Measurement Based Admission Control for a Class-based QoS Framework

Thomas Bohnert, Edmundo Monteiro

University of Coimbra
Laboratory of Communications and Telematics
CISUC/DEI, Po II, Pinhal de Marrocos, 3030-290 Coimbra Portugal
Phone: +351239790000 Fax: +351239701266
WWW home page: `http://cisuc.dei.uc.pt`
{`tbohnert,edmundo`}`@dei.uc.pt`

Abstract. In the Laboratory of Communications and Telematics (LCT) of the University of Coimbra researchers have developed a novel, adaptive class-based QoS framework (LCT-QoS). In contrast to common, static resource allocation the core concept behind this approach is a dynamic resource distribution based on delay and loss measurements. As a further step towards a complete framework, currently research is done to find an effective admission control algorithm for resource control on the edge of a QoS domain. In this paper we present our first achievements and the disclosed issues towards a delay and loss measurement based admission control. We first discuss the general need of admission control to manage resource demand. Next, we derive an algorithm starting with an introduction of the LCT-QoS and its inherent loss and delay measurements and the general definition of admission control. We present the simulation setup we used and as well as some results we collected by examining the efficiency of our algorithm under various conditions. The paper is concluded with a discussion of the results and present identification of directions for further steps.

Keywords: Measurement Based Admission Control, Adaptive QoS, Delay and Loss Measurements.

1 Introduction

The decision about an appropriate dimension of a link between two nodes in a packet switched network can mainly be considered as a trade-off between initial and operational costs, the link utilisation and finally the Quality of Service (QoS) a provider aims to offer to its customers. Regarding the first two arguments, statistical multiplexing has proven to guarantee high link utilisation along with minimising required costs and has therefore become the de facto standard in modern network technology. Although the deployment of statistical multiplexing has this advantage it imposes some vast challenges regarding the last argument, the provisioning of QoS. While in times of average network load QoS parameters like delay and loss are to a certain extent, these parameters can rapidly trespass given thresholds when many flows become active simultaneously. Independent of the applied multiplexing strategy, every approach is doomed to fail if the load is simply overwhelming. Therefore, to guarantee boundaries for QoS parameters in a network with finite resources, an entity, which performs Admission Control (AC) of new flows to a common link is mandatory and became subject of extensive research in the past years.

In the Laboratory of Communications and Telematics (LCT) at the University of Coimbra (UC) researchers have developed a novel, adaptive QoS framework (LCT-QoS) inspired by the IETF[1] Differentiated Service (DiffServ) model [1]. Since the conceptual need of AC is not limited to a certain QoS architecture research is done approaching to an efficient *Measurement Based* Admission Control (MBAC) algorithm for LCT-QoS. A core concept of this architecture is delay and loss measurement at packet level. Consequently, a first step towards an integrated AC was to scrutinise the practicability of these measurements for AC and the results of this investigation are presented in this article.

In the residual sections we first present an overview about the evolution of AC and related work, Sect. 2. In Sect. 3 we introduce the LCT-QoS framework and the AC algorithm we deployed. The experimental setup and the results are presented in Sect. 4. Finally, in Sect. 5 we conclude.

2 Background and Related Work

Admission Control is applied in various areas of telecommunication like Call Admission Control (CAC) in the POTS[2]. The role of a particular CAC algorithm in this connection-oriented circuit-switched network is to decide whether to switch a new call or not according to knowledge about resource availability. In this network with its inherent, static and therefore highly predictable behaviour, AC algorithms can be derived from strong mathematical models on a high confidence level. CAC is classified as *Parameter Based* AC since the decision is based on previously provided, *a priori* characterisation of traffic behaviour and thus, resource demands. The significant feature of parameter based AC is its provision of so-called *hard* or *quantitative* guarantees (e.g. fixed transmission rate).

In connection-less packet-switched networks like the Internet, however, mathematical models proven to be valid in telecommunication for almost one century are not longer applicable. While the Poisson process has provided a sound analytical model for the POTS, the same model fails applied to model Internet traffic behaviour [2]. Various reasons contribute to that *Failure of Poisson Modelling* [3] and can be theoretically abstracted by *long-range-dependence* (LRD). In short, the discrepancy lies in burstiness of traffic over different timescales. While traffic following a Poisson process "smoothes-out" over large timescales, Internet traffic exhibits extreme variability [2]. More specifically, LRD traffic correlates over large scales and its autocorrelation function is non-summable. Contrary to

[1] Internet Engineering Task Force
[2] Plain Old Telephone Service

Poisson traffic and its well-known exponential decay, LRD traffic can parsimoniously be described using so-called *heavy-tailed distributions*, i.e. distributions following a power law [2][4]. [5] provides a good overview about Internet traffic and corresponding probability distributions.

