” *In the 13th WSEAS International Conference on Computers,* JUL 23-25, 2009 Rhodes, Greece, pp.217-220.

[9] M. Kavacky, E. Chromy, L. Krulikovska and P. Pavlovic, ” Quality of Service Issues for Multiservice IP Networks, ” *In: SIGMAP 2009 International Conference on Signal Processing and Multimedia Applications*, Milan, Italy, 7-10 July, 2009, pp. 185 – 188.

[10] V. Novotny and D. Komosny, ”Large- Scale RTCP Feedback Optimization, ” *Journal of Networks*, 2008, roč. 2008, č. 3, s. 1-10. ISSN: 1796- 2056.

[11] S. Bergida and Y. Havitt, ”Analysis of shared memory priority queues with two discard levels. ” *In IEEE Israel Conference,* 2006, p. 42-46. ISBN 1-4244-0230-1.

[12] V. Marianov and D. Serra, ”Location Models for Airline Hubs Behaving as M/D/c Queues, ” Univ. Pompeu Fabra, Economics and Business Working Paper No. 453. Available online: doi:10.2139/ssrn.224595

[13] I. Baronak I and M. Halas M, ”Mathematical representation of VoIP connection delay, ” *RADIOENGINEERING* , Volume: 16 Issue: 3 Pages: 77-85 , SEP 2007.

Miroslav Voznak holds position as an associate professor with Department of Telecommunications, VSB - Technical University of Ostrava, Czech Republic. He received his M.S. and Ph.D. degrees in telecommunications, dissertation thesis “Voice traffic optimization with regard to speech quality in network with VoIP technology” from the Technical University of Ostrava, in 1995 and 2002, respectively. Topics of his research interests are the next generation network, IP telephony, speech quality and network security.

Michal HALAS graduated from Slovak University of Technology and received the electronic engineering degree in 2003. Since this year he started postgraduate study at Department of Telecommunications STU Bratislava and in 2006 he received his PhD from Slovak University of Technology. Nowadays he works as a lecturer in Department of Telecommunications of FEI STU in Bratislava and topics of his research interests are speech quality, next generation networks and IP telephony.

Bohdan Borowik is an assistant professor with Electrical and Automation Department, Technical-University of Bielsko-Biała, Poland. He is a graduate of the Institute of Electronics in Cleveland. He underwent doctoral studies in the Department of Electronics, Automation and Computer Science. His doctoral dissertation is concerned with the: Algorithms of parallel programming applied for control problems. Topics of his research interests are wireless sensor networks and microprocesor systems.

Zbynek Kocur received the M.S. degree in electrical engineering from the Czech Technical University in Prague in 2008. Since 2008 has been studying Ph.D. degree at the same university. He gives lectures on Communication in data networks and Networking technologies. His research is focused on wireless transmission and data flow analysis, simulation and optimization.