

# A model for Endpoint Admission Control Based on Packet Loss

Ignacio Más\* and Gunnar Karlsson

ACCESS Linneaus Center  
School of Electrical Engineering  
KTH, Royal Institute of Technology  
10044 Stockholm, Sweden  
{nacho,gk}@kth.se

**Abstract.** Endpoint admission control solutions, based on probing a transmission path, have been proposed to meet quality requirements of audio–visual applications with little support from routers. In this paper we present a mathematical analysis of a probe–based admission control solution, where flows are accepted or rejected based on the packet–loss statistics in the probe stream. The analysis relates both system performance to the design parameters, and the experienced probe packet loss probability to the packet loss probability of accepted flows. The goal is to provide a simple mathematical method to perform network dimensioning for admission control based on end–to–end probing.

## 1 Introduction

Applications that produce traffic in the Internet can be broadly classified into elastic or inelastic. Elastic flows appear mostly as a result of the transfer of digital documents, like files, web pages or multimedia downloading for local storage. These flows use TCP as transport protocol. Inelastic flows appear nowadays mainly from the streaming of audio and video over UDP without congestion control functionality.

The sudden increase of video over IP on the Internet requires a better and more predictable service quality than what is possible with the available best–effort service. Audio–visual applications can handle limited packet loss and delay variation without affecting the perceived quality. Interactive communication, in addition, requires stringent end–to–end delay requirements. For example, IP telephony requires that a maximum of 150 ms one–way delay should be maintained during the whole call.

The DiffServ architecture is an attempt to meet the distinct quality of service requirements of elastic and inelastic traffic, but it still lacks an important function: admission control. Without admissions control, inelastic flows can overload the network to a point where both elastic and inelastic flows suffer intolerable packet loss. Admission control has been proposed to support applications with

---

\* Corresponding author

quality of service (QoS) requirements by limiting the network load. These proposals aim at providing QoS with very little or no support in the routers. They also share a common idea of endpoint admission control: A host sends probe packets before starting a new flow and decides about the flow admission based on statistics of the probe packet loss [1, 2], or delay and delay variation [3]. The admission decision is thus moved to the hosts and is made for the entire path from the source to the destination, rather than on a per-hop basis. Consequently, the service does not require any explicit support from the routers other than one of the various scheduling mechanisms supplied by DiffServ, and the mechanism of dropping or marking packets.

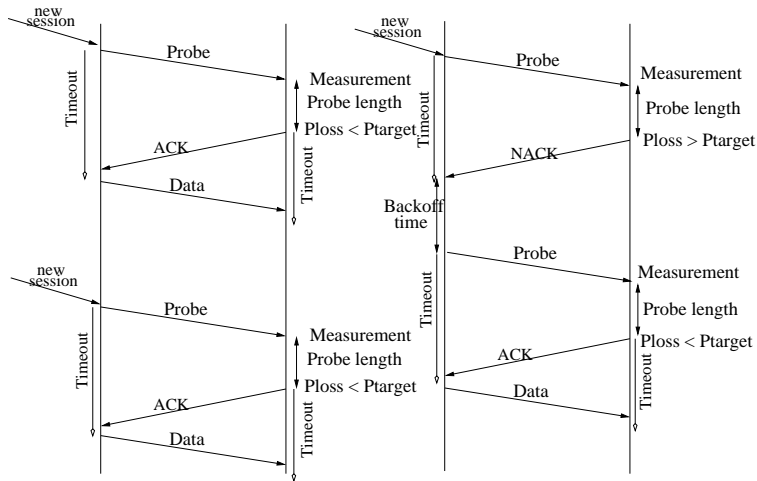
In this paper we provide a mathematical analysis of the probe-based admission control (PBAC) scheme proposed in [1, 2]. In this scheme, the admission control is based on the measured packet-loss ratio of the probe stream. The aim of the admission control is to provide a reliable upper bound on the packet loss probability for accepted flows. The end-to-end delay and delay jitter is limited by the use of small buffers inside the network. The goal of the mathematical analysis is to relate performance parameters, such as probe and data packet loss probabilities, flow acceptance probability and network utilization to system parameters, such as buffer size, probe length and admission threshold.

