

Managing Services Quality through Admission Control and Active Monitoring

Solange Lima, Paulo Carvalho, Alexandre Santos, and Vasco Freitas

Universidade do Minho, Dept. de Informática, 4710-057 Braga, Portugal
{solange,paulo,alex,vf}@uminho.pt

Abstract. We propose a lightweight traffic admission control scheme based on on-line monitoring which ensures multimedia services quality both intra-domain and end-to-end. The AC strategy is distributed, service-oriented and allows to control QoS and SLS without adding complexity to the network core. For each service class, AC decisions are driven by rate-based SLS control rules and QoS parameters control rules, defined and parameterized according to each service characteristics. These rules are essentially based on systematic on-line measurements of relevant QoS and performance parameters. Thus, from a practical perspective, we discuss and evaluate methodologies and mechanisms for parameter estimation. The AC criteria is evaluated as regards its ability to ensure service commitments while achieving high network utilization. The results show that the proposed model provides a good compromise between simplicity, service level guarantee and network usage, even for services with strict QoS requirements.

1 Introduction

The deployment of multimedia services in the Internet has been fostering the adoption of QoS models and related traffic control mechanisms in order to handle different applications QoS requirements while using network resources efficiently. Class of service (CoS) networks, such as Diffserv, are a step forward in pursuing this objective, where flows with similar characteristics and service requirements are aggregated in the same class. Controlling the admission of flows sharing a class allows to support new traffic flows conveniently without compromising existing QoS commitments. An admission control (AC) strategy should consider three vectors: (i) assurance level; (ii) control complexity; and (iii) network resources usage. Overprovisioning is currently the most common way to provide QoS guarantees in network backbones. Although for some ISPs overprovisioning is an attainable solution, in general, it is either not available or a solution too expensive. In our opinion, some degree of overprovisioning is recommended so that the AC process can be relaxed and simplified. The level of service guarantee to be provided is closely related to the complexity of the underlying traffic control strategy. In fact, either using centralized or decentralized AC approaches, the provision of guaranteed services, e.g. for hard real-time traffic, involves controlling the state and load of traffic aggregates in the core nodes [1–3]. These solutions tend to require significant network state information and, in many cases, changes in all network nodes [3]. Furthermore, as they are closely tied to network topology and routing, their complexity increases with the network dynamics. The provision of predictive services,

e.g. for soft real-time traffic, allows a flexible AC, reducing AC control information and overhead. In this context, measurement-based AC (MBAC) solutions involving all network nodes [4], or only edge nodes [5–7] (EMBAC) have deserved special attention. However, while leading to more efficient resource utilization, QoS degradation may occur. The need to control elastic traffic, for more efficient network utilization, has also been discussed and implicit AC strategies have been defined [8].

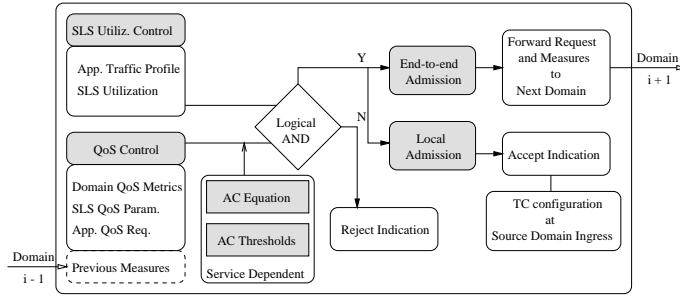
A further step towards a generic and light AC model oriented to multi-service networks, able to operate both intra-domain and end-to-end has been proposed in [9]. The present work details the main components of this model (section 2) focusing on its implementation (section 3). The tuning and performance of the AC model is evaluated as regards the underlying measurement process and effectiveness of AC rules (section 4).

2 The Admission Control Model

The AC model proposed in [9] considers: (i) the control of distinct network services and assurance levels, to handle different application QoS requirements and traffic profiles; (ii) the operation intra-domain and end-to-end, controlling both the QoS levels in a domain and the sharing of the existing SLS between domains to fulfill end-to-end QoS requirements. As explained in [9] ingress routers perform explicit or implicit AC depending on the application type and corresponding traffic class, while egress routers perform edge-to-edge on-line QoS monitoring and SLS control. QoS Monitoring measures relevant parameters for each service (service metrics) reflecting service availability from each ingress. SLS Control monitors the usage of downstream SLSs at each egress to ensure that traffic to other domains does not exceed the negotiated profiles and packet drop will not occur due to an indiscriminate traffic conditioning (TC) process. QoS monitoring statistics, SLS utilization and associated parameters are then sent to the corresponding ingress routers to update an Ingress-Egress service matrix used for distributed AC and active service management.

