# Packet management techniques for measurement based end-to-end admission control in IP networks

*Giuseppe Bianchi*

Università di Palermo
Dipartimento di Ingegneria Elettrica
Viale Delle Scienze, Parco D'Orleans
90128 Palermo, Italy
tel:+39.091.6566247 fax:+39.091.488452
email:*bianchi@elet.polimi.it*

*Chiara Petrioli*

Politecnico di Milano
Dipartimento di Elettronica e Informazione
Piazza L. Da Vinci 33
20133 Milano, Italy
tel:+39.02.23993611 fax:+39.02.23993413
email:{*petrioli*}*@elet.polimi.it*

**Abstract**

End-to-end Measurement Based connection Admission Control (EMBAC) mechanisms have been proposed to support real-time flows quality of service requirements over a Differentiated Services Internet architecture. The EMBAC basic idea is to decentralize the admission control decision, by requiring each individual user to probe the network path during flow setup, and by basing the accept/reject decision on the probing traffic statistics measured at the destination. In conformance with the differentiated services framework, routers are oblivious to individual flows and only need to serve data packets with a higher priority than probing traffic.

In this paper we build upon the observation that a tight form of congestion control of the probing packets queue at each router is necessary to provide performance effective EMBAC operation. The original contribution of the paper is threefold. First, we provide a thorough investigation, by means of both approximate analytical modeling and extensive simulation, of an EMBAC scheme (denoted in the following EMBAC-PD), in which probing queues congestion control is based on the enforcement of a probing packet expiration deadline at each router. Second, we prove that EMBAC-PD can efficiently operate even in the presence of very light probing load (few probing packets per flow setup). Third, and most important for commercial purposes, we show that EMBAC-PD does not necessarily require long probing phases (to accurately estimate the network load subject to statistical fluctuations), but can provide effective operation even in the presence of extremely short (tens of ms) probing phase duration.

# 1 Introduction

It is widely accepted that the best effort model of the today Internet is not able to satisfactorily support the emerging market demand for real-time audio and video services. These services may require very stringent Quality of Service requirements (e.g. 150 ms mouth to ear delay for toll quality IP telephony), that have to be maintained for all the call holding time. The goal of achieving such a tight QoS control on the Internet, meanwhile leaving untouched its fundamental architectural principles, is an open research issue. The question is whether an Internet stateless and scalable architecture, such as that envisioned by the Differentiated Services framework [1, 2], has the intrinsic capability to provide performance comparable to that achievable by heavyweight per-flow resource management approaches, such as Integrated Services/RSVP [3, 4].

Recent literature [5, 6, 7, 8, 9, 10] has envisioned the possibility to introduce in IP networks a purely distributed per flow admission control scheme, which does not rely on any state information held into the core routers. The enabling idea is to allow each pair of connection endpoints to probe the network before setting up the connection. The goal of this probing phase is to estimate whether the considered connection can be accepted with a predetermined QoS, and without degrading the QoS already established for accepted calls. In what follows, we will refer to solutions based on the described operation with the descriptive name End-to-end Measurement Based Admission Control (EMBAC) mechanisms.

The fundamental differences with classical MBAC mechanisms proposed in the literature [11, 12, 13] are two. First, the decision on whether to accept or reject a connection is not taken by the internal network routers, which are oblivious to individual flows, but it is independently taken by the edge nodes of each specific connection. Second, measures are taken end-to-end by each connection instead of being centrally taken by core routers.

Despite the fairly large amount of literature on EMBAC, we argue that the effectiveness of this approach has not been fully proven yet, as well as the effect of some apparently fundamental engineering parameters has not been fully understood. For example, the duration of the probing phase is a critical issue in practical terms, as a toll quality connection setup duration should be mandatory lower than 1 second. Conversely, if the scope of the probing phase is to accurately measure the traffic load over a network path, the probing phase duration is bounded to span a suitable amount of time to cope with statistical traffic fluctuations, and this amount of time can be eventually much longer than one second, as it depends on the traffic correlation properties (see the excellent formalization of such a sentence in [13]).

This paper sheeds some new light on the effectiveness of EMBAC. By means of both analytical modeling and extensive simulation, we prove that EMBAC operation is extremely effective when a tight form of congestion control on the probing packet traffic is exerted at each network router. In particular, for reasons that will be clear in section 3, we propose to enforce a short probing packet expiration deadline.

Two fundamental consequences of our study are the following. First, the probing phase duration is not a critical issue, i.e. EMBAC can remain effective even with a few ms probing phase duration. Second, the probing packet load, i.e. the number of probing packets per flow setup, can be kept low at the price of a light, but tolerable, degradation in the EMBAC efficiency. Both consequences have fundamental implication on the practical utilization of EMBAC, as they allow on one side a short flow setup time, and on the other side a very limited waste of network resources (which can be exploited for best effort traffic) due to probing traffic overload.

The paper is organized as follows. In section 2, the basic characteristics of EMBAC schemes are presented. Section 3 is the core of the paper. In this section, we explain the details of EMBAC-PD, i.e an EMBAC scheme where each router enforces control of a a short probing packet expiration deadline. We then show (by means of simulation results) that EMBAC-PD is able to support strict QoS requirements and provide an intuitive explanation of the physical reasons under the protocol effectiveness. Section 4 presents an analytical model able to quantitatively capture the EMBAC-PD operation, and provides further understanding on the reasons at the basis of the EMBAC-PD effectiveness. Section 5 deals with the performance evaluation of EMBAC-PD, and studies the dependency of EMBAC-PD on several important engineering parameters, among which probing phase duration and probing load. Concluding comments are presented in section 6, along with further research directions.

# 2 End to end Measurement Based Admission Control Approaches

EMBAC schemes have been independently introduced by Borgonovo et. al. [6, 7], Karlsson [5] and Gibbens and Kelly [8] (as a side issue in a more general pricing-based approach). Although these works consider different frameworks and traffic scenarios, and differ in some important technical details, some general directions are common, and are summarized (and further detailed with our own interpretation of EMBAC) in what follows.

The basic and characterizing feature of EMBAC schemes is to rely on end-to-end in-band measures to determine whether there are enough resources in the network to accept a new connection. As shown in figure 1, a connection is composed of two phases: a probing phase, eventually followed, in the case the connection is accepted, by a data phase. A traffic source that wants to set-up a real time connection starts transmitting a signaling flow, composed of equally spaced (i.e. constant rate) low priority packets tagged as *Probing* in the IP header.

Upon reception of the first probing packet (i.e. after the time interval $\Delta$ shown in the figure), the destination node starts measuring the probing packets arrival statistics over

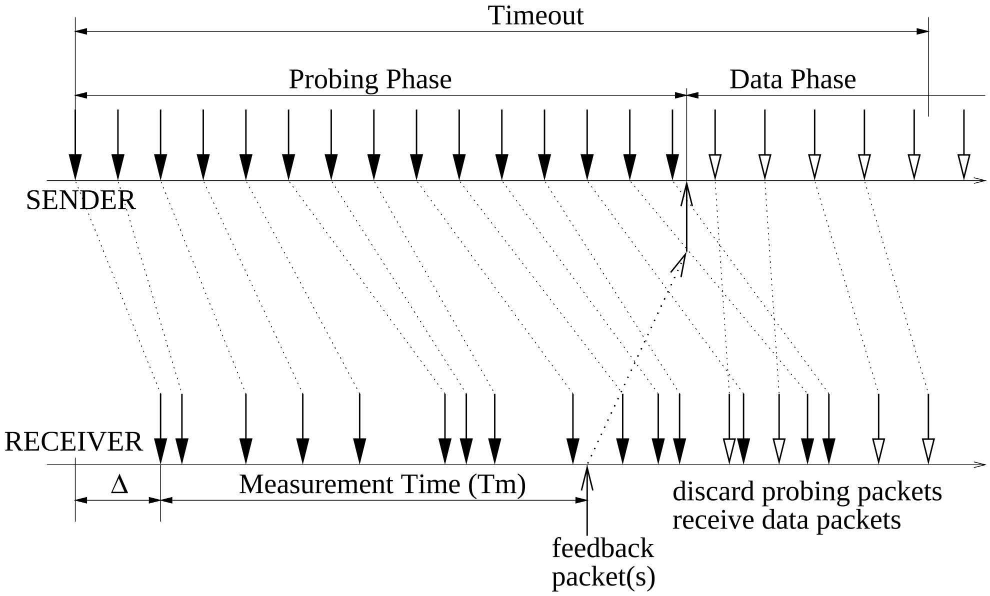


Figure 1: *EMBAC connection setup scheme*

a fixed length measurement period $T_m$. The measured statistics can be as simple as the count of the number of received packets or, by difference, of lost packets [5, 7] within the time interval $T_m$, or more complex including for example delay and delay jitter statistics [6, 7].