The paper is organized as follows: In Section 2 we provide a short description of the PBAC solution, while Section 3 presents an approximate analytical model to calculate probe and data loss probabilities. Section 4 gives a performance evaluation of the system as well as the validation of the analytical model, while Section 5 shows an example of the service dimensioning for an access operator. Finally, we conclude our work in Section 6.

## 2 Probe-Based Admission Control

QoS provisioning with probe based admission control is based on the following main ideas: i) Network nodes have short buffers for data packets of accepted flows to limit end-to-end delay and delay jitter; ii) all hosts in the network perform admission control; iii) the admission control is responsible for limiting the packet loss of accepted flows; iv) the admission decision is based on the packet loss ratio in the probe stream, thus flows are accepted if the estimated packet loss probability is below a given acceptance threshold; v) probe packets are transmitted with low priority at the routers to ensure that probe streams do not disturb accepted flows; and vi) best effort traffic utilizes the remaining available capacity of the network link at a lower priority than the probe packets.

To provide priority queuing for probe and data packets at the network nodes we consider a double-queue solution in this paper: One buffer is dedicated for high priority data packets and the other for low priority probe packets. The size of the high priority buffer for the data packets is selected to ensure a low maximum queuing delay and an acceptable packet loss probability, i.e., to provide packet scale buffering [4]. The buffer for the probe packets can accommodate few probe packets at a time, to ensure an over-estimation of the data packet loss. Similar



**Fig. 1.** The probing procedure

priority queuing can be achieved using a single buffer with a discard threshold for the probe packets. The two solutions are compared in [5]. The threshold-queue is not included in the mathematical analysis of this paper.

The acceptance threshold is fixed for the admission controlled traffic service class and it is the same for all flows. The reason for this is that the QoS experienced by a flow is a function of the load from the flows already accepted in the class. Considering that this load depends on the highest acceptance threshold among all flows, by having different thresholds all flows would be degraded to the QoS required by the one with the least stringent requirements. The class definition has also to state the maximum data rate allowed to limit the size of the flows that can be set up. Each data flow should not represent more than a small fraction of the service class capacity, to ensure that statistical multiplexing works well. An study of the effect of ‘thicker’ flows in the admission control scheme (over 10% of service class capacity) can be found in [5]

In Fig. 1 we can see the phases of the probing procedure. When a host wishes to set up a new flow, it starts sending a constant bit rate probe to the destination host at the maximum rate the flow will require ( $r_{pr}$ ). The probe packet size ( $s_p$ ) should be small to have a high number of packets in the probing period ( $T_p$ ) in order to perform the acceptance decision with a sufficient level of confidence. The probe packets contain information about the peak bit rate and length of the probe, as well as a sequence number. With this information the receiving host can perform a quick rejection, based on the expected number of packets that it should receive in order not to surpass the target loss probability. The probe also needs to contain a flow identifier to allow the end host to distinguish probes for different flows, since one sender could transmit more than one flow simultaneously. The IP address in the probes would consequently not be enough to differentiate them.

**Table 1.** Main parameters of the mathematical analysis.

Static parameters:	
$C_l$	Link capacity [b/s]
$Q_p$	Probe queue length [p]
$Q_d$	Data queue length [p]
$\theta$	Loss threshold
Probe session parameters:	
$\Lambda_p$	Probe session mean arrival rate [1/s]
$T_p$	Probe session length [s]
$I_p = \Lambda_p T_p$	Probe session traffic intensity
Probe parameters:	
$s_p$	Packet size [b]
$r_p$	Bit rate [b/s]
$\mu_p = \frac{C_l}{s_p}$	Mean service rate [p/s]
$\lambda_p = \frac{I_p r_p}{s_p}$	Arrival rate [p/s]
$\rho_p = \frac{\lambda_p}{\mu_p}$	Traffic intensity
Data parameters:	
$s_d$	Packet size [b]
$\gamma_d$	mean-to-peak ratio
$r_d = r_p \gamma_d$	Average bit rate [b/s]
$T_d$	Data session length [s]
$\mu_d = \frac{C_l}{s_d}$	Mean service rate [p/s]