Explicit and Implicit AC - Explicit flow AC is oriented to applications able to signal the network with their traffic profile and QoS objectives, e.g. streaming applications. In this case, the AC decision requires two initial verifications (see Fig. 1): (i) SLS Utilization Control checks if the downstream SLS can accommodate the traffic profile of the new flow; (ii) QoS Control checks if, for the corresponding egress node and service, the domain QoS metrics, the SLS QoS parameters agreed with the downstream domain and the previous measures (if any) fulfill the application QoS requirements.

Each AC decision is based on a service dependent AC equation and thresholds defined to ensure specific service guarantees. When a flow is accepted in the domain, the notification may be generated either locally (local admission) or remotely (end-to-end admission). The *end-to-end case* [9] is viewed as a repetitive and cumulative process of admission control and available service computation, performed at ingress nodes. At each domain the ingress node decides if a flow can be accepted, and if so the domain service metric values are added to the flow request to inform the downstream domain of the service available so far. Using the incoming and its own measures each domain performs AC. This solution leads to a generic AC model, which can be applied both to source and transit domains.


Fig. 1. Admission Control Criterion

Implicit flow AC, oriented to elastic applications which do not use signaling, use implicit detection of flows [8]. This type of AC, likely to be implemented only in the source domain, will be restricted to SLS information and QoS monitoring.

3 Model Implementation

3.1 Definition of Service Classes and SLS

A differentiated services architecture needs to be supported by an adequate traffic classification strategy [10, 11]. As initial policy, we treat TCP and UDP traffic separately, being UDP traffic further divided according to the applications QoS requirements stringiness. As result, three initial service classes were defined (see Table 1). Service Class 1 (SC1), supported by EF PHB, provides a high QoS performance service guarantee and is oriented to streaming applications imposing hard real-time constraints. Due to the high priority treatment this class requires in each network node, which may starve low priority classes, the access to the corresponding service is tightly controlled. SC1 AC criterion will follow a conservative schema, with TC giving a severe treatment on excess traffic (see section 3.2). Service Class 2 (SC2), supported by AF PHB, provides a predictive type of service with low delay, low loss and minimum bandwidth guarantee and is oriented to streaming applications with soft real-time constrains. SC2 AC criterion will be less conservative, taking more advantage of statistical multiplexing. TC will act on non-conformable traffic using a three-color marker (TSW3CM). In a first set of experiments, only AF1x is considered. Service Class 3 (SC3) provides best-effort service to adaptive TCP applications. Detailed classification rules for TCP differentiation

Table 1. Definition of Service Classes

Serv.Class	Serv.Level	Traffic type	PHB	AC	Policing	Scheduling
SC1	guaranteed	UDP (hard RT)	EF	explicit & conserv.	drop excess	strict priority
SC2	predictive	UDP (soft RT)	AF	explicit & flexible	3 color marker	WRR
SC3	best-effort	TCP	BE	implicit & relaxed	3 color marker	WRR

will be considered in the future taking the remaining AF classes. SC3 AC criterion will be implicit and relaxed. The service classes are implemented resorting to class-based queuing with priority weighted round-robin scheduling.

As regards SLS definition, both SLA template defined in [9] and QoS parameters upper bounds for common applications defined in [12] have been considered here to support SLS instantiation.

3.2 Admission Control Criterion

Establishing an admission criterion consists of defining the rules by which flows are accepted or rejected. Usually AC criteria are parameter-based, measurement-based or follow an hybrid scheme combining both. Parameter-based AC algorithms, oriented to flows requiring a guaranteed service, tend to be conservative leading to low utilization for bursty traffic. Measurement-based AC (MBAC) algorithms [4, 5] are less conservative, taking advantage of statistical multiplexing of traffic to increase network utilization at an eventual cost in QoS degradation. In this way, MBAC is more suitable for flows requiring a predictive service.

In the proposed model, the AC criterion is essentially measurement-based¹ controlling both the QoS in the domain and the downstream SLS utilization. This leads to the specification of two types of rules: (i) rate-based SLS control rules; and (ii) QoS parameters control rules.

Rate-based SLS Control Rules - Let I be the set of ingress nodes, i.e. $I = \{I_1, I_2, \dots, I_N\}$ and E the set of egress nodes in a domain, i.e. $E = \{E_1, E_2, \dots, E_M\}$. For each egress $E_m \in E$ with $1 \leq m \leq M$, one or more SLSs can be in place, one per service type and per downstream domain. At this point, it is assumed a single mapping between a service class within the domain and a downstream SLS for the corresponding service type. As each SLS has a specified negotiated rate, a rate based Measure-Sum (MS) algorithm is applied to control SLS utilization. Thus, for each ingress $I_n \in I$ with $1 \leq n \leq N$, the equation used to verify if a new flow can be admitted takes both rate estimate and flow traffic description, , i.e.

$$\rho_s + r_j \leq \beta_s R_s \quad (1)$$

In (1) ρ_s is the current measured load or estimated rate of flows using SLS_s of $E_m \in E$ considering all the ingress-egress estimated rate for class i going through E_m , i.e. $\rho_s = \sum_{k=1}^N \rho_{i,k}$; r_j is the rate specified by the new flow j ; β_s is the utilization target for the SLS (with $0 < \beta_s \leq 1$); and R_s is the rate defined in SLS_s .