At the end of the measurement period, the receiver estimates whether there are enough resources available along the connection path to meet a predetermined QoS requirement. This decision is notified back to the sender (by means of one - or more - *feedback packets*), which either switches from probing to data phase, and starts transmitting high priority *Data Packets*, or aborts the call setup. For sake of robustness, upon initiating the probing phase the sender activates a timeout, which aborts the call setup in case no feedback packets are received before the timeout expiration. This feature is important in practice, when network congestion does not allow any of the probing packets to reach destination.

In EMBAC schemes, core routers are stateless. In full agreement with the DiffServ paradigms, they only need to discriminate between classes of packets. In particular, to protect the already accepted traffic, each core router handles packets offered to a given output link with two[1] distinct queues. One is dedicated to data packets, generated by already accepted connections. The other accommodates probing packets, which are served with lower service priority, i.e. probing packets are transmitted only when the data packets queue is empty. This forwarding mechanism has the fundamental effect that probing packets

[1]Note that, as long as just two priority levels (probing/data) are used within the network routers, EMBAC schemes are limited to provide a single QoS requirement in the whole network, i.e. all accepted connections share the same loss/delay performance.

are forwarded only when resources unused by data traffic are available. Conversely, when the data traffic load is large, probing flows will suffer severe QoS degradation which will be detected by the measurements running at the end nodes. The proposed mechanism can therefore be envisioned as a stable and robust congestion control mechanism which regulates the QoS provided by the network by adaptively increasing and decreasing, during the time and according to the network status, the probability of acceptance of new calls.

An important point is how much probing load is necessary. In principle, as stated in [6, 7], the role of the probing phase should consist in understanding if the already accepted traffic *plus* the additional one carried by the new connection still satisfies the QoS requirements. This implies that a new connection should offer, during its probing phase, at least as much load as that it would provide during the data phase[2]. Despite in principle reasonable, such an approach carries at least two major disadvantages. First, an heavy volume of probing traffic load wastes network resources that could be used for best effort traffic. Second, the throughput performance versus the offered load exhibits a non monotonous behaviour (see details in [9]), which suggest instability in the case of extremely high demand for scarce network resources.

In the reasonable hypothesis [13] that the peak rate of offered calls is small with respect to link capacities, the contribute of each new call in terms of QoS degradation is marginal. This implies that the probing rate is not bounded to be related to the data traffic rate. Owing to these comments, in the paper we will evaluate the EMBAC performance considering a probing traffic generation profile independent of the data traffic profile.

It is also important to remark that, while data packets queue saturation is prevented by EMBAC operation, this is not the case for the probing packet queue. Indeed, the probing offered load is uncontrolled (connections freely start setup/probing phases), and thus it can become eventually far greater than the (time-varying) probing bandwidth available at each node. To avoid probing packets starvation in the case of probing queue overload, some form of congestion control mechanism must be enforced. In previous literature, this has been deployed by means of either an explicit packet lifetime within each router [7, 9] or, more simply, by limiting the probing packets queue size [5, 10].

We claim that a tight control of probing packet congestion is a powerful tool which allows EMBAC to support strict QoS requirements, as the resulting packet discarding may provide useful information on the network congestion status to the end point. This claim was somehow hidden in the technical details of [5, 10], which report results for buffer sizes as short as few packets, and in the mechanism proposed in [7]. In the concluding remarks of our former paper [9], we have tried to achieve preliminary understanding on how tight (and why) this congestion control should be, in order for EMBAC to achieve effective throughput/delay performance, and how this control should be implemented. A

[2]This is evident for the case of constant rate flows considered in [6, 9]: a probing flow transmitted at a rate lower than that requested for the data flow may turn into an accepted call that overloads one or more network links.

much deeper investigation of this topic is the main contribution of this paper.

# 3 EMBAC-PD

In this paper we focus on a particular EMBAC scheme, referred in the following as Packet Delay EMBAC (EMBAC-PD). Also in EMBAC-PD, probing packets are served at each router with a lower priority than data traffic. The novelty of EMBAC-PD is that each router exerts a strict control on the probing packet lifetime (PLT), i.e. the maximum amount of time a probing packet is allowed to remain in the buffer before being served. If the probing packet does not receive service within PLT ms from its arrival in the buffer, the packet expires and is discarded from the buffer.

If the PLT value exerted by each router is small, even a relatively small probing packet buffer (say 200+ packets) is sufficient to guarantee that no buffer overflow occurs. Also, we can neglect packet losses due to transmission errors on the wired channel. Thus, we can safely assume that all probing packet losses are triggered by PLT expiration.

Packet losses are therefore a mean to deliver indirect information on the network congestion status to the end point, which can then use this information to make the accept/reject decision. In particular, in EMBAC-PD, probing packets are transmitted at constant rate, spaced by $I_{prb}$ ms. In a measurement time $T_m$, $n_{prb} = T_m/I_{prb}$ is the number of packets that should be received. By measuring the number of packets received within this time window, the destination is able to determine whether all transmitted packets were received (and in this case accept the connection), or if one or more packet losses occurred (and in this case reject the connection)[3]

We claim that the tuning of the PLT parameter plays a key role in the effectiveness of EMBAC-PD. This claim is proven in figure 2, by means of simulation results. Fig. 2 shows various data packet delay percentiles vs. PLT. It appears that delay percentiles significantly increase with the PLT threshold, reaching unacceptable values as high as $114ms$ in the case of the $99-$th percentile and $PLT = 30ms$.

The simulation model and parameters adopted in the figure is described as follows. The link capacity is 2 Mbit/s, the measurement interval is equal to 1 s and the number of transmitted probing packets over the set up phase is set to 38. An IP telephony traffic scenario is considered in the figure (and, unless otherwise specified, it is also considered throughout the whole paper). Voice calls in the data phase are modeled assuming silence suppression. In particular, we have adopted the two states (ON/OFF) Brady model [14], where the ON (active - i.e. talkspurts) and OFF (silent) states are exponentially distributed with averages equal, respectively, to 1 s and 1.35 s. For these sources, the percentage of time they are in the ON state, called activity factor, is $B_u = 0.4255$. The peak rate is set to

[3]In principle, the decision can be more general, such as accept if a certain percentage of packets is received (as in [9]). Note that it is trivial to modify our analysis, and particularly formula 4 to cope with such a more general approach.