To perform the acceptance decision, the end-host measures the empirical probe loss rate,  $P_{me}$  in the probe stream. Assuming a normal distribution of the probe loss [5], the flow is accepted, if:

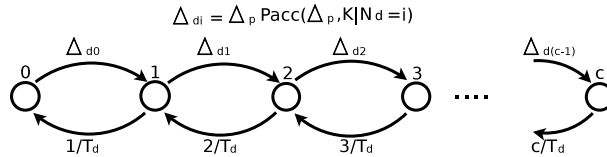
$$p_{me} + z_R \sqrt{\frac{p_{me} \times (1 - p_{me})}{s}} \leq \theta, \text{ given that } \theta \times s > 10, \quad (1)$$

where  $\theta$  is the acceptance threshold,  $s$  is the number of probe packets sent,  $R$  is the confidence level we want to have and  $z_R$  is the  $1 - (1 - R/2)$ -quantile of the normal distribution. The second condition ensures that we have sufficient number of samples for the estimation.

When the same host or a new one wants to set up a new flow, it repeats the procedure with the maximum rate of the new flow and receives an independent acceptance decision. In the case in which the empirical probe packet loss is greater than the upper level of the confidence interval for the admission threshold, then session is rejected and has to back-off for a certain time before trying again. More details of the procedure can be obtained from [2].

### 3 Mathematical model

The approximate analytical model presented in this section determines the performance of the PBAC system depending on the main system parameters such



**Fig. 2.** Birth-death Markov chain for accepted sessions

as the queue size for the high priority data packets ( $Q_d$ ) and for the low priority probe packets ( $Q_p$ ), the acceptance packet loss threshold for new flows ( $\theta$ ), the length of the probe ( $T_p$ ), the probe rate ( $r_p$ ) and the probe packet size ( $s_p$ ). Table 1 summarizes the different parameters used in the analysis, where capital letters refer to session level parameters, and lower letters to packet level parameters, and subscripts  $p$  and  $d$  refer to probe and data packets or sessions.

Our mathematical analysis uses the probe session mean arrival rate ( $\Lambda_p$ ) and the size of the two queues as the independent variables, while the acceptance probability and data packet loss are the values we wish to obtain. The rest of the parameters are treated as constant in the analysis.

### 3.1 Computation of maximum non-blocking probe load:

The probe queue is a  $n^*D/D/1$  queue, which can be upper bounded by an  $M/D/1$ . The  $M/D/1$  queue in turn is strictly shorter for the same arrival rate than an  $M/M/1$  queue with the same average service time [4], so we can consider the queue as an  $M/M/1/Q_p$ . This approximation will have an effect in the closeness of our analytical results with the simulation and experimental results, as will be seen in the evaluation section. We are as well assuming independence of packet arrivals, which can be justified under intense traffic (i.e. close to the saturation point, where we are interested in obtaining performance values). The packet loss probability is thus given by [6]:

$$P_{loss}(\rho_p, Q_p) = \frac{(1 - \rho_p)\rho_p^{Q_p}}{1 - \rho_p^{Q_p+1}} \quad (2)$$

From this equation we look for the traffic intensity in the probe queue  $\rho_p^*$ , such that the loss in the queue is less or equal than the threshold:

$$\rho_p^* | P_{loss}(\rho_p^*, Q_p) \leq \theta \quad (3)$$

This non-blocking load needs to be translated into number of concurrent probing sessions ( $N_p^*$ ), as follows:

$$N_p^* = \lfloor \frac{\rho_p^* C_p}{r_p} \rfloor, \quad (4)$$

where  $C_p = C - (N_d r_d)$  is the capacity left for the probe packets, being  $N_d$  the number of ongoing data sessions. This capacity left for the probe packets is approximate since it does not take into account the packet loss for the ongoing data sessions. We are thus assuming independence of the two queues, which has a small impact on the accuracy of the results, as seen in the evaluation section. For a given number of accepted data sessions, we can then obtain the maximum number of simultaneous probing sessions  $N_p$ , and then we can compute the probability of having up to that number of probing sessions. Assuming that new sessions arrived according to a Poisson process, with intensity  $\Lambda_p$ , the probability of having  $n$  concurrently probing sessions is obtained as:

$$p_p(n) = \frac{I_p^n e^{-I_p}}{n!}, \quad (5)$$

where  $I_p$  represents the probe session traffic intensity. With this probability, we can already compute the probability of acceptance conditioned to having  $N_d$  ongoing data sessions, as:

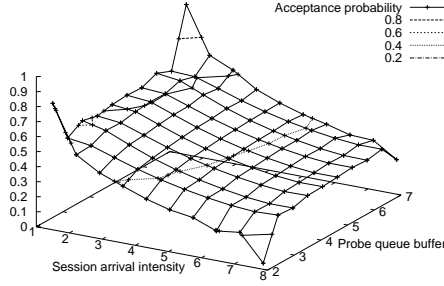
$$P_{acc}(Q_p, \Lambda_p | N_d) = \sum_{n=0}^{N_p^*} p_p(n) \quad (6)$$

This equation gives us the acceptance probability as a function of the probe session arrival rate and the probe packet queue for a given number of ongoing data sessions. To obtain the unconditioned acceptance probability, we need to sum all the conditional probabilities for every  $N_d$  multiplied by the probability of having  $N_d$  ongoing data sessions in the system. The last stochastic variable can be model as a Markov-modulated Poisson process with  $c$  states, where the arrival rates are state dependent. Figure 2 shows the birth-death Markov chain. The number of servers ( $c$ ) corresponds to the highest value of  $N_d$  that gives  $P_{acc}(Q_p, \Lambda_p | c) > 0$ . The M/G/c/c system is insensitive to the service time distribution and satisfies the M/M/c/c birth-death process also for state dependent arrivals. In this way, the steady state probabilities of an M/M/c/c birth-death process can be obtained as [6]:

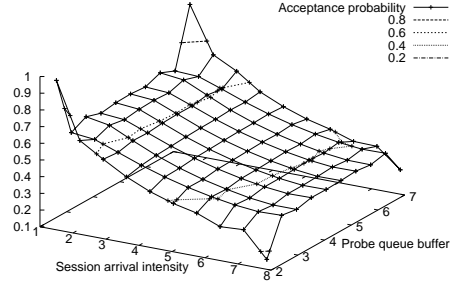
$$p_d(N_d = i) = p_0 (T_d)^i \frac{\prod_{i=0}^{i-1} P_{acc}(Q_p, \Lambda_p | N_d = i)}{c!}, \quad (7)$$

with:

$$p_0 = p_d(N_d = 0) = (1 + \sum_{i=1}^c [(\frac{\lambda}{\mu_d})^i \frac{\prod_{j=0}^{i-1} P_{acc}(Q_p, \Lambda_p | N_d = j)}{c!}])^{-1} \quad (8)$$



**Fig. 3.** Acceptance probability for  $\theta=0.01$



**Fig. 4.** Acceptance probability for  $\theta=0.05$

With these steady state probabilities we can already compute the unconditioned acceptance probability as:

$$P_{acc}(Q_p, A_p) = \sum_{\forall N_d | P_{acc}(Q_p, A_p | N_d) \neq 0} (P_{acc}(Q_p, A_p | N_d) p_d(N_d)) \quad (9)$$

Figures 3 and 4 show the acceptance probability as a function of the probe queue size and the offered load to the system. The probe queue size varies from 1 to 8 positions and the offered load varies from 0.64 to 3.34 (which corresponds to session arrival rates  $A_p$  from 1.5 to 8 roughly), while the acceptance threshold is 0.01 and 0.05 probe packet loss ratio. The surface lines mark the points where the acceptance probability is 0.8, 0.6, 0.4 and 0.2 respectively.