QoS Parameters Control Rules - When controlling QoS levels in a domain, the QoS parameters under control and corresponding thresholds can vary depending on each service class commitments, the statistical properties of the traffic and degree of overprovisioning. A new flow is accepted or rejected by checking the controlled parameters $P(i) = \{p_1, p_2, \dots, p_p\}$ of class i against the corresponding pre-defined threshold t_p , i.e. $\{\forall p \in P(i) : p_p \leq t_p\}$, in which t_p can be affected by a safety margin to the

¹ Some degree of overprovisioning is considered for stringent QoS classes so that simplicity and flexibility of MBAC can be useful to control these classes too.

QoS parameter bound. Tuning these limits, making them useful and realistic indicators of the overall QoS status is a fundamental aspect for AC, as shown in section 5.

Following the service definition provided in section 3.1, AC for SC1 and SC2 uses both type of rules defined above. For SC1, a more conservative criterion is taken, considering the worst-case scenario (flow peak rates, concurrent AC taking place at other ingress nodes and optimistic measures), larger safety margins and tighter thresholds. Table 2 summarizes the AC criteria illustrating the type of QoS parameters under control for SC1, SC2 and SC3, which are then defined in Table 3. Recall that, due to the nature of TCP traffic, where a flow has not a pre-defined rate, Equation (1) is applied as a threshold for the estimated rate. The estimation mechanisms for the parameters under control and the time granularity used in the estimation is discussed in the section 3.3.

From an *end-to-end* perspective, whenever a flow AC request specifies end-to-end QoS requirements, QoS parameter control rules need to be extended to accommodate an additional verification. For each specified parameter, this verification weights the corresponding QoS parameter estimate/bound in the domain, its negotiated value with the downstream domain and the cumulative value computed so far.

Table 2. Admission Control Criteria

Class	Flow Inputs		Network Inputs	SLS Util. Control		QoS Parameter Control	
	T.Desc.	QoS	Measures	Parameter	Method	Parameter	Method
SC1	peak rate	if any	load, IPTD, ipdv, IPLR	rate	MS	IPTD, ipdv, IPLR	thresh.
SC2	mean rate	if any	load, IPTD, IPLR	rate	MS	IPTD, IPLR	thresh.
SC3	n.a.	n.a.	load, IPLR	rate	thresh.	IPLR	thresh.

Table 3. Controlled QoS Parameters

Rate Parameters	
Throughput ρ (bps)	$\rho_i = \sum bits_received_i / \Delta t_i$
Utilization U (%)	$U_i = \rho_i / C$
Delay Parameters (ms)	
IP Transfer Delay ($IPTD$)	$IPTD_i(pkt) = (t_E(pkt) - t_I(pkt))$
Mean IPTD (\overline{IPTD})	$\overline{IPTD}_i(\Delta t_i) = \sum (t_E i - t_I i) / pkts_received_i(\Delta t_i)$
Inst. Packet Delay Var. ($ipdv$)	$ipdv_i = (IPTD_{i,pkt_k} - IPTD_{i,pkt_{k-1}})$
Mean ipdv (\overline{ipdv})	$\overline{ipdv}_i(\Delta t_i) = \sum ipdv_i / pkts_received_i(\Delta t_i)$
Maximum IPTD	$max(IPTD_i(pkt), \Delta t_i)$
Minimum IPTD	$min(IPTD_i(pkt), \Delta t_i)$
IPTD Alarm ($Alarm_IPTD$)	$Alarm_IPTD_i = \overline{IPTD}_i(\Delta t_i) - \overline{IPTD}_i$
Loss parameters (%)	
IP Loss Ratio (IPLR)	$IPLR_i = pkts_lost_i / pkts_sent_i$
Mean IPLR (\overline{IPLR})	$\overline{IPLR}_i(\Delta t_i) = pkts_received(\Delta t_i) / pkts_sent_i(\Delta t_i)$

3.3 On-line Monitoring

In our study, the objective of on-line monitoring is twofold: (i) it allows SLS auditing in the domain and (ii) it provides inputs for the AC decision module, which being measurement based, requires a realistic view of the network status. The systematic use of on-line monitoring for traffic load and QoS estimation, while allowing an active service management avoids the common per application intrusive traffic and the initial latency of EMBAC approaches. Furthermore, the effect of cross-traffic and other internally generated traffic (e.g. routing, management, multicast traffic) is implicitly considered.