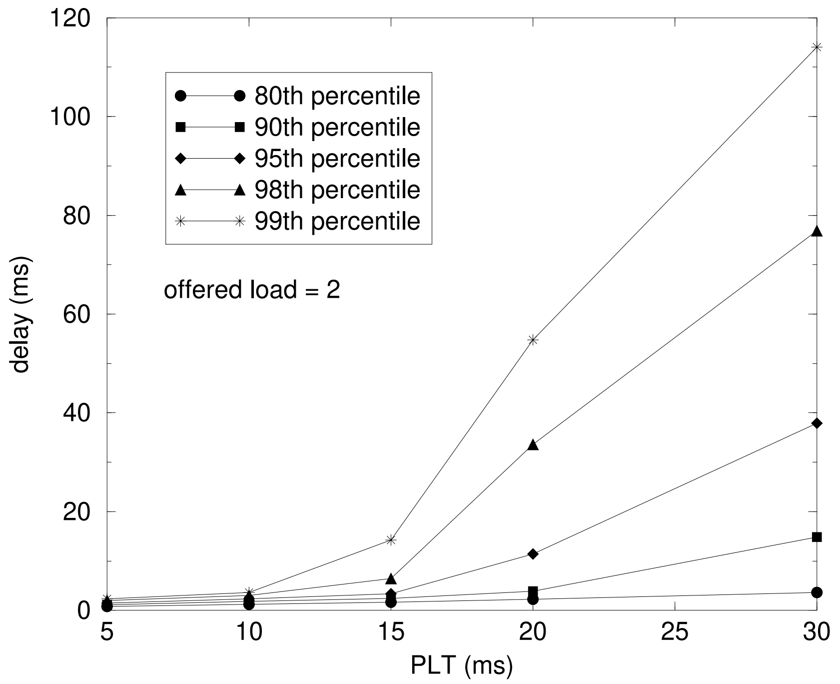


Figure 2: *Delay percentiles vs. PLT*

$B_p = 32$ Kbit/s and the packet length is equal to 1000 bits. Calls are generated according to a Poisson process and have an exponentially distributed duration, with mean value equal to 3 minutes. In figure 2, a 2.0 normalized offered load is considered. The arrival rate $\lambda$, in calls/s, and the normalized offered load are related by the formula:

$$\text{normalized load} = \frac{\lambda}{\mu} \cdot \frac{B_p B_u}{C}$$

where $1/\mu$ is the average call duration measured in seconds, and $C$ is the channel rate in Kbit/s. In the simulation model, for sake of simplicity, we have assumed instantaneous feedbacks. In this case tight timeouts can be adopted (i.e. few tens of ms longer than the measurement time $T_m$).

The results of figure 2 make us believe that almost arbitrarily tight quality of service control (e.g. 99th percentile delays of the order of few ms) can be obtained by adopting small PLT values within each network router. To understant *why* this is possible, and what is the role of the PLT threshold on protocol performance let us focus on the graphical illustration of what happens at the packet level given in figure 3. This figure reports two example patterns of link utilization by data packets. Assuming that an interarrival data packet period is divided in 20 slots (i.e. that the link capacity is equal to 20 active connections), the figure illustrates the case of link utilization equal to 60%, i.e. 12 active connections, and 30%, i.e. 6 packets to be transmitted into an interarrival period.

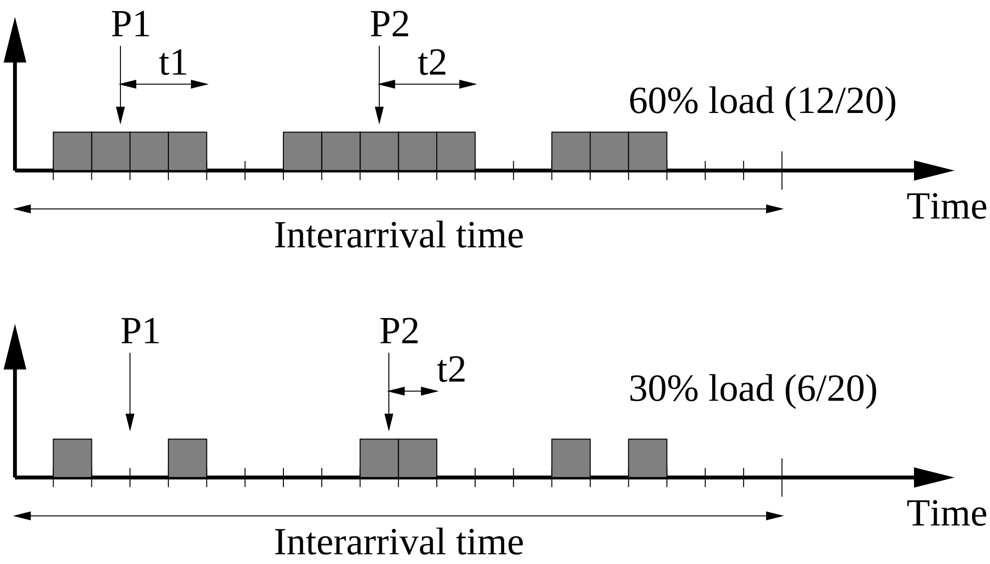


Figure 3: *EMBAC-PD rationale*

Consider now two probing connections, P1 and P2, offered to the link. If the PLT is greater or equal than the interarrival time, as in the case $PLT = 30ms$ displayed in Fig 2, the probing packets will encounter an idle slot for transmission, regardless of the link utilization pattern and of the data traffic load. In other words, if the PLT value is large enough, our measurement scheme is not able to detect *how much load* is offered to the link, but only to detect if there is enough spare capacity available.

In a variable rate scenario, where the aggregate traffic on a link varies in time according to source activity, the detection of enough residual bandwidth over the measurement interval (as studied, for example, by the analytical model presented in [9]) does not translate into guarantees of a suitable QoS over the whole call holding time, so that a stricter control on the link load has to be enforced.

Let us now see what happens when a PLT value much shorter than the data packets interarrival time is adopted. We refer again to Fig. 3. Even if there is enough bandwidth to transmit the probing packets, these packets can expire, because of a short PLT value, *before an idle slot becomes available*. Moreover, as graphically shown in the figure, the expected time before receiving service strongly depends on the data traffic load (the values $t_1$ and $t_2$ are greater in the 60% case of the figure than in the 30% case). Thus, the probability that a packet is lost because of PLT expiration grows with the data traffic load. These remarks explain the behavior graphically shown in Fig. 2: the shorter the PLT value employed in each network node, the earlier congestion is detected by the end to end measurement process, the tighter the control of the accepted load. In the following section 4, an analytical model will be presented to quantitatively capture these remarks.

Before proceeding, it is useful to note that, given a traffic pattern (in the considered

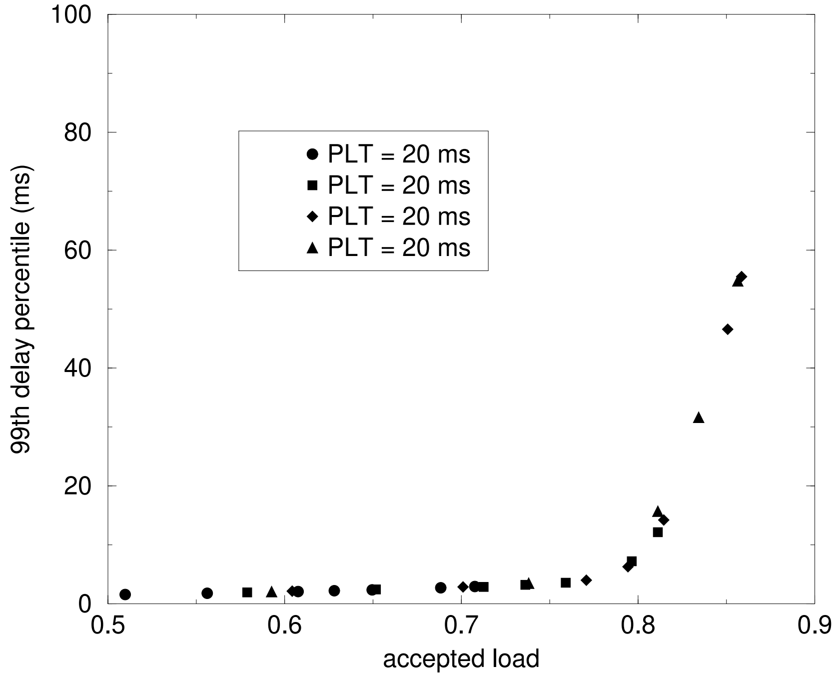


Figure 4: *Delay percentiles vs. throughput*

scenario, 32Kbit/s ON-OFF Brady sources) and a channel capacity, the delay performance is strongly related to throughput. This is graphically shown in Fig. 4 which compares the 99−th percentile vs. throughput performance when varying PLT among 5,10,15 and 20ms and setting all the other parameters as in Fig. 2. To provide QoS guarantees it is therefore sufficient to limit the accepted load below a given threshold. For example, in the reference case of IP Telephony traffic, the mouth to ear delay for toll quality voice must be limited to 150ms. As in a typical VoIP scenario packetization delay, transmission delay, propagation delay, processing delay and jitter compensation generally account for 100-120ms, the network queueing delay must be limited within a few tens milliseconds (so that a few ms per link appears a reasonable target). A PLT threshold strict enough to limit throughput below 0.75% has therefore to be adopted. In the scenario considered in Fig. 2 this corresponds to set the PLT threshold to 10ms.