To compute the packet loss that ongoing data sessions will experience, we model the data queue as an M/M/1/ $Q_d$ , where  $Q_d$  is the data packets queue length<sup>1</sup>. To obtain the traffic to substitute in Eq. (2), we use the obtained steady state probabilities. On average, for a given  $Q_p$  and  $A_p$ , we will have  $E[N_d] = \sum_{i=1}^c i p_d(N_d = i)$  ongoing sessions transmitting data packets. The mean data packet arrival rate will be  $\lambda_d = \frac{E[N_d] r_d}{s_d}$ , which then gives  $\rho_d = \lambda_d T_d$ . Substituting the values of  $Q_d$  and  $\rho_d$  in Eq. (2), we obtain the average data packet loss as:

$$P_{data\_loss}(\rho_d, Q_d) = \frac{(1 - \rho_d) \rho_d^{Q_d}}{1 - \rho_d^{Q_d+1}} \quad (10)$$

Figures 5 and 6 show the data packet loss ratio with the same parameters as the acceptance probabilities figures. The data packet (high priority) queue was 20 packets in both figures.

Finally, Figures 7 and 8 illustrate what would be the achieved utilization as a function of  $A_p$  and  $Q_p$ . The utilization increases as the offered load increases up to a point slightly over 0.8 of link capacity for an offered load of  $A_p \approx 1.9$ ,

<sup>1</sup> A more sophisticated queuing model could be used for the aggregate traffic of the  $N_d$  sessions.

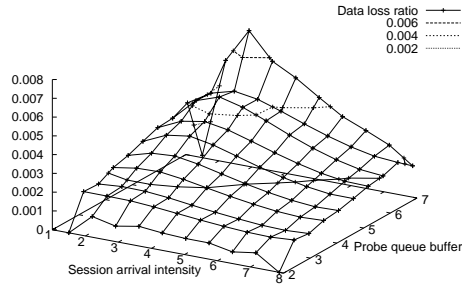


Fig. 5. Data packet loss  $\theta=0.01$

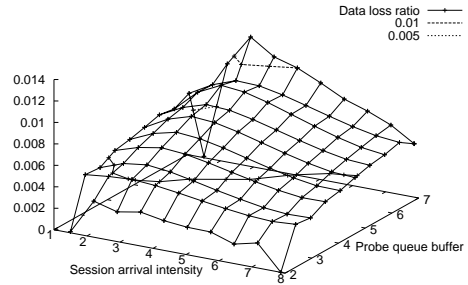


Fig. 6. Data packet loss for  $\theta=0.05$

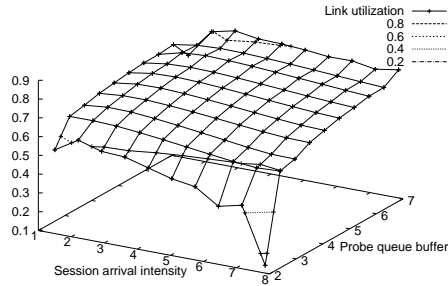


Fig. 7. Achieved utilization for  $\theta=0.01$

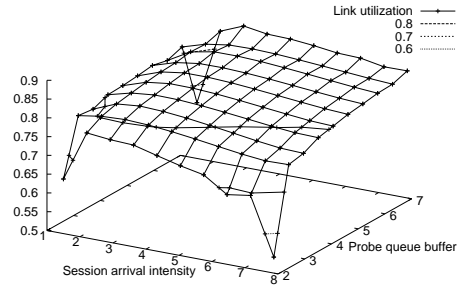


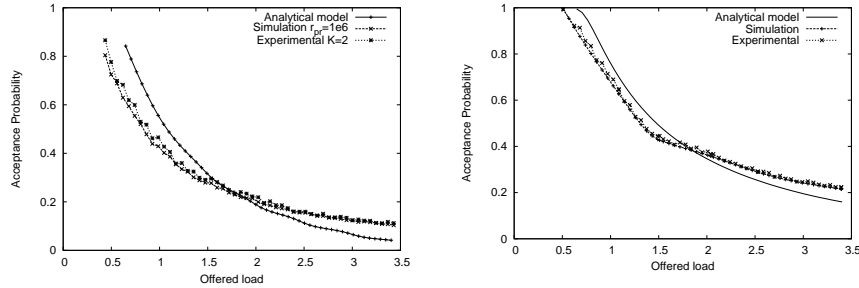
Fig. 8. Achieved utilization for  $\theta=0.05$