The problematic of monitoring involves the definition of metrics, measurement methodologies and timing decisions. ITU-T work on QoS in IP networks and particularly IETF IPPM have defined a set of standard QoS and performance metrics and have proposed measuring methodologies for them [13, 14]. Several tools useful for measuring the SLS metrics have also been developed and tested [12, 15]. Taking these inputs into account, on-line monitoring implementation options are discussed below.

QoS and Performance Metrics - Several edge-to-edge QoS and performance parameters have been identified to be controlled at each egress node. They are specified for an given ingress-to-egress pair (I_n, E_m) , class i and time interval Δt (see Table 3).

Measurement Methodology - A measurement methodology can be either passive, active or combinations thereof. Passive measurements are carried out on existing traffic and are particularly suitable for troubleshooting; active measurements inject extra traffic (probing) in the network for measurement purposes allowing to check QoS and SLS objectives in a more straightforward way. Probing brings an additional advantage when measuring edge-to-edge performance and QoS. As specific packets are injected in the network containing timestamping and sequencing data, delay and loss estimations are simplified. Obtaining these estimates combining link-by-link measures is not an efficient and easy solution. However, as probing is an intrusive process, its impact on the network load needs to be minimized. In the proposed model, in-band probing is used per class and not per application which is a clear advantage over other EMBAC approaches as overhead is reduced. For each class, the parameters in Table 3 are estimated and controlled, resorting to passive and active measurements. Comparing the outcome of both approaches allows to assess and tune the probing process (see section 5.1).

Parameters Estimation - Apart from the measurement methodology, there are several measurement mechanisms which can be used for parameter estimation. In particular, *Time-Window* (TW), *Point Sample* (PS) and *Exponential Averaging* (EA) mechanisms are commonly an option due to their simplicity [16, 4]. Although these mechanisms are usually applied to a single node, we have applied them to edge-to-edge measurements. For *SLS utilization control*, the class traffic load is the estimated parameter, obtained resorting to the estimation mechanisms described above in order to assess which one more closely reflects the real network behavior (see section 5.2). In order to obtain a statistically meaningful number of samples, following [4], $T/S \geq 10$ with $S > 100 * L/C$, where S is the sampling period controlling the measurement sensitivity, T is the window size controlling the mechanism adaptability, L is the packet size (bits) and C the transmission rate. As our estimates are edge-to-edge, dimensioning S considers the one-way delay instead of C . As in [16], for each sampling period S , in-

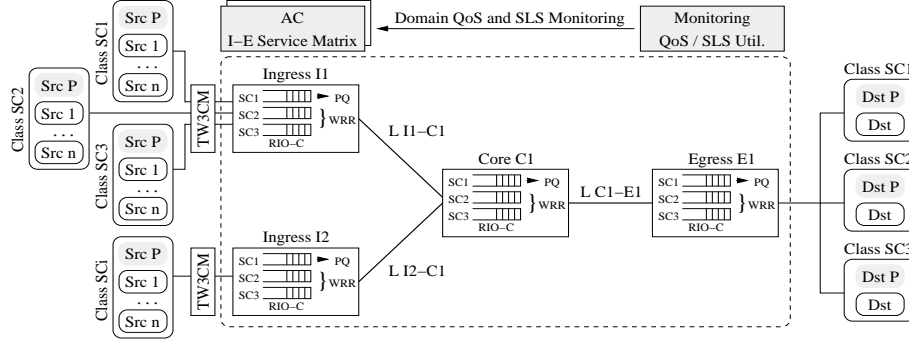


Fig. 2. Simulation topology

independently of which estimation method is in use, the parameter average is evaluated. For *QoS control*, the QoS parameter average in Δt ($\Delta t = S$) is used as estimate.

4 Simulation Scenario

To test the proposed AC model in a multi-class domain, a simulation prototype was devised and set up based on NS-2 platform. The main objectives of the experiments are threefold. First, we intend to assess the active measurement methodology as a whole. Both probing patterns, probing periodicity and probing ability to capture each class behavior are studied. Second, a comparison of the estimation mechanisms TW, Avg_PS and EA is carried out in order to evaluate which one provides the closest estimate to reality, tuning S and T timing parameters. Third, the proposed AC criterion is evaluated as regards its ability to ensure service commitments are not violated, while assessing both network utilization and QoS safety margins.