It is evident that finding theoretical models for such a highly stochastic traffic nature is difficult and fraught with limited accuracy. Therefore, parameter based AC with its obligatory demand on *a priori* information can *not* be applied with reasonable degree of efficiency. To overcome this issue, MBAC was introduced. Instead of using statistical source models, the method of this approach is realtime measuring of QoS parameters as substitution for *a priori* source characterisation. Consequently, MBAC became an integral part of the first standardised Internet QoS architecture Integrated Services [6].

Naturally, sampling a certain QoS parameter at one instant in time has less meaning due to extreme variability of Internet traffic. Thus, various mathematical models derived from different areas were developed with the objective to provide highly precise QoS parameter estimations in combination with measured samples. A popular approach is headed *Equivalent Bandwidth* (EB) and is mathematically founded on Hoeffding bounds. The equivalent bandwidth of a set of flows is considered as a random variable $C(\epsilon)$, which will be exceeded in the future with probability at most ϵ [7]. An extended approach of EB was introduced by [8] where not only bandwidth but also delay is constrained. MBAC based on queue occupation and bandwidth measurement in combination with exponential averaging (EMAVG) is presented in [9]. A similar approach, measuring average queue length, drop rate and link utilisation, was proposed in [10] also with EMAVG to smooth temporal peaks in samples. This proposal, however, differs from others in the scope of measurements. In contrast to the previously listed proposals, this scheme fetches measurements from all nodes in a QoS domain. Thus, the AC algorithm can determine the availability of a path, which is capable to comply requested QoS boundaries.

All presented MBAC schemes have one property in common: measuring is performed locally by each node. In contrast to this method and for the sake of completeness we finally mention *Probe Based* AC (PBAC), which also belongs to MBAC. PBAC is performed by any node, e.g. an end-point running a streaming application by sending so-called probing packets from the source to the destination *probing* the network for its state. In [11] PBAC is applied by sending probe packets with fixed inter-departure times. At the destination node jitter is measured and if predefined thresholds are met, a feedback packet is send back, admitting the requesting flow.

Finally, we point the interested reader to [12] and [13] for an implementation based performance comparison of various AC algorithms.

3 MBAC for the LCT-QoS Framework

The organisation of this section is as follows: after introducing the LCT-QoS framework in Sect. 3.1, we describe the applied MBAC algorithm and its interconnection with components of LCT-QoS in Sect. 3.2

3.1 The LCT-QoS Framework

The Laboratory of Communications and Telematics of UC (LCT-UC) pursues a project whose main goal is to investigate an alternative IP service model. The central idea behind the proposed model is still to treat the traffic using the classical best effort approach, but with the traffic divided into several classes instead of a single class. This corresponds to a shift from a *single-class best-effort* paradigm to a *multiple-class best-effort* paradigm. The classification and forwarding of traffic according to its QoS demands adheres to the DiffServ architecture [1] and is called Per-Hop-Behaviour D3[3] (PHB-D3). To achieve these objectives, the framework is composed of different layers. The lowest layer is an adaptive queuing module used to support the multiple-class-best effort model referred to before [14]. The LCT-QoSR module is placed on top and dynamically routes traffic classes within a QoS domain [15]. Eventually, there is an admission control entity (LCT-QoS-AC) on the network edges as a way of managing system resources which is addressed by this and following work.

Fundamental to all layers is a novel QoS metric, which defines degradations and superfluity zones which, in turn, define how the impact varies with QoS characteristics variations [16]. According to this metric, the quality of service level is quantified through a variable named congestion index (CI), where a high CI value means low quality and vise versa. For each class, there will be a CI related to transit delay and a CI related to loss. The concept of degradation slope (DSlope) is used by the metric for the definition of the classes', i.e. all sensitivity to delay and loss degradation. A traffic class highly sensitive to QoS degradation for a given QoS characteristic will have a high DSlope associated to that characteristic. Figure 1 refers to three classes with different sensitivities to delay degradation (it would be the same if we were talking of loss). Classes with lower DSlope (measured in degrees) will be less sensitive to degradation, so their CI will grow slowly. Using this metric one can say that resources are being shared in a fair way when the different classes have the same CI value related to delay and the same CI value related to loss. In fact, in this case, one can say that the impact of the degradation on applications is barely the same for all of them, despite the different absolute values.