from where the utilization starts decreasing, due mainly to the probe thrashing effect studied in [2, 5]. This effect happens when the arrival intensity of probe packets is so high that they collide in the probe queue and are dropped even if the system has available capacity to admit new flows. This behavior then increases the blocking probability of concurrent sessions, potentially rejecting sessions that should be accepted. It's important to notice that we have not modeled a back-off mechanism in our arrival rate which is constant even when the acceptance probability decreases. In a real scenario, peaks of offered load over 1 should fade out as the blocking increases and sessions back off for longer periods of time.

## 4 Evaluation

We have performed simulations with the network simulator NS-2 to validate our analytical model. The simulations contained sources with exponentially distributed on-off times, and peak rates ( $r_p$ ) of 100 kb/s over a 10 Mb/s link, or 1 Mb/s over a 100 Mb/s link. The on-off holding times had an average of 20





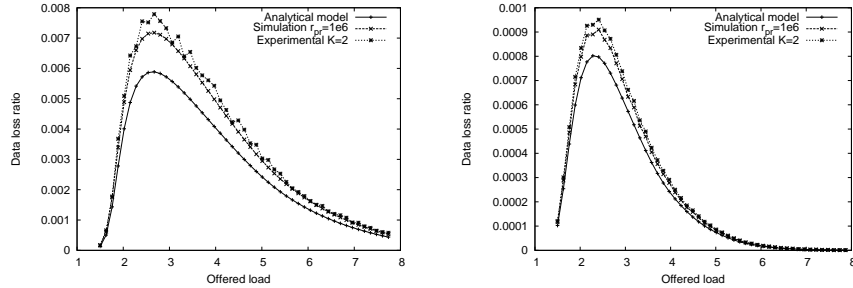
**Fig. 9.** Acceptance probability for 100 kb/s call peak rate, with an acceptance threshold of  $5 \times 10^{-2}$ . **Fig. 10.** Acceptance probability for 1 Mb/s call peak rate, with an acceptance threshold of  $10^{-2}$ .

and 35.5 ms respectively. Packet sizes were 64 bytes for the probe packets and 128 bytes for the data packets, while the probe length was always 2 seconds. We used a confidence interval for the admission decision of 95%. The simulation time was 2000 seconds and the average holding time for accepted flows ( $1/\mu_d$ ) was 120 seconds. We have discarded the first 500 seconds of simulation to let the simulation achieve a steady state, since the utilization of the link stabilizes after 150 seconds with fluctuations of  $\pm 2$  accepted flows. Confidence intervals for each simulation are too small to be plotted in the figures. The probe queue size ( $Q_p$ ) and the flow arrival rate ( $\lambda_p$ ) varied in each simulation to increase the offered load to the system. The queue used was a double queue with a low priority buffer of two packets in the results shown and a high priority buffer of 20 packets. To prove the behavior of our probing scheme we have used a simple one bottleneck link topology. The results for a multilink scenario obtained in [1] show that the highest loaded link dominates the behavior and that the scheme discriminates against flows with longer paths.

We have as well used an experimental prototype to validate the simulations in our lab environment. All the experimental results were generated by using a Spirent SmartBits 6000B traffic generator to generate the background traffic with the same characteristics as the traffic in the simulations. The probing sources have been generated by using a simple software UDP traffic generator<sup>2</sup>, to which we have added the needed probing functions from the PBAC library. The topology used in the laboratory implements that of the simulations with one bottleneck link. The routers are PC's running Debian Linux with a 2.6.9 kernel providing the special double queue system, with 100 Mb/s Ethernet interfaces.