Simulation topology - The simulation topology is illustrated in Figure 2. The network domain consists of ingress routers I_1 , I_2 , a core router C_1 and an edge router E_1 . While I_1 multiplexes three types of sources, each type mapped to a different class, I_2 is used to inject concurrent traffic. This allows to evaluate concurrency in distributed AC and assess cross traffic impact. The domain internodal links capacity is 34Mbps, with a 15ms propagation delay. $L_{C_1-E_1}$ link works as a bottleneck in this network topology. Access links have been configured so that intra-domain measurements are not affected. In each node, each class queue is 150 packets long. The scheduling discipline follows an hybrid Priority Queuing - Weighted Round Robin (PQ-WRR) and the active queue management mechanism is RIO-C. The PQ-WRR(2,1) discipline gives to the highest priority class (SC1) a strict priority treatment with a tight limit on a pre-defined rate (10% of the bottleneck link), whereas the remaining class queues are served with a 2 to 1 proportionality. At network entrance, each traffic class is policed using a TSW3CM.

Source models - Generically, three source models have been considered: Constant Bit Rate (CBR) sources, Exponential on-off (EXP) and Pareto on-off (PAR) sources. PAR sources with $1 < \alpha < 2$ under aggregation will allow to generate traffic exhibiting

Table 4. Source Parameter Configuration

Class	Protocol	Src Type	Src Parameters	Inter. t	Hold. t
SC1	UDP	CBR_{SC1}	($r=100\text{kbps}$, $l=128\text{B}$)	0.4-2s	60s
	UDP	EXP_{SC1}	($r=200\text{kbps}$, $l=128\text{B}$, $\text{on=off}=500\text{ms}$)	0.4-2s	60s
	UDP	PAR_{SC1}	($r=200\text{kbps}$, $l=128\text{B}$, $\text{on=off}=500\text{ms}$, $\alpha=1.5$)	0.4-2s	60s
SC2 and SC3	UDP	$CBR_{SC2,3}$	($r=0.5\text{Mbps}$, $l=512\text{B}$)	0.4-2s	120s
	UDP	$EXP_{SC2,3}$	($r=1\text{Mbps}$, $l=512\text{B}$, $\text{on=off}=500\text{ms}$)	0.4-2s	120s
	UDP	$PAR_{SC2,3}$	($r=1\text{Mbps}$, $l=512\text{B}$, $\text{on=off}=500\text{ms}$, $\alpha=1.5$)	0.4-2s	120s
SC3	TCP	FTP App.	($r=\text{unspecified}$, $l=512\text{B}$)	0.4-2s	120s
Probing	UDP	CBR_P	($r=1.6\text{kbps}$ (2pkts/s), $l=100\text{B}$)	1 src	sim. dur.
	UDP	POI_P	($r=1.6\text{kbps}$, $l=100\text{B}$, Poisson)	1 src	sim. dur.
	UDP	EXP_P	($r=3.2\text{kbps}$, $l=100\text{B}$, $\text{on=off}=250\text{ms}$)	1 src	sim. dur.

long-range dependence [10]. As this property has a significant impact on queuing behavior and on the nature of congestion leading to unexpected QoS degradation, larger safety margins may be needed. SC1 comprises UDP traffic with small to medium peak rate and packet sizes, as usually generated by some real-time streaming applications, such as VoIP. SC2 comprises UDP traffic with higher peak rate and packet sizes. Initial tests consider UDP traffic in SC3. The flow arrival process is Poisson with exponentially distributed interarrival and holding times. Table 4 details the parameters choice. As regards probing, three in-band source types have been defined. In [17], two packets per second according to a Poisson distribution are used as a probing scheme to assess loss and delay between any two network measurement points. Here, the adequacy of this pattern is tested and compared with both CBR and Exponential on-off probing. The use of EXP-like sources prevent possible synchronization among probing and other events in the IP network [17].

Service and AC Configuration - Table 5 illustrates the main parameters used to configure the AC rules, for controlling both SLS utilization and domain QoS levels. Three downstream SLSs have been considered, one per service class, with a negotiated rate (R_s) defined according to the traffic load share intended for the corresponding class in the domain. The MS algorithm that rules SLS utilization has specific utilization target (β_s) values depending on how conservative the AC decisions must be. For instance, a $\beta_s=0.75$ corresponds to impose a safety margin of 25% to absorb load fluctuations and optimistic measures. This value can be viewed as a degree of overprovisioning. The AC thresholds for QoS control in the domain are set taking into account the domain topology dimensioning, queuing and propagation delays, and perceived QoS upper bounds for common applications and services [12].

5 Simulation Results

In this section, the results are discussed as regards: (i) the probing process (ii) the estimation mechanism and (iii) the AC criteria. The results were obtained running multiple simulations of about 8 minutes, discarding results from an initial convergence period.

Table 5. Service Parameter Configuration

Service Class	SLS Rate R_s (% share)	Util. Target β_s	QoS Parameter	Threshold
SC1	3.4Mbps (10%)	0.75	IPTD,ipdv,IPLR	35ms,1ms, 10^{-4}
SC2	17.0Mbps (50%)	0.90	IPTD,IPLR	50ms, 10^{-3}
SC3	13.6Mbps (40%)	1.00	IPLR	10^{-1}

5.1 Evaluation of the Probing Process

Generically, the probing process is assessed as regards its ability to capture classes behavior realistically, and used for monitoring network status. Initial tests consider three distinct probing sources (CBR, POI, EXP) to measure the parameters defined in Table 3. For each class, the probing measuring outcome was cross-checked against the corresponding measures using the class real traffic.