# 4 Approximate Analytical Model

A queueing analysis able to capture the data packets delay performance of EMBAC appears to be overly complex. In this section we provide a much simpler approximate asymptotic analysis able to estimate the throughput performance. As noted in the previous section, once the traffic pattern and channel capacity is fixed, delay percentile requirements can be directly related to throughput performance. Thus, the numerical results derived by means of our approximate analytical model indeed provide useful insights on the overall system performance.

As usually done in call admission control schemes analysis, let us focus on a single link. Assume that new calls (i.e. calls entering the probing phase) are offered to the link according to a Poisson process of rate $\lambda$ calls/s. Let $1/\mu$ be the average connection holding time. Assume that each call is an on/off source with peak rate $B_p$, and activity factor (i.e. percentage of time the source is in the ON state) $B_u$. Unless otherwise specified, in the numerical results we will consider voice on/off sources, modeled according to the Brady model [14], for which $B_u = 0.4255$.

Let $K(t)$ be the stochastic process representing the number of accepted connections at time $t$. Under the approximations that (A1) subsequent accept/reject decisions are independent, and that (A2) the times at which probing connections take an accept/reject decision are still Poisson[4] the process $K(t)$ can be modeled as a birth-death Markov chain, with birth/death coefficients (using standard notation [15]):

$$\begin{cases} \lambda_K &= \lambda P_a(K) \\ \mu_K &= K\mu \end{cases} \tag{1}$$

This system resembles that of an Erlang loss system [15] with the difference that the arrival rate, $\lambda \cdot P_a(K)$, depends on the system status $K$, $P_a(K)$ being the probability (to be determined in section 4.1) that a probing call is accepted in the system given $K$ already admitted connections.

Let $\pi_K(k)$ be the steady state probability of finding $k$ calls in the system, obtained by numerical solution of (1). According to known results in queueing theory [16], $\pi_K(k)$ depends on the call duration distribution only through its mean value $1/\mu$. Hence, our model applies to general distribution of the call duration. By using the values $\pi_K(k)$ we are able to determine several performance figures. In particular, the normalized system throughput $\eta$, i.e. the average fraction of link occupied by accepted calls, is immediately expressed as:

$$\eta = \sum_k \frac{k \cdot B_p B_u}{C} \pi_K(k) \tag{2}$$

where $C$ is the channel capacity (e.g. in Kbit/s), and $B_p \cdot B_u$ is the mean source rate.

[4]While approximation A2 is very reasonable (the time elapsed between the arrival of a probing call and the time in which a decision is taken can be considered equal to $T_m$), approximation A1 is the most critical. It implies that subsequent probing calls see the system in independent conditions, and thus it is less accurate for large offered load, i.e. short time between subsequent decisions, and for highly correlated traffic. However, as we'll show later, despite this approximation, the matching between analytical and simulative results is quite good.

### 4.1 Determination of $P_a(K)$

Let $N_K(t)$ be the number of *active* calls (calls in the ON state) at time $t$, assuming $K = K(t)$ accepted calls. Clearly, $0 \geq N_K(t) \geq K$, and, at a random time instant $t$,

$$P\{N_K(t) = k|K\} = B(K,k) = \binom{K}{k} B_u^k (1 - B_u)^{K-k} \tag{3}$$

where $B_u$ is the activity factor of each source. To make the determination of $P_a(K)$ a tractable problem, we assume (approximation A3) that subsequent arrivals of probing packets within a measurement period see the same link occupation status, i.e. the same number of active connections. In other words, we assume that a probing call samples the status of the channel only once, i.e. that the effect of $T_m$ to estimate channel variations is null (in this sense our analysis is *asymptotic*, i.e. it acts as $T_m \to 0$).

We remark that this assumption is the worst possible case from the call admission point of view, since it implies that during a probing phase each call has no mean to better estimate the link load by extending the probing phase duration. This assumption, thus, renders the analytical model independent of the traffic correlation model.

Consider now a probing call entering the system. Owing to approximation A3, it samples the link load once. Being $k$ the number of active calls encountered, the link load is given by

$$\rho(k) = k\frac{B_p}{C}$$

In the case $\rho(k) \geq 1$, probing packets do not find bandwidth available, and thus are rejected. If, conversely, $\rho(k) < 1$, we can study the link behaviour at the packet level. It comes quite reasonable to approximate (Approximation A4) the link performance with that of an M/D/1 queueing system [5] with offered load equal to $\rho(k)$. Unless the probing packets of the considered connection are very close, they are expected to fall into different busy periods. In particular, owing to approximation A1 (which, for this purpose, can be restricted to simply assume that a probing packet never encounters other packets lined up in the probing buffer), a probing packet is dropped if it encounters a busy period (i.e. the data packet queue busy), whose remaining duration lasts for more than the packet lifetime. Define with $\phi$ the probing packet lifetime, and define with $P_{nexp}(\phi, \rho(k))$ the probability that a probing packet is successfully transmitted, i.e. it does not encounter a remaining busy period duration which lasts more than $\phi$ in an M/D/1 queue with load $\rho(k)$. Being $n_{prb}$ the number of probing packets transmitted within a measurement time $T_m$, the connection

[5] It might perhaps appear more natural to consider the link as modeled by an ND/D/1 queue, as long as all $k$ active sources have the same rate. However, we argue that the M/D/1 approximation is simpler, it is more robust to transient effects (because of complexity, we haven't included in our analysis the activation - deactivation transient behavior of offered sources), and can be trivially generalized to heterogeneous traffic mix as expected in real systems.

is accepted if all the probing packets are successfully transmitted. This allows us to finally express the probability for a probing connection to be accepted as:

$$P_a(K) = \sum_{k=0}^{K} B(K,k) P_{nexp}\left(\phi, k\frac{B_p}{C}\right)^{n_{prb}} \tag{4}$$

To conclude the analysis, it remains to express $P_{nexp}(\phi, \rho)$. This is an easy task for an M/D/1 system. In fact, consider an M/D/1 system with normalized offered load $\rho$, i.e. $\rho$ represents the Poisson arrival rate of customers to the system assuming an unitary service time. For such a system we recall that the busy period distribution, i.e. the probability that exactly $n+1$ customers are served during one busy period $\beta$, is well known and given by:

$$P\{\beta = n+1\} = e^{-\rho}(n+1)^{n-1}(\rho e^{-\rho})^n/n! \tag{5}$$

the cumulative busy period distribution $F_\beta(t) = P\{\beta \leq t\}$ is readily computed as:

$$F_\beta(t) = \begin{cases} 0 & t < 1 \\ \sum_{i=0}^{\lfloor t-1 \rfloor} e^{-\rho}(i+1)^{i-1}(\rho e^{-\rho})^i/i! & t \geq 1 \end{cases} \tag{6}$$

being $\lfloor t-1 \rfloor$ the greatest integer value lower or equal to $\phi - 1$.

A probing packet arriving to the system finds an empty data packet queue (i.e. the queue approximated with an M/D/1) with probability $1-\rho$, while with probability $\rho$, it encounters the queue busy. The probability density function of the residual duration of the busy period, $f_r(t)$, is related to the cumulative distribution of the busy period by the general formula:

$$f_r(t) = \frac{1 - F_\beta(t)}{E[\beta]} \tag{7}$$

where, $E[\beta]$ is the average busy period given[6] by $1/(1-\rho)$.