To integrate this concept for differentiated service provisioning, a monitor, which is a part of the queuing module, continuously measures average package transit delay and loss for each class, and calculates the correspondent congestion indexes (CI for delay and loss). These samples are then reported to the involved modules. Contrary to static allocation of resources, the scheduler of the queuing

[3] D3 stands for Dynamic Distribution of Degradation

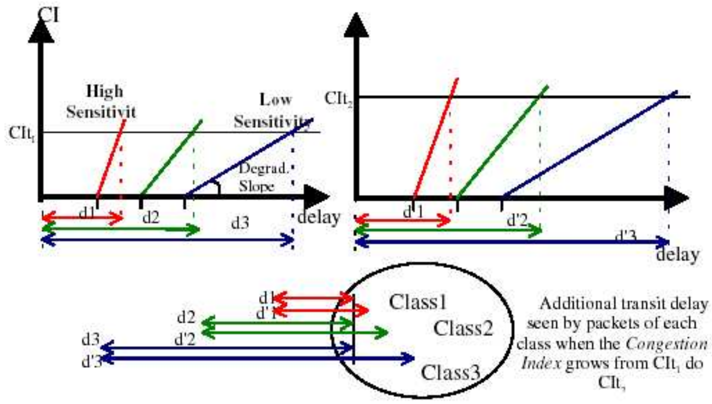

Fig. 1. Congestion Index Calculation for Delay and three Traffic Classes.
With an increasing absolute value of measured delay the associated CI increases also, with a rate depending on the degradation slope (DSlope). Thus, the classes sensitivity to delay can be expressed through the DSlope concept and further, the QoS level of a traffic class is expressed through the corresponding CIs.

module then slows down the forwarding of packets of some classes and speeds up others, respectively, the dropper provides more memory to some queues removing it from others, both in a independent way. The criterion to rule the dynamic distribution of resources is, as mentioned before, the equalisation of CIs.

3.2 Delay and Loss Aware MBAC Algorithm

The role of an Admission Control Entity (ACE) in a service constrained environment can be *generally* defined as follows:

Definition 1. *An Admission Control Entity (ACE) incorporates a decision algorithm to determine whether to reject or admit a new consumer on a link on which admitted consumers compete for finite and shared resources. The admission is positive if, and only if the traffic characteristics, respectively resource demands of the requesting consumer can be accommodated* without *violating the QoS commitments, made to already admitted consumers as well as its own QoS demands, in both cases for their residual lifetime.*

Based on this general definition we derived a tailored algorithm for LCT-QoS-AC. The initial motivation of our approach is the existence of delay and loss measurements performed by the monitor module as part of the PHB-D3 implementation, recall Sec. 3.1. The focus of this first approach is to examine the practicability of these measurements as an indicator of the state of the queueing module and thus, for resource availability. Henceforth, certain peculiarities claim for discussion.

i.) Intuitively, a first idea was to use the same metric, i.e. the CIs as performance parameters. Since CIs are abstractions of delay and loss measurements, however, it turned out to be more convenient use the latter directly.

ii.) LCT-QoS is class-based, i.e. the measurements and thus the decision arguments can not provide strict per-flow but only statistical indicators for flows of a certain class. As a result, a single flow could experience temporal QoS degradation. This is however, fully compliant with the *multiple-class best-effort* paradigm of LCT-QoS and a general feature of class based QoS frameworks.

iii.) Another peculiarity is the focus on delay and loss as QoS parameter. This is founded in human perception and the growing number of streaming content in todays Internet traffic. However, delay and loss are only from minor value for the evaluation of TCP traffic cause of the inherent, adaptive congestion control of TCP. A more detailed investigation of this effect is presented in Sec. 2.

Considering the above mentioned points we developed a *two-staged* algorithm, named *Second Chance* algorithm. The following paragraph presents the algorithm first informal and afterwards for the sake of completeness, also formally.

In the first stage, called *charging stage*, the decision argument is solely based on bandwidth and *a priori* knowledge, i.e. peak rates of traffic sources. Using this information, we simply calculate the number of currently active flows plus one for the requesting source, times the peak pre-defined rate for the associated traffic class. The result is assumed as the required bandwidth for this traffic aggregate. If this value is below or equal to the reserved rate for this class, the new flow is admitted; else rejected. This allows to load (*charge*) the network independent from actual traffic nature and network conditions, achievable with negligible computation overhead. This algorithm is well known as *RateSum* or *SimpleSum* and is a parameter based AC algorithm.

Given the fact that the peak rates of traffic sources are usually higher than their sustained rate (e.g. the transmission rate of a Voice over IP (VoIP) application is namely capped by the codec but also depends on user interaction), we expect that the actual physical load will be lower than the registered rate, i.e. the sum of the peak rates, remaining resources left unused. As an adaptive response to these random influences, the admission controller shifts into a second stage, so-called *saturation stage*, whenever a flow would be rejected in charging stage, examining the admission based on delay and loss measurements. If delay and loss boundaries for all classes are in predefined boundaries, the admission decision is positive, irrespective the negative decision in charging stage, and an additional flow can be admitted. Contrary to charging stage, saturation stage is a pure feedback based decision.