Despite the traffic type in each class, the results obtained are consistent and similar for the probing sources considered. Our major findings are: (i) probing can be successfully used to evaluate both IPTD and mean IPTD. This is true both for capturing the shape and scale of these QoS metrics. Figure 3 shows an almost perfect match for the mean IPTD. This is valid even for probing rates as low as two packets per second. (ii) for the test conditions, probing is inappropriate for measuring both ipdv and IPLR, as Figure 3 shows. As ipdv is a consecutive packet measure, probing gaps lead to higher measures as consequence of queue occupancy variations. For IPLR, as loss is typically seen as a rare event and probing packets are marked as high priority (green) packets, the probing method is again inadequate. The exception to this behavior, occurs when traffic in a class suffers heavy loss (over 10%). This mis-behavior can be reduced resorting to higher probing rates and using red-marked probe traffic. However, the overhead introduced may be prohibitive, in particular, if it is performed in-band. Due to its particular rate characteristics probing rate cannot be directly compared to class throughput. For bandwidth monitoring, specific probing techniques must be used [18, 19].

5.2 Evaluation of the Estimation Mechanism

In a first instance, the evaluation method consists of determining on how close an estimate is to the real traffic load. In this way, the rate estimation of each service class using the estimation mechanisms TW, Avg_PS and EA is compared (see Figure 4), taking Avg_PS estimates as reference. This is because Avg_PS represents the real aggregate mean rate in a pre-defined interval S .

Usually, TW leads clearly to over estimation of the metric and, in practice, it can be a very conservative method. In special, when the window T is reinitialized upon a new flow admission and for short flow interarrivals, the estimate increases steadily as the departure of flows is not taken into account. The ratio between flow duration and T is studied in detail in [4]. However, this method allows to consider in advance the weight the new admission might have. This also occurs with EA, where the estimation is artificially increased when a new flow is admitted. This estimation method is controlled by the weight parameter γ , where for $\gamma = 0.25$ a closer match is achieved (Fig. 4).

As far as AC is concerned, there are other aspects to consider: (i) the estimate is due to be used during a time interval T and (ii) the estimate needs to reflect, and somehow foresee, the network behavior trends. The tests on AC criteria show that Avg_PS allows to achieve high network utilization without service violations, for CBR traffic. However, for EXP and PAR traffic, all services have suffered disruption. EXP/PAR traffic fluctuations and a particularly low estimate results in over acceptance. When this happens, in the next estimation period, the AC rate and QoS control rules will stop the new flows' entrance, however, degradation occurs during the lifetime of the existing ones.