Accounting for the fact that a probing packet finds a busy period only with probability $\rho$, we finally obtain the probability that a probing packet is successfully transmitted, as:

$$P_{nexp}(\phi, \rho) = (1-\rho) + \rho \int_0^\phi f_r(t)dt \tag{8}$$

This probability can be efficiently computed by noting that $f_r(t)$ is a step-wise function, and thus its values can be stored only for discrete values $i$, yielding:

$$P_{nexp}(\phi, \rho) = (1-\rho) + \rho \left[ \sum_{i=0}^{\lfloor \phi - 1 \rfloor} f_r(i) + f_r(\lfloor \phi \rfloor) \cdot (\phi - \lfloor \phi \rfloor) \right] \tag{9}$$

[6]This is trivial, noting that $E[\text{idle}] = 1/\rho$, due to Poisson arrival assumption, and

$$\rho = \frac{E[\beta]}{E[\beta] + E[\text{idle}]} \rightarrow E[\beta] = \frac{\rho}{1-\rho} E[\text{idle}] = \frac{1}{1-\rho}$$

# 5 Performance Evaluation

Figures 5, 6, and 7 report a comparison of the analytical results with that obtained by means of simulation. The figures plot the link throughput versus the normalized offered load. In the three figures, the traffic scenario is that of voice calls modeled according to the Brady model, and described in section 3. The link capacity is set to 2 Mbit/s. Three different PLT values are considered: 2, 5, and 8 ms. The figures differ in the number of probing packet transmitted in a fixed measurement time $T_m$ set slightly lower than 1 second. Figure 5 considers the case of probing packet interdeparture time $I_{prb} = 200$ ms, and thus only 5 probing packets transmitted (in other words, the accept decision is taken if all 5 packets arrive to destination). The case of $I_{prb} = 100$ ms ($n_{prb} = 10$) and $I_{prb} = 50$ ms ($n_{prb} = 20$) are presented, respectively, in figures 6 and 7.

The figures show an acceptable matching between the analytical results and the simulative ones, especially considering the four approximations employed in the analysis. Interesting is the case of PLT=2 ms, for which the agreement is excellent (at least in the cases presented in figures 5 and 6). This seems to imply that the approximation of the remaining busy period distribution with that of an M/D/1 queue is good for small remaining busy periods (for a 2 Mbit/s link and 1000 bit packets, PLT=2 ms means 4 packets) while it is less accurate for longer remaining busy periods. This seems to suggest that a better matching can be obtained by using an ND/D/1 queueing model in the analysis, although we argue that, besides the additional complexity, this model would be too specific for homogeneous traffic scenarios (as that simulated).

Besides the analysis versus simulation matching, by comparing the three figures we note that a a much greater throughput control is exerted in the case of 20 probing packets (figure 7). As explained in section 3, to obtain 99th data packet delay percentiles of the order of few ms (toll quality voice), the system throughput should be limited to about 0.75. While, in the case of figure 7, this control is achieved for a normalized offered load up to 8, and a PLT value of 8 ms, in the case of only 5 probing packets (figure 5) the same PLT value leads to a throughput value as high as 0.9 (and a corresponding 99th delay percentile greater than 150 ms). To achieve a 0.75 throughput target at load 8, we see from the figure that, in this case, it is necessary to enforce a PLT value lower than 2 ms.

To understand the effect of the setup phase duration, it is interesting to analyze the simulation results presented in figures 8 and 9. These figures report the accepted vs. offered load for 10min average call duration and 2Mbit/s link capacity. A fixed number of probing packets are transmitted over different measurement time periods, varying accordingly the probing packet interarrival time $I_{prb}$. Figure 8 considers the case of 5 probing packets, while figure 9 considers 10 probing packets. Quite interesting, the probing phase duration has a very limited effect on the system performance: differences are evident only for a high number of transmitted packets, and in any case never exceed 4%. We then conclude that performance ultimately depend on the number of probing packets transmitted (and in turn