Recall the dynamic distribution of resources in the queuing module. The conceptual rationale for our approach is an AC algorithm, which additionally includes state information of the queueing module. This is contrary to common approaches where congestion control on AC level is clearly separated from that on queueing level. Further, we accept the loss of packets, however, in controlled quantities. This is justifiable by the fact that many streaming sources are rather tolerant to occasional packet loss. The decision finding logic can formally be expressed as:

$$A_{k,m} = \begin{cases} \geq 0 & \text{admitt flow k} \\ < 0 & \text{reject flow k} \end{cases} \tag{1}$$

where $A_{k,m}$ denotes the cumulative admission criterion of flow k of class m and is defined as

$$A_{k,m} = \min(\Omega_s) \tag{2}$$

where Ω_s is the set of QoS parameters in stage s

$$\Omega_s = \begin{cases} \widehat{q}_{n,i} - ((\sum_{f=0}^{F} \widehat{p}_{f,i}) + \widehat{p}_{k,i}) & \text{if } s = n = 0,\ i = m \\ \widehat{q}_{n,i} - q_{n,i} & \text{if } s = 1,\ n = \{1, 2\},\ i = \{0, 1, \ldots, I\} \end{cases} \tag{3}$$

In (3), QoS parameters are denoted by $q_{n,i}$ with $n = 0$ for bandwidth, $n = 1$ for delay and for loss $n = 2$. Index i stands for the associated traffic class. Admission thresholds (predefined maximal values of QoS parameters) are denoted by $\widehat{q}_{n,i}$. Parameter F holds the present number of active flows. Further, to define the stage of the algorithm, $s = 0$ for charging stage and $s = 1$ in saturation stage.

Using this decision logic as the core, the whole ACE consits of two modules, a flow management module and a decider. The flow management module classifies and detects new, inactive, admitted and reject flows, where the decision module continuously requests delay and loss measurements from the queuing module as well as information about inactive flows from the flow management module. For each new flow arrival, the flow management module request admission from the decider and dependend on the decision, adds the flow to a corresponding list.

4 Experimental Results

To evaluate the presented second chance AC we developed a module for the NS[4] network simulator. The results of this evaluation are presented in this section. First, we introduce the general experimental setup, and afterwards the results under varying conditions.

4.1 General Experimental Setup

The topology we use consists of an core and access network resembling an general QoS domain, see Fig. 2. Common for such topologies, the bottleneck link is between the access network, i.e. ingress router, and the core network respectively a core router. Admission Control is performed at the ingress router. Traffic sources are evenly distributed among source nodes and applications are chosen in compliance with the LCT-QoS service model. Table 1 contains the class, delay and loss sensitivity, the application and the source models used to simulate the applications. For streaming traffic (ST) we choose either On/Off sources with Exponential (EXPOO) or Pareto (POO) distributed On/Off times, or constant

[4] NS-2 Network Simulator http://www.isi.edu/nsnam/ns/

bit rate sources with some random noise influencing the scheduling of packets (CBRR). For elastic traffic (ET) solely FTP over TCP is used. The duration is fixed to 3600 simulated s, which we expect to be enough for POO source aggregates to exhibit LRD [12]. The analysis starts after 300 s warm-up. (t_w), the time the system needs to enter equilibrium. Further, only flows which start and end between t_w and the simulation end are considered. The arrival pattern of flows is modelled by a Poisson process with $\lambda = 2.4$ and $\lambda = 0.6$ for ET and ST reflecting realistic Internet traffic distribution. Flow lengths are exponentially distributed for FTP, CBRR and EXPOO sources and log-normal for POO. For all traffic patterns, the transmission rate of video sources is 0.125 Mbit/s and for VoIP 0.0625 Mbit/s. These values correspond to the rate on which On/Off sources emit packets in On state.

The settings for the ACD module are as follows. In charging stage the target transmission rate (R_t) for ET is 8.0 Mbit/s and for both streaming sources 1.0 Mbit/s, corresponding to a bottleneck speed of 10 Mbit/s. In saturation stage we set the delay threshold (D_t) 80 ms for ET, and 10 ms as well as 5 ms for ST, assuming 300 ms RTT and 10-15 intermediary nodes from source to destination as typical for the Internet. Respectively, for the local drop threshold (L_t) we set 10, 5 and 10 percent for each class. The quantification of delay and loss was inspired by [17]. Delay and loss measurements are taken with 100ms frequency like proposed in [14]. In the flow management module, the flow activity timeout is set to 10 s.