Figure 9 and Figure 10 show the acceptance probabilities of a new call with an acceptance threshold of 0.05 and 0.01, with call peak rates of 1 Mb/s. In both cases the analytical curve intersects the curves obtained by simulation and in the experimental evaluation. The separation between the analytical curve and the simulation and experimental results comes from the assumptions taken

<sup>2</sup> <http://rude.sourceforge.net>



**Fig. 11.** Data packet loss for 1 Mb/s call **Fig. 12.** Data packet loss ratio for 1 Mb/s peak rate, with an acceptance threshold call peak rate, with an acceptance threshold of  $5 \times 10^{-2}$ . old of  $10^{-2}$ .

in the mathematical model: The queue of the probes has been modeled as an  $M/M/1/Q_p$ , while it should be a  $nD/D/1/Q_p$ , since we have constant bit rate probes and fixed probe packet sizes. The  $D/D/1/Q_p$  model should offer a closer curve to the acceptance probability of experimental and simulation results, but it requires a numerical procedure to compute the packet loss that was out of the scope of our simple analysis.

Figures 11 and 12 finally show the data packet loss experienced by accepted sessions as a function of the offered load. In these figures it can clearly be seen the effect of the probe thrashing effect previously mentioned. The packet loss ratio increases as the offered load increases up to a value of  $2.5 A_p$ , which corresponds to a link capacity of approximately 1.1 percent. After that point, sessions begin to be rejected due to probe thrashing and the blocking probability increases heavily, so as less sessions are admitted in the system, the packet loss ratio decreases sharply. It is interesting to observe the slightly higher values of packet loss ratios in the simulation and experimental results, which can be explained by the fact that we have modeled the ongoing sessions as an  $M/M/1/Q_d$ , while the simulation and experimental sources are on-off sources with exponentially distributed on period. The on-off sources are thus more bursty in nature and that produces the higher packet loss. A better model would be to use an MMPP for the ongoing sources as well.

## 5 A dimensioning example

As a practical example of how to dimension the service in an access operator, we consider a 100 Mb/s Ethernet access network. The access operator is offering a video-conference service to the users in the access network itself. The maximum end to end distance between users is four hops, i.e. there will be a maximum of four routers/switches between the communicating users. The access operator wants to have an upper loss threshold of 1% packet loss, since the video-conference media carries FEC that can correct up to that loss ratio. If we consider a maximum distance of 5 km in Ethernet cable between users (in

several connected Ethernet links), and a propagation speed of  $2 \times 10^8$  m/s, the propagation end-to-end delay becomes 0.025 ms.

The video-conference service uses a packet size of 400 bytes and has a peak rate of 1 Mb/s, while the average rate is 256 kb/s. A 100 Mb/s Ethernet implies a service time of 0.032 ms of service time per packet in every hop. The total end-to-end delay can then be approximately computed as:

$$Tot\_delay = Prop\_delay + Queueing\_delay = 0.025ms + 4Q_d0.032ms \quad (11)$$

To keep an end-to-end delay under 150 ms the data queue size has to be smaller than 1169 packets, which shows the fact that end-to-end queuing delay does not have a big role when the number of hops is small. As an example, if the number of hops increases to 10, then the maximum data queue size decreases to 468 packets. Of course this is a simplification in which we have not considered the processing time in every node, only the queuing delay per node.

A 400 bytes packet sent at a 1 Mb/s peak rate gives 312 packets/s. Since we are trying to limit the packet loss to under 1% this ensure that we get enough samples of probe packets with just a few seconds. The operator decides that it does not want to have a longer call setup time than 10 secs, so we limit the probing phase to 9.5 seconds. With this probing phase, our admission threshold can be computed using Eq. 1, where the  $s$  is 2964 packets,  $R$  is 0.95 and  $p_{me}$  is the maximum packet loss we accept, i.e. 0.01. With this values the admission threshold becomes 0.00842, or roughly 0.84% probe packet loss. If we were to reduce the probing phase to only one second, with the same confidence interval and target loss,  $\theta$  would then become 0.00592 instead.