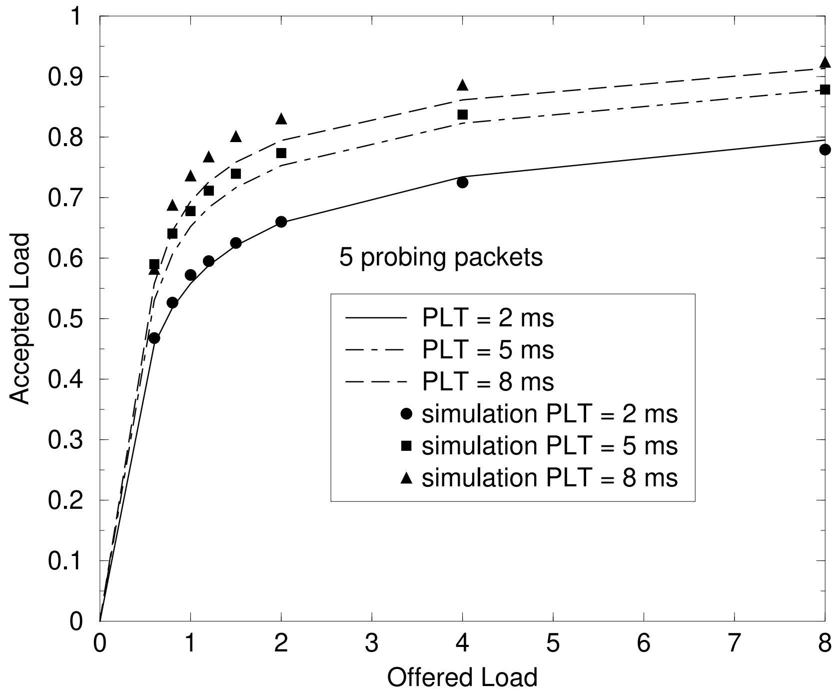


Figure 5: *Analysis versus Simulation:* $n_{prb} = 5$

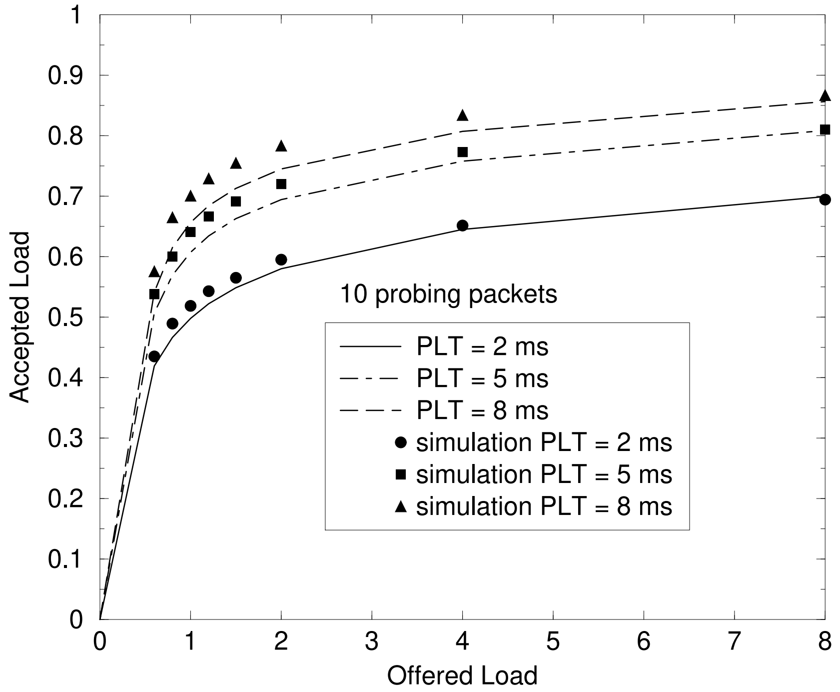


Figure 6: *Analysis versus Simulation:* $n_{prb} = 10$

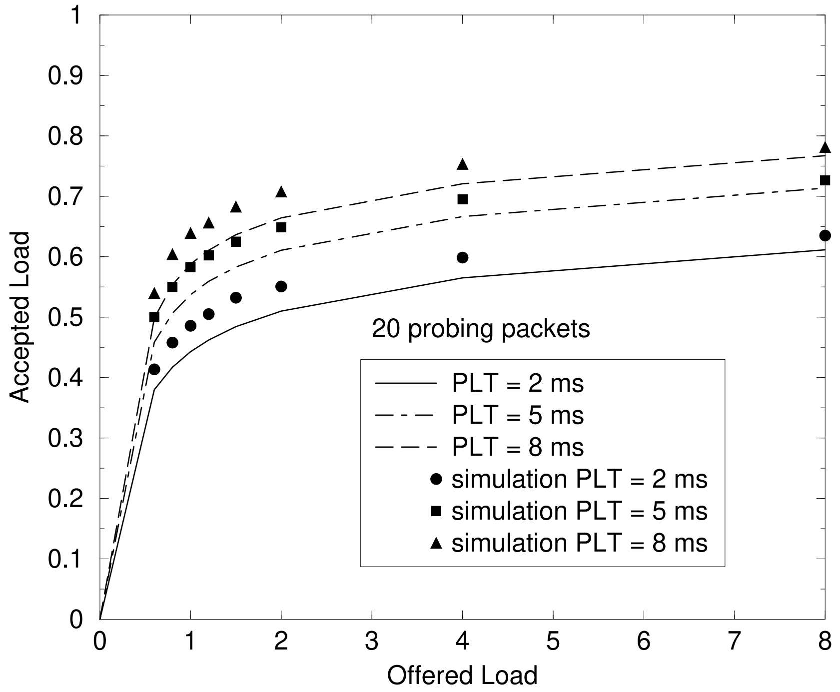


Figure 7: *Analysis versus Simulation:* $n_{prb} = 20$

on the different probability to receive all the transmitted packets correctly) rather than on the time interval on which probing packets transmission spans. We also conclude that the capability of EMBAC-PD to estimate offered traffic variations by employing longer measurement times (but still below toll quality requirements) is rather limited. This fact is extremely important, as it clearly shows that the EMBAC-PD effectiveness does not stay in its capability to estimate the offered traffic dynamics (and thus the need for long measurement times would occur), but stays in the preventive packet loss occurring because of PLT expiration before the system gets congested. Hence, the major constraint on the probing phase duration is therefore due to the practical implementation requirements on the set-up phase length. In the following, where non specified, we will assume a probing phase duration lasting for 1 second.

From the above discussion, it appears that the two most critical parameters in the EMBAC-PD engineering are the PLT value and the number of probing packets $n_{prb}$. These parameters appear to play a somehow interchangeable role. This is evident by considering equation 4, which shows that the connection acceptance probability depends on

$$P_{nexp}\left(\phi, k\frac{B_p}{C}\right)^{n_{prb}}$$

By increasing the PLT value $\phi$, and correspondingly increasing $n_{prb}$, the above value results invariant (once the number of active connections $k$ is fixed).