According to the delay and loss sensitivities listed in Tab. 1, column two and three, the degradation slopes of the PHB-D3 module are set to 37 degree for high, 36 degree for medium and 33 degrees for low sensitivity, see Sec. 3.1.

Finally, for every scenario (A,B,C) we run three simulations in a row, one for each traffic pattern (X,Y,Z).

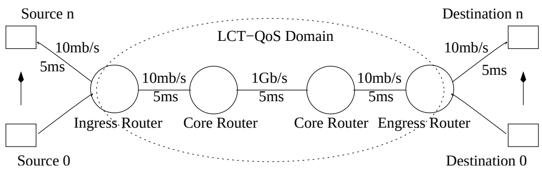

Fig. 2. Simulation Topology

4.2 Simulation Scenarios

Scenario A: Initially, we applied the AC algorithm as described in Sect. 3.2 to manage all traffic classes. Temporal peaks in delay and loss measurements at time t are filtered using a Moving Average with $J = 100$, chosen empirically (4).

$$q_{n,i,t} = 1/J \sum_{j=0}^{J-1} q_{t-j}. \tag{4}$$

Table 1. LCT-QoS Service Model

Type/Class	Delay	Loss	Application	Pattern-X	Pattern-Y	Pattern-Z
Elastic/0	Low	Low	File Transfer	FTP/TCP	FTP/TCP	FTP/TCP
Streaming/1	Medium	High	Video Stream	CBRR/UDP	CBRR/UDP	EXPOO/UDP
Streaming/2	High	Medium	VoIP	POO/UDP	EXPOO/UDP	POO/UDP

The results are presented in Tab. 2. The abbreviations are, charging stage target rate R_t, avg. registered rate (R_r), avg. assigned, i.e. measured rate (R_m) and per-flow throughput (T_f), all in Mbit/s. Local delay threshold (D_t), avg. measured delay (D_m) in s. Local loss threshold (L_t), total drops (L_r) counted at the router, and drops after t_w (L_w) counted at destinations (i.e. either retransmissions for TCP or lost packets for UDP) both in percentage. Finally, flows completed (F_c) between t_w and simulation end, avg. flows admitted (F_a) and avg. flows blocked (F_b). In general, and simply due to space limitation, not all parameters can be discussed in each simulation. However, to provide additional indicators, the whole set is presented for every scenario.

Table 2. Results Scenario A for Traffic Pattern X

Source	Class	R_t	R_r	R_m	T_f	D_t	D_m	L_t	L_r	L_w	F_c	F_a	F_b
FTP/TCP	0	8.0	33.45	8.09	0.020	0.080	0.058	10.0	**18.1**	**20.3**	3873	268	163
CBRR/UDP	1	1.0	1.22	1.08	0.122	0.010	0.0022	5.0	2.7	2.8	258	10	3
POO/UDP	2	1.0	1.00	0.37	0.024	0.005	0.0015	10.0	0.6	0.6	443	16	2

As expectable, this simulation scenario reveals an important fact. Since the majority of flows are elastic (TCP) sources, mostly those are additionally admitted in charging stage, see the difference between R_t and R_r also illustrated by Fig. 3. Recall that the LCT-QoS model distributes resources dynamically according to delay and loss measurements controlled by the respective sensitivity. Due to their inherent, adaptive nature, however, TCP sources share resources fairly. Thus, the resource distribution keeps almost constant since the resource demand does not increase sharply enough. Eventually, enough TCP sources are admitted and their cumulative probing for resources is sufficient to cross the drop threshold. see L_t, L_r and L_w. In this state, while the system goodput is very high, a single TCP connection, however suffers low bandwidth, see T_f and Fig.4. It is a matter of fact that single TCP sources need a minimum bandwidth for acceptable performance [18]. As a result of this evaluation, we limited the saturation stage admission to streaming sources. Since this observation is valid for all traffic patterns, we refrained from presenting the results for the remaining traffic patterns.