The access operator has a statistical study that predicts that it will be receiving between 0.1 to 1.5 calls a second ( $A_p$ ) and every call on average is maintained during 5 minutes. With the average rate of 256 kb/s that gives an offered load of 7.68% to 115% in the 100 Mb/s links. With these values, equation 9 gives us the unconditional acceptance probability for different values of the probe queue ( $Q_p$ ). For a probe queue of 2 packets, the acceptance probability is 1 until the offered load equals 0.52 of link capacity, while it is 0.46 for an offered load of 1.5 calls per second. Increasing  $Q_p$  to 5 packets would move the 1% acceptance probability level to 0.63 of link capacity, while the acceptance probability for 1.5 calls per second becomes 0.73.

The packet loss of accepted sessions for this two extremes of our call session arrival interval can be computed with Eq. (10). With a data packet buffer ( $Q_d$ ) of 20 packets, for example, there is no packet loss for both values of  $Q_p$  and an offered load of 0.1 calls per second, while for 1.5 calls per second the loss becomes  $2 \times 10^{-6}$  for  $Q_p = 2$  and  $6.15 \times 10^{-3}$  for  $Q_p = 5$ . Both values are well under our target of 1% loss ratio, though it is interesting to notice that increasing the probe queue size increases the acceptance probability of incoming sessions as well as the loss ratio operational point for accepted sessions.

Finally, a look at the utilization of the link capacity gives the following numbers: For  $Q_p = 2$  and  $\lambda_p = 0.1$  the utilization of the link is roughly the offered

load, since acceptance probability is 1. The maximum utilization is achieved for an offered load of 0.82 of link capacity, while the utilization at 1.15 drops to 0.53. If we take a probe buffer size of  $Q_p = 5$  instead, then the maximum utilization is 0.85 and occurs for an offered load of 1.1 calls per second.

## 6 Conclusions

In this paper we have presented an approximate analytical model of the probe based admission control scheme based on the end-to-end measurements of packet loss probabilities in the probe streams. In this solution the admission control is responsible for limiting the end-to-end packet loss probability of accepted flows, while the end-to-end delay and delay jitter requirements are ensured by the use of small buffers in the routers.

The analysis focuses on the flow acceptance probabilities at a given offered load, the link utilization achieved, and on the packet loss probabilities of accepted flows. The analytical results, verified by simulations, prove that the considered probe-based admission control leads to a stable link utilization which has a clear upper bound on the packet loss probability.

As the acceptance decision is based on the probe loss probability it is important to see how the loss admission threshold and data loss probabilities relate at different link utilization. The analysis predicts that the data loss is always under half an order of magnitude of the targeted loss threshold, independently of the actual link load, which is confirmed in our experimental results.

Consequently, the probe-based admission control based on measurements of the packet loss probabilities in the probe stream provides a reliable and efficient solution for QoS provisioning for delay and loss sensitive applications, without extensive support in the routers.

## References

1. V. Fodor (née Elek), G. Karlsson, and R. Rönngren, "Admission control based on end-to-end measurements," in *Proc. of the 19th Infocom*, (Tel Aviv, Israel), pp. 623–630, IEEE, March 2000.
2. I. Más and G. Karlsson, "Probe-based admission control for a differentiated-services internet," *Elsevier Computer Networks*, vol. 51, pp. 3902–3918, September 2007.
3. G. Bianchi, F. Borgonovo, A. Capone, and C. Petrioli, "Endpoint admission control with delay variation measurements for QoS in IP networks," *ACM Computer Communication Review*, vol. 32, pp. 61–69, April 2002.
4. J. W. Roberts, U. Mocci, and J. Virtamo, eds., *Broadband Network Teletraffic – Final Report of Action COST 242*, vol. 1155 of *LNCS*. Springer, 1996.
5. I. Más and G. Karlsson, "PBAC: Probe-based admission control," in *Proc. of QoFIS 2001*, vol. 2156 of *LNCS*, (Coimbra, Portugal), pp. 97–109, Springer, September 2001.
6. R. Jain, *The art of computer systems performance analysis*. Wiley Professional computing, John Wiley & Sons, 1991.