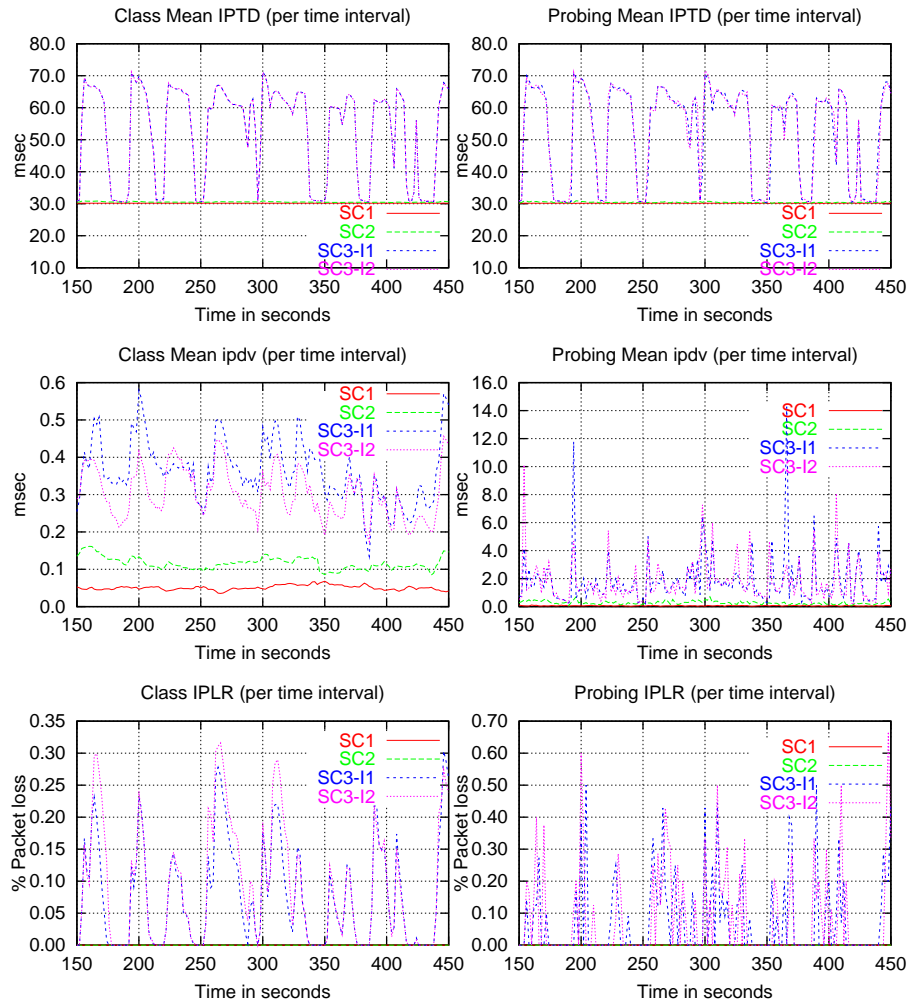


Fig. 3. Class (left) and Probing (right) Mean IPTD, Mean ipdv and IPLR (time interval of 2s).

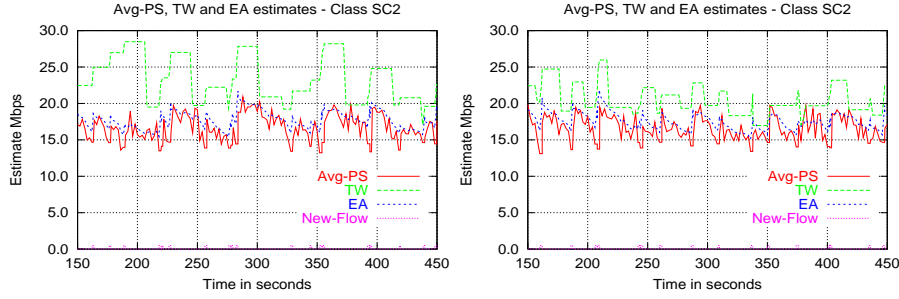


Fig. 4. Estimation mechanisms for ($T=10$; $S=1$; $\gamma = 0.25$; flow (EXP, 500kbps, i.a.t.=0.4s, h.t.=120s) (a) resetting T (b) without resetting T on flow admittance.

This effect can be reduced with TW and EA due to their initial accounting on the flow rate. In fact, an immediate increase on the estimate, reflecting the impact of the new flow will have, allows a more adaptive and conservative AC.

5.3 Evaluation of the AC Criteria

While the active monitoring process is being tuned, passive measurements are used to evaluate the proposed AC criterion so that results are not misleading. The Avg_PS mechanism (with $S = 2s$) which represents the real average values for the parameters under control is used. In these experiments, concurrent SC3 traffic is injected through ingress I2. As expressed in Table 5, SC1 traffic is blocked whether the sum of the rate estimate ρ_s and the flow's peak rate r_j is above 75% of the class share (see Equation 1), or any of the QoS controlled parameters exceed the pre-defined thresholds. For SC2, a safety margin of 10% ($\beta_s=0.9$), was defined and the flow mean rate is used instead. SC3 does not use safety margin but controls IPLR. The tests are performed under high demanding conditions with a flow interarrival of 400ms.

Table 6 summarizes the results obtained for each class and each source type as regards: (i) the average of concurrent active flows; (ii) the percentage of packets exceeding the pre-defined delay bounds; (iii) the total loss ratio; (iv) the new utilization target proposal, for which no packet QoS violations occur and (v) the new average of concurrent active flows. The results obtained show that while for CBR traffic there is no loss and reduced delay violations, for EXP and PAR sources an increasing packet loss is clearly noticed. Although, the AC rules are effective in blocking new flows when QoS degradation or an excessive rate is sensed, the effect of previously accepted flows persists over the next intervals while they last. This over acceptance is caused by traffic fluctuations combined with the type of estimation mechanism used (without prevision). To minimize this, more conservative estimates or larger safety margins are needed. We have explored this last option for EXP traffic, where new safety margins leading to no packet QoS violations were established. Fig. 5 (b) shows that, under the new defined margins, the AC criteria achieves good network utilization. In this tuning process, we found that, in SC3, IPLR is a difficult parameter to control. Even when the total loss

Table 6. AC Test Results

Class	Src Type	#act_flows	%pkts_viol:(IPTD;ipdv)	%IPLR	New Util.Target	new #act_flows
SC1	CBR_{SC1}	26.0	(0.41 ; 0.14)	0.0	0.55	19
	EXP_{SC1}	27.0	(0.33 ; 0.08)	2.2		
	PAR_{SC1}	26.5	(0.30 ; 0.08)	2.9		
SC2	CBR_{SC2}	31.5	(0.05 ; n.a)	0.0	0.75	27.5
	EXP_{SC2}	34.5	(0.07 ; n.a)	1.5		
	PAR_{SC2}	34.0	(0.08 ; n.a)	1.1		
SC3	CBR_{SC3}	16.0+16.0	(n.a. ; n.a)	0.7	0.80	13.5+14.5
	EXP_{SC3}	15.5+17.5	(n.a. ; n.a)	17.4		
	PAR_{SC3}	16.5+19.0	(n.a. ; n.a)	20.2		

is below the defined upper bound, the Mean IPLR per interval may exceed the corresponding threshold. This may be due to the low weight in the scheduling discipline which serves the corresponding queue. The results also shows that SC1 can be particularly affected by the scheduling mechanism. While PQ is suitable and commonly used to handle in-profile high priority traffic, if the aggregate rate exceeds the maximum rate allowed by the scheduler, the class is severely punished. This is evident through an increase of IPTD and IPLR (see Fig. 5 (a)), which stems from a head-of-line blocking while waiting for the scheduling cycle to be completed. This stresses the need of having a tight control on this class and a wider safety margin.

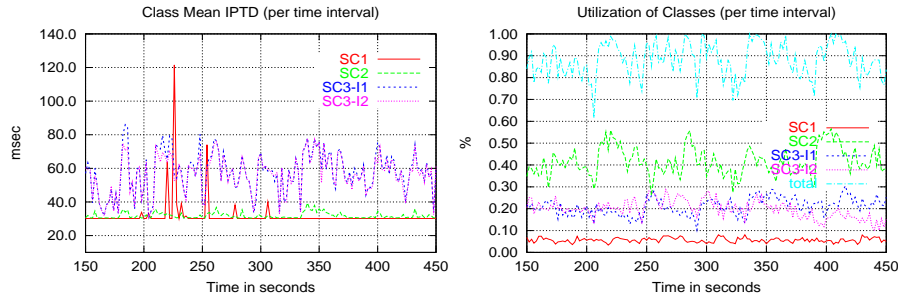


Fig. 5. (a) Mean IPTD (for EXP sources and initial utilization target) (b) Utilization for the new utilization target (considering the bottleneck capacity of 34Mbps)

6 Conclusions

In this paper we discussed and evaluated a distributed AC model for the management of multimedia services quality in class-based networks. We evaluate the two main components of the model: the monitoring process, which controls QoS and SLS parameters,

providing inputs to AC; and the AC criteria which guide the AC decisions. Parameter estimation methodologies and mechanisms are compared and tuned. Both probing patterns and periodicity were assessed as regards its ability to capture the behavior of the different classes. The results show that probing is a good solution to measure IPTD, but did not capture ipdv and IPLR behavior properly. The evaluation of the proposed AC criteria as regards its ability to ensure service commitments shows that using proper AC rules and safety margins the simplicity and flexibility of this measurement-based AC approach can be successfully used to manage service quality.

References

1. B. Teitelbaum, S. Hares, L. Dunn, R. Neilson V. Narayan, and F. Reichmeyer. Internet2 QBone: building a testbed for differentiated services. *IEEE Network*, 13(5):8...16, September/October 1999.
2. Zhi-Li Zhang et al. Decoupling QoS Control from Core Routers: A Novel Bandwidth Broker Architecture for Scalable Support of Guaranteed Services. In *SIGCOMM'00*, 2000.
3. I. Stoica and Hui Zhang. Providing Guaranteed Services Without Per Flow Management. In *ACM SIGCOMM'99*, October 1999.
4. S. Jamin, S. Shenker, and P. B. Danzig. Comparison of measurement-based call admission control algorithms for controlled-load service. In *INFOCOM (3)*, pages 973–980, 1997.
5. L. Breslau et al. Endpoint Admission Control: Architectural Issues and Performance. In *ACM SIGCOMM'00*, 2000.
6. C. Cetinkaya, V. Kanodia, and E. Knightly. Scalable Services via Egress Admission Control. *IEEE Transactions on Multimedia*, 3(1):69...81, March 2001.
7. V. Elek, G. Karlsson, and R. Rnngren. Admission Control Based on End-to-End Measurements. In *IEEE INFOCOM'00*, 2000.
8. R. Mortier and I. Pratt and C. Clark and S. Crosby. Implicit Admission Control. *IEEE Journal on Selected Areas in Communications*, 18(12):2629...2639, December 2000.
9. S. Lima and P. Carvalho and A. Santos and V. Freitas. A Distributed Admission Control Model for CoS Networks using QoS and SLS Monitoring. In *ICC'2003 - IEEE International Conference on Communications*, May 2003.
10. S. Lima and M. Silva and P. Carvalho and A. Santos and V. Freitas. Long