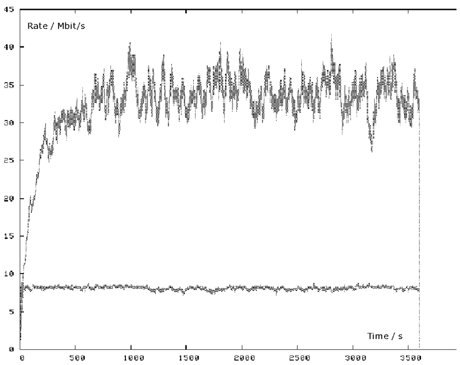

Fig. 3. Elastic traffic registered (top) and assigned throughput rate (down). Offered load is much higher than the available capacity

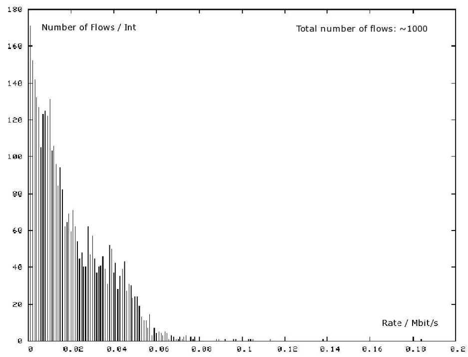

Fig. 4. Elastic traffic per-flow throughput frequency. The majority of the flows suffer too low bandwidth cause of oversubscription

Scenario B: Using the identical setup as before, we evaluated the performance of the modified algorithm. Tab.3 lists the outcomes for all patterns.

Again, the difference between R_t and R_r is a quantitative indicator for flows admitted in charging stage, where now the majority is consumed by class 1, e.g. up to 4 times more than R_t for pattern Z. Moreover, comparing R_t and R_m acknowledges the performance of the PHB-D3 where resources are shared by priority and temporal demands. In contrast to this benefit, except for class 2 in pattern Z, bound violations for delay and loss are experienced, emphasised using bold font. This inaccuracy of the AC algorithm is due to measurement interval length and the lack of schedulability prediction. More precisely, each delay and loss measurement is valid for one measurement interval, and thus arrivals within one interval are admitted based on equal criteria. The linearity of admission in a single interval implies that if the latest value of a single criterion is close to its threshold, a burst of flow arrivals can cause an oversubscription, which in turn leads to bound violations. This is further problematic if the time elapsed until the impact of a flow is reflected in measurements is large.

Another conceptual issue is the flow activity and the corresponding timeout. As explained in Sect. 3.2 the ACE keeps records about flow activity, so if a certain flow is inactive for a given period its admission will be revoked. This is, however, a sensitive parameter. A short timeout means a more reactive AC algorithm, but also an increase of communication interruption probability. A too large setting supports oversubscription since inactive flows mean low delay and loss measurements, which in turn leads to positive admission decisions. If after a time of inactivity, many admitted flows become simultaneously active again, competing for resources with the additionally admitted flows, the PHB-D3 responds with a dynamic redistribution, which leads to bound violations.

Table 3. Results for Scenario B with Traffic Pattern X, Y, Z, top down

Source	Class	R_t	R_r	R_m	T_f	D_t	D_m	L_t	L_r	L_w	F_c	F_a	F_b
FTP/TCP	0	8.0	8.0	6.16	0.085	0.080	0.068	10.0	8.5	9.4	489	64	145
CBRR/UDP	1	1.0	3.20	2.94	0.117	0.010	**0.013**	5.0	**7.0**	**7.1**	83	26	4
POO/UDP	2	1.0	1.09	0.43	0.018	0.005	0.001	10.0	0.8	0.7	130	18	4

Source	Class	R_t	R_r	R_m	T_f	D_t	D_m	L_t	L_r	L_w	F_c	F_a	F_b
FTP/TCP	0	8.0	7.99	6.11	0.084	0.080	0.068	10.0	8.5	9.4	515	64	145
CBRR/UDP	1	1.0	3.16	2.90	0.117	0.010	**0.014**	5.0	**6.8**	**6.9**	85	25	4
EXPOO/UDP	2	1.0	1.30	0.52	0.026	0.005	0.002	10.0	1.0	1.0	68	21	4

Source	Class	R_t	R_r	R_m	T_f	D_t	D_m	L_t	L_r	L_w	F_c	F_a	F_b
FTP/TCP	0	8.0	8.0	6.21	0.084	0.080	0.0721	10.0	8.3	9.3	502	64	144
EXPOO/UDP	1	1.0	5.36	2.79	0.066	0.010	0.0097	5.0	**8.2**	**8.5**	154	43	3
POO/UDP	2	1.0	1.31	0.53	0.019	0.005	0.002	10.0	1.0	0.9	135	21	4

Scenario C: To overcome that issue, we refined the saturation stage admission so that each time a flow was rejected all subsequent flows will also be rejected as long as no previous admission expires, since the latter implies a release of occupied resources. That means, the collection of admission criteria is augmented by another argument, a Boolean flag indicating whether to check for saturation admission or not. To examine the efficiency of this modification, we used exactly the same simulation setup than before. Results are presented in Tab. 4.