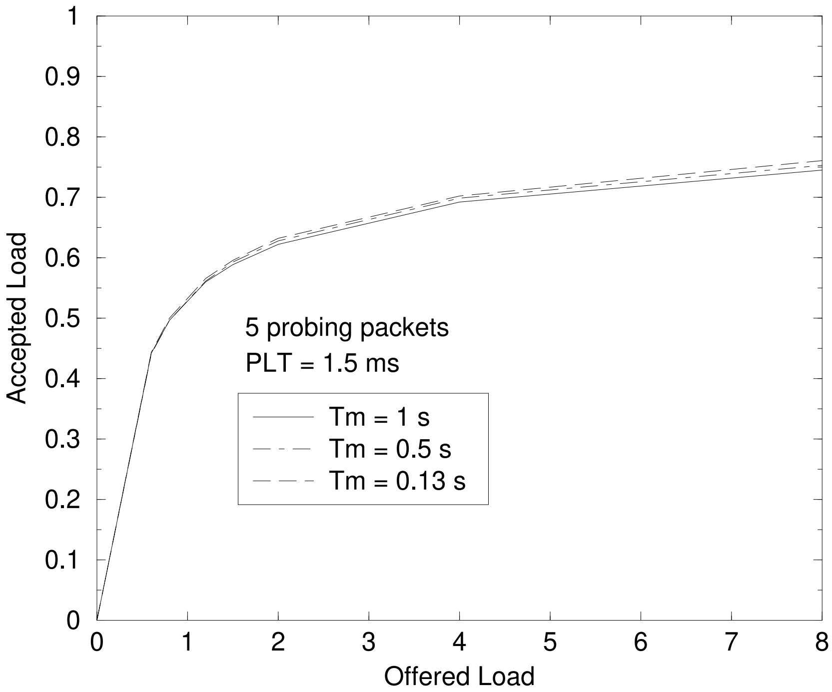


Figure 8: *Effect of the length of the set up phase,* $n_{prb} = 5$

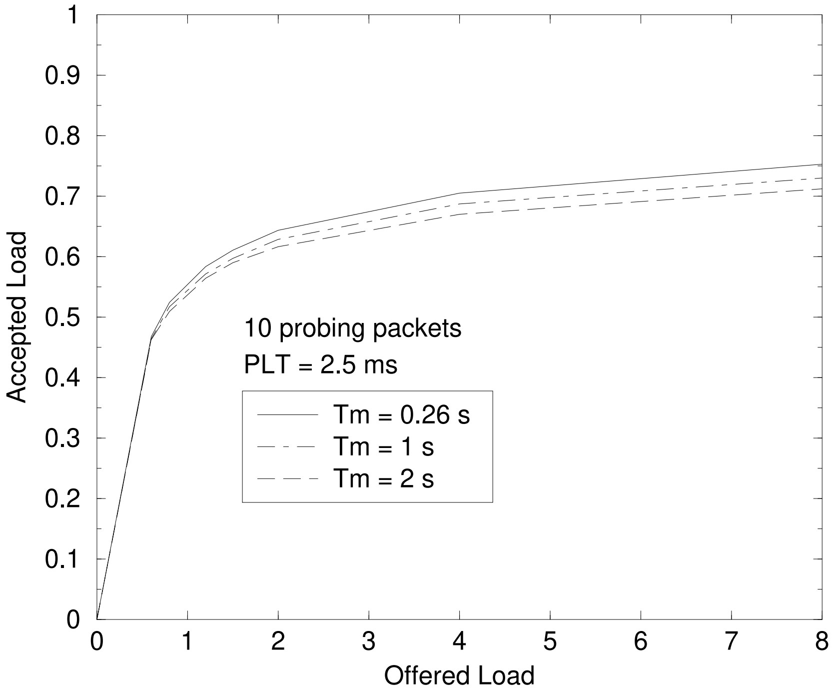


Figure 9: *Effect of the length of the set up phase,* $n_{prb} = 10$

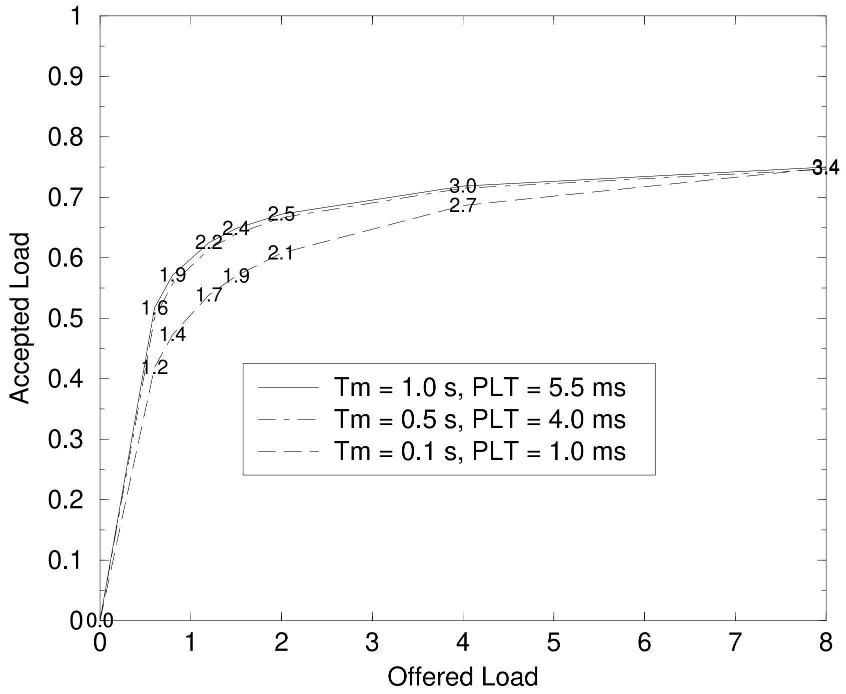


Figure 10: *Different parameter combinations to provide same QoS control*

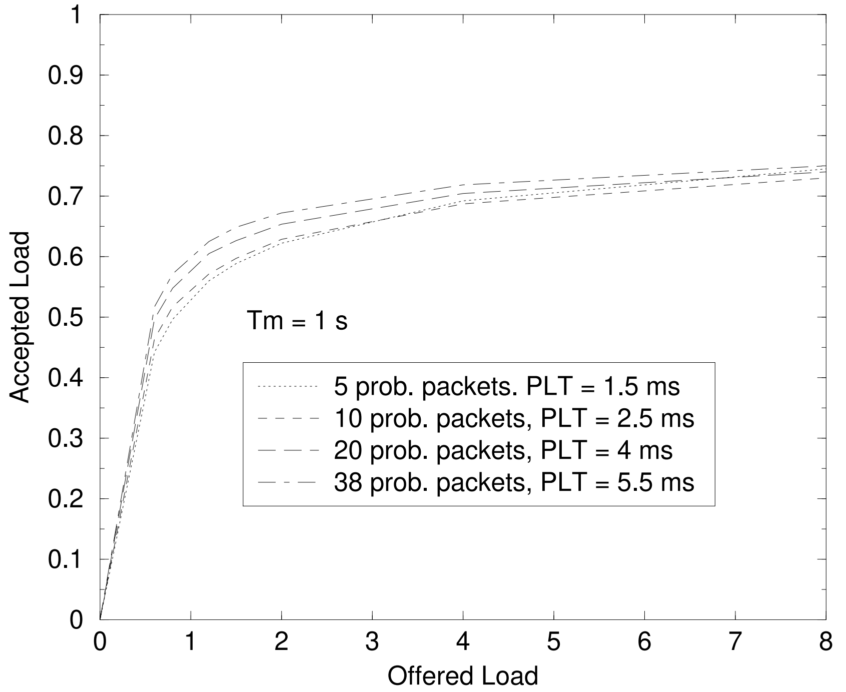


Figure 11: *EMBAC-PD throughput and delay performance for varying number of probing packets*

This phenomenon is graphically shown in Fig.10. In the figure, the same probing packets interdeparture time has been adopted. The three cases of $T_m$= 0.1s, 0.5s and 1s have been plotted, corresponding to $n_{prb} = 4, 19, 38$. The PLT values have been chosen to provide a normalized throughput equal to 0.75 at load 8.0. This guarantees extremely low (few ms) 99th delay percentiles, as reported by the numerical values (for different loads) on top of the figure. All the other parameters has been set as in Fig.2.

Figure 10 shows that a suitable tuning of the PLT threshold allows to satisfy the QoS requirements even when very short set-up times are adopted. A substantial reduction of the overhead due to probing packets transmissions is also achieved in this case, as a short set-up time corresponds to significant reduction in the number of probing packets transmitted over set-up. At load 2.0, for example, the probing overhead decreases from 3% of the channel capacity when 38 probing packets per connection are sent, to only 0.26% when 4 probing packets are transmitted.

In the same time figure 10 shows that the achieved reduction of the probing traffic load is paid with a strong reduction of the throughput achieved in underload conditions (e.g. load $< 0.75$). In this case, we see that a very small PLT value leads to a throughput reduction in the underload range, meaning that several connections are blocked even if sufficient capacity is available. This extremely critical behaviour leads us to conclude that a better way to tune the protocol is to maintain a reasonably high PLT value (say 3 to 10 ms), and correspondingly increase the number of probing packets transmitted, until the target QoS performance is achieved.

Although some of the above considerations are due to the extremely short PLT value employed to stand a measurement time as low as 100 ms, the above remarks are confirmed by the results presented in figure 11. Here, an homogeneous measurement period of 1s is employed, and the different number of probing packets is realized by varying the probing packets interdeparture time.

We conclude this performance evaluation section by showing some additional results concerning different scenarios. The encouraging results for Brady voice traffic lead us to question whether the EMBAC-PD scheme is also effective with slowly activating/deactivating traffic sources. To implement a traffic model with higher time correlation, we have modified the Brady model by setting the ON and OFF periods average length to, respectively, 10s and 13.5s. The sources activity factor is therefore the same as in the Brady model, but ten times longer silence and activity periods are introduced to test the robustness of the call admission control protocol. A comparison of the delay vs. PLT performance at load 4, for the two source models and $n_{prb} = 5, 10$ is reported in Fig. 12. The results show that a suitable setting of PLT allows to control traffic with a dynamic much longer than the measurement interval. However, we note that for the long dynamic traffic, stricter PLT thresholds are required to achieve the same performance of the Brady model. This is paid, as explained before, with a lower efficiency at low load.

Figures 13 and 14 compare the protocol performance for 32 kbit/s Brady voice sources

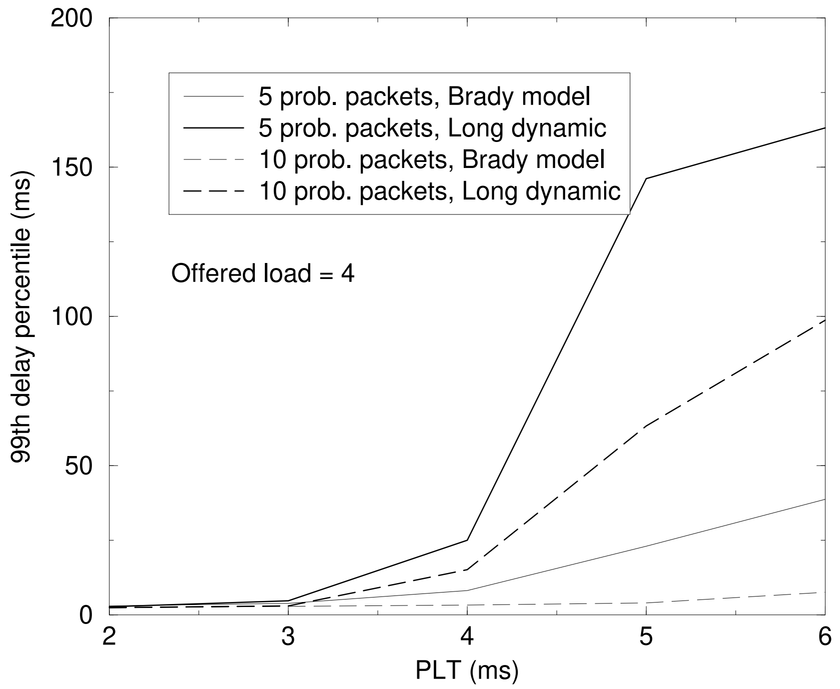


Figure 12: *Different source models: delay percentile vs. PLT at load 4.0*

in a 2 Mbit/s and 5 Mbit/s link scenarios. The set-up time was set to 1 s, $n_{prb}$ varied between 5 and 10 packets, and the offered load was set to 4.0. In particular, Fig. 13 plots throughput vs. PLT performance when varying the number of packets and link capacity, while Fig.14 displays the 99−th percentiles of the delay vs. PLT. We observe that the same PLT thresholds adopted in the 2 Mbit/s case can be safely adopted in the 5 Mbit/s, and actually result in better delay performance and channel utilization, which increases of around 10% at load 4.0. Similar results can be derived also for slow traffic dynamic and confirm the robustness and efficacy of the protocol.

# 6 Conclusions

In this paper we have discussed the performance of End-To-End Measurement Based Admission Control (EMBAC) schemes for IP networks. These schemes, recently proposed, allow to realize a fully distributed and scalable admission control scheme over a stateless Internet architecture, such as that considered in the Differentiated Services Framework. The EMBAC implementation allows core network routers to be oblivious to each individual flow, and only requires routers able to differentiate between two classes of packets (probing and data packets) and handle them with different forwarding priorities.