It is evident, that the extended AC algorithm its reactivity suffers and is aligned to flow end margins. Juxtaposing the results, however, discloses some interesting facts. Where in scenario A the AC algorithm exhibits large inaccuracy, the modified version does comply delay (D_t), and approaches all loss bounds L_r and L_w with much lower positive deviation, however, still in an order between 10 to 20 percent of the given threshold. We expected a more conservative admission behaviour as a consequence of the last modification, and indeed this is confirmed and quantified in F_c, F_a and F_b. However, no strong degradation is determinable for F_c and F_a. The more reserved response to arrivals is also expressed in R_r. As a means to provide a visual impression we defined a performance index as $I_p = (R_r * L_t)/(R_t * L_w)$. A larger I_p means a better saturation stage admission gain to bound violation ratio as a smaller one. In Fig. 5 and Fig. 6, left bars in a couple represent scenario B and right bars scenario C.

As the last scenario proofed, the performace of the algorithm could further be increased. Obviously, the goal, the development of an AC algorithm, which cooperates with the adaptive queuing module in a self-contained, requirements preserving way, could be closely achieved. The algorithm allows to saturate the link, in all scenarios the link utilisation is almost 95 percent (compare the sum of column R_m with the bottleneck speed), while delay and loss boundaries for the

Table 4. Results for Scenario C with Traffic Pattern X, Y, Z top down

Source	Class	R_t	R_r	R_m	T_f	D_t	D_m	L_t	L_r	L_w	F_c	F_a	F_b
FTP/TCP	0	8.0	8.0	6.52	0.089	0.080	0.073	10.0	7.8	8.6	499	64	143
CBRR/UDP	1	1.0	2.80	2.60	0.118	0.010	0.004	5.0	**5.8**	**6.0**	82	22	4
POO/UDP	2	1.0	1.03	0.41	0.019	0.005	0.001	10.0	0.6	0.6	122	17	4

Source	Class	R_t	R_r	R_m	T_f	D_t	D_m	L_t	L_r	L_w	F_c	F_a	F_b
FTP/TCP	0	8.0	8.0	6.51	0.087	0.080	0.073	10.0	7.7	8.6	529	64	141
CBRR/UDP	1	1.0	2.72	2.53	0.118	0.010	0.048	5.0	**5.6**	**5.6**	85	22	4
EXPOO/UDP	2	1.0	1.23	0.49	0.025	0.005	0.002	10.0	0.8	0.8	78	20	4

Source	Class	R_t	R_r	R_m	T_f	D_t	D_m	L_t	L_r	L_w	F_c	F_a	F_b
FTP/TCP	0	8.0	8.0	6.68	0.090	0.080	0.072	10.0	7.6	8.4	497	64	141
EXPOO/UDP	1	1.0	4.36	2.34	0.068	0.010	0.004	5.0	**5.7**	**5.8**	127	35	3
POO/UDP	2	1.0	1.30	0.52	0.019	0.005	0.002	10.0	0.8	0.8	139	21	4

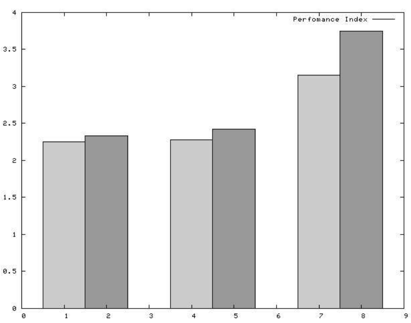

Fig. 5. Class 1 Performance Index comparison for traffic pattern X-Z, left to right

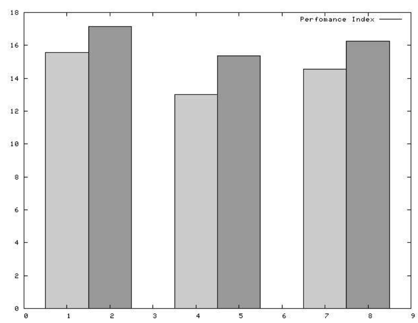

Fig. 6. Class 2 Performance Index comparison for traffic pattern X-Z, left to right

last scenario are complied with acceptable deviation. We justify this acceptance considering the simplicity of our approach and would like to stress that the performance of our AC algorithm could be further improved by incorporating more sophisticated smoothing filters.

5 Conclusion

In this article we presented an effective MBAC algorithm for the LCT-QoS. Starting from a parameter based AC with its well known limitations, we developed an effective AC by analysing and identifying the main issues and subsequent consequent refinements. By exploiting the adaptive nature of the queuing module with its inherent delay and loss measurements, our algorithm performs with high efficiency and acceptable imprecision, juxtaposed with its limited computation resource demand.

However, the simulation results also indicate that MBAC based on pure measurements is doomed to limited precision. With the lack of prediction it is difficult to gain perfect control about resource management, the aim of AC. Nonetheless, we achieved our goal to track down the issues and the practicability of delay and loss measurements as well the dependencies to the PHB-D3. We also conclude, that considering the myriad tuning knobs, we need to capaciously examine the algorithm also incorporating more sophisticated smoothing filter like Exponential Moving Averaging to verify efficiency based on comparison. Another future research direction would investigate the feasibility of a mathematical model of the PHB-D3, which in turn would allow extend the presented approach, or to develop a novel AC algorithm.

As a subsequent step, and depending on the results of the latter, we plan to consider not only local load estimations, but also information spread by the LCT-QoSR layer to process AC based on domain scope.

Acknowledgements
This work was partially supported by the Portuguese Foundation for Science and Technology (FCT), and by European Union FEDER under program POSI (Project QoSMAP).

References

1. S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang and W. Weiss, An Architecture for Differentiated Services, Request For Comments 2475, December 1998. http://www. ietf.org/rfc/rfc2475.txt?number=2475
2. W. Willinger and V. Paxson, Where Mathematics meets the Internet. Notices of the American Mathematical Society, 45(8), Aug. 1998.
3. V. Paxson and S. Floyd, Wide-area Traffic: The Failure of Poisson Modeling IEEE/ACM Transactions on Networking, pp.226-244, June 1995.
4. F. Herndez-Campos, J. Marron, F. Smith, G. Samorodnitsky, Variable Heavy Tailed Durations in Internet Traffic, Part I: Understanding Heavy Tails 10th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems October 11 - 16, 2002.

5. M. Fischer, T. Fowler, Fractals, Heavy-Tails, and the Internet.
6. R. Braden, D. Clark and S. Shenker, Integrated Services in the Internet Architecture: an Overview, Request For Comments 1633, June 1994. http://www.ietf.org/rfc/rfc1633.txt?number=1633
7. S. Floyd, Comments on measurement-based admissions control for controlled-load services. Technical report, Lawrence Berkeley Laboratory, July 1996.
8. Z. Zhifei, Q. Zhengding, A Novel Approach for Real-Time Equivalent Bandwidth Estimation ICCT 2000.
9. J. Kim and Y. Lee, A Dynamic Admission Control Scheme in a DiffServ Domain Information and Communications University, Korea Workshop on High Performance Switching and Routing (HPSR 2002).
10. S. Krasser, H. Owen, J. Grimminger, H. Huth, J. Sokol, Online Traffic Engineering and Connection Admission Control Based on Path Queue States, IEEE Southeastcon 2004, Greensboro, North Carolina, USA, pp. 255-260, March 2004.
11. G. Bianchi, F. Borgonovo, A. Capone, L. Fratta, C. Petrioli, Endpoint admission control with delay variation measurements for QoS in IP network, SIGCOMM Computer Communication Review archive, Volume 32, Issue 2 (April 2002) P. 61-69.
12. S. Jamin, S. Shenker, and P. Danzig, Comparison of Measurement-based Admission Control Algorithms for Controlled-Load Service. In Proceedings of IEEE INFOCOM '97, Kobe, Japan, April 1997.
13. A. Moore, An Implementation-Based Comparison of Measurement-Based Admission Control Algorithms, Journal of High Speed Networks, 13(2), 2004, pp87-102, IOS Press.
14. G. Quadros, A. Alves, E. Monteiro, F. Boavida, An Approach to Support Traffic Classes in IP Networks, Proceedings of QofIS 2000.
15. M. Oliveira, J. Brito, B. Melo, G. Quadros, E. Monteiro, Quality of Service Routing in the Differentiated Services Framework, SPIE s Symposium on QoS and Future Directions, Boston, Massachusetts, USA, Nov 5-8, 2000.
16. E. Monteiro, F. Boavida, G. Quadros, and V. Freitas, Specification, Quantification and Provision of Quality of Service and Congestion Control for New Communication Services, Proceedings of the 16th AFCEA Europe Brussels Symposium & Exposition , AFCEA/IEEE ComSoc, Les Pyramides, Brussels, Belgium, 1995.
17. T. Kostas, M. Borella, I. Sidhu, G. Schuster, J. Grabiec, and J. Mahler, Real-time voice over packet switched networks. IEEE Network, 12(1):18–27, Jan/Feb 1998.
18. R. Mortier, I. Pratt, C. Clark, S. Crosby, Implicit Admission Control, IEEE Journal on selected areas in communications, vol. 18, no. 12, December 2000.