In this paper we have concentrated our attention on a specific EMBAC scheme, called

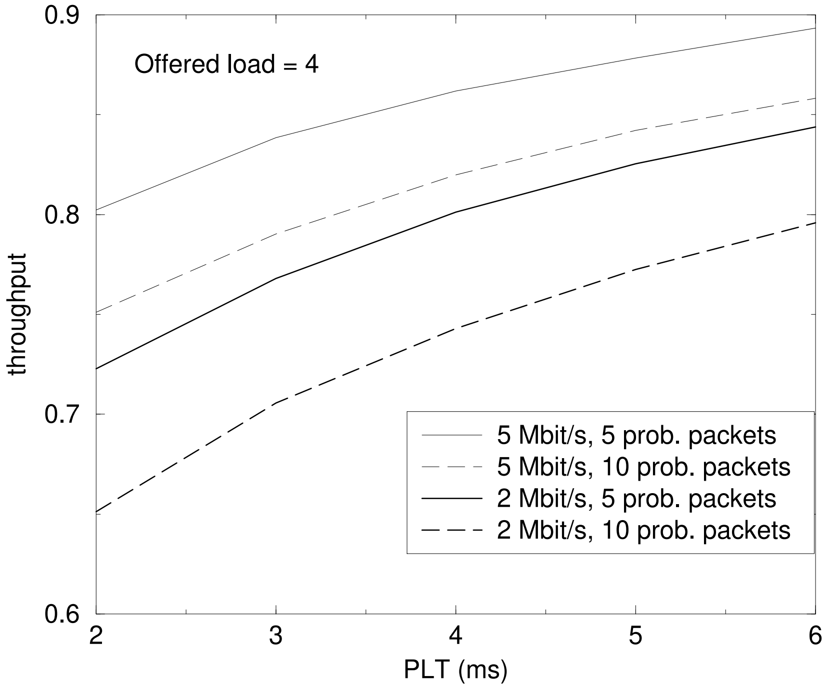


Figure 13: *2Mbit/s vs 5Mbit/s link capacity: throughput vs. PLT*

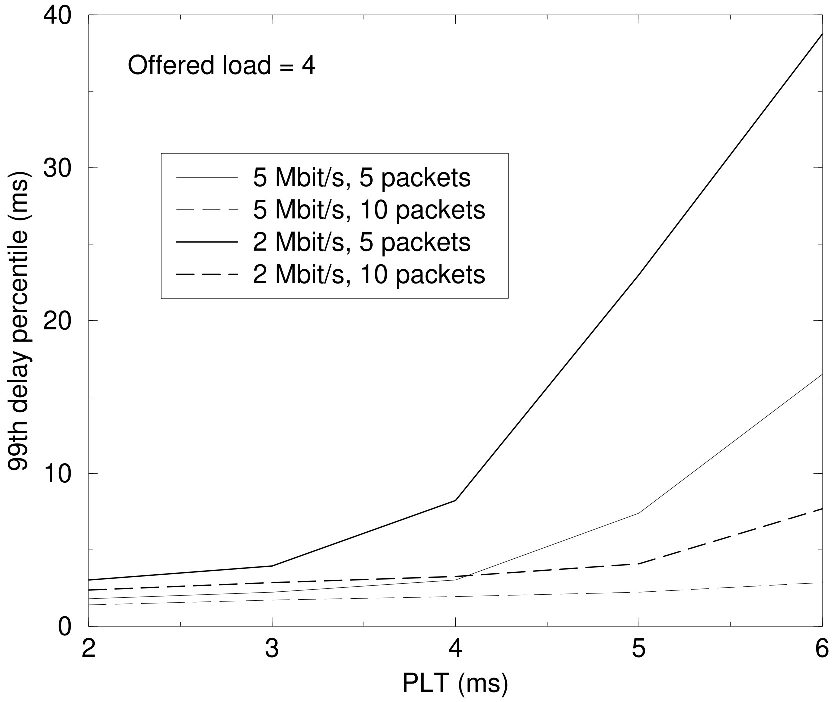


Figure 14: *2Mbit/s vs 5Mbit/s link capacity: delay percentiles vs. PLT*

EMBAC-PD (Packet Delay). In addition to the forwarding scheme, EMBAC-PD requires network routers to exert a strict control of the maximum amount of time a probing packet is allowed to stay in the probing packets buffer. We have shown that probing packet losses occurring within network routers are detected at each connection end-point, and can be effectively used to define a fully distributed connection admission control scheme able to provide extremely tight control on the QoS performance of the accepted real-time connections (say 99th delay percentiles of the order of few ms).

The effectiveness of EMBAC-PD has ben thoroughly evaluated by means of analytical modeling and simulation results. Our extensive performance investigation has put into light the dependency of EMBAC-PD on various engineering parameters, such as probing packet transmission rate and probing phase duration. Moreover, a clear physical interpretation of the reason why EMBAC-PD is effective has been given, in both qualitative (section 3) and mathematical (section 4) terms.

A potential drawback of the considered EMBAC-PD scheme is that the packet lifetime approach is of difficult implementation in the network routers. Alternative and simpler implementation schemes able to provide a similar forwarding behaviour such as very short push-out probing buffers (a probing packet remains in the buffer for short time, until it is pushed out by the arrival of the next one), or scheduling disciplines which allow to transmit probing packets only if the bandwidth utilized by already accepted connections is lower than a given threshold, can be considered and are currently under study.

# References


[1] K. Nichols, S. Blake, F. Baker, D. Black, “Definition of the Differentiated Services Field (DS Field) in the IPv4 and Ipv6 Headers”, RFC2474, December 1998.

[2] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, W. Weiss, “An Architecture for Differentiated Services”, RFC2475, December 1998.

[3] R. Braden, L. Zhang, S. Berson, S. Herzog, S. Jamin, “Resource Re SerVation Protocol (RSVP) - Version 1 Functional Specification”, RFC2205, September 1997.

[4] J. Wroclawsky, “The use of RSVP with IETF Integrated Services”, RFC2210, September 1997.

[5] G. Karlsson, ”Providing Quality for Internet Video Services”, CNIT/IEEE 10th International Tyrrhenian Workshop on Digital Communications , Ischia, Italy, September 1998.

[6] F. Borgonovo, A. Capone, L. Fratta, M. Marchese, C. Petrioli, “PCP: A Bandwidth Guaranteed Transport Service for IP Networks”, IEEE ICC’99, Vancouver, Canada, June 1999.

[7] F. Borgonovo, A. Capone, L. Fratta, C. Petrioli, “VBR bandwidth guaranteed services over DiffServ Networks”, IEEE RTAS Workshop, Vancouver, Canada, June 1999.

[8] R. J. Gibbens, F. P. Kelly, ”Distributed Connection Acceptance Control for a Connectionless Network”, Proc. of 16th International Teletraffic Conference, Edimburgh, June 1999.

[9] G. Bianchi, A. Capone, C. Petrioli, “Throughput Analysis of End-to-End Measurement-Based Admission Control in IP”, Proc. of INFOCOM 2000, Tel Aviv, Israel, March 2000.

[10] V. Elek, G. Karlsson, R. Ronngren, "Admission Control Based on End to End measurements", Proc. of INFOCOM 2000, Tel Aviv, Israel, March 2000.

[11] S. Jamin, P. Danzig, S. Shenker, L. Zhang, “A measurement based Admission Control algorithm for Integrated Services Packet Networks”, IEEE/ACM Transactions on Networking, 5(1):56-70. February 1997.

[12] R.J. Gibbens, F.P. Kelly, "Measurement-Based Connection Admission Control", Proc. of 15th International Teletraffic Congress, June 1997.

[13] M.Grossglauser, D. Tse, “A Framework for Robust Measurement-Based Admission Control”, IEEE Trans. on Networking, vol. 7, no. 3, June 1999, pp. 293-309.

[14] P. T. Brady, “A Model for Generating On-Off Speech Patterns in Two-Way Conversation”, The Bell System Technical Journal, September 1969, pp.2445-2471.

[15] L. Kleinrock, *Queueing Systems*, Voll. 1 & 2, John Wiley & Sons Inc., 1975.

[16] S. S. Lam, “Queueing networks with population size constraints”, IBM J. Res Develop. July 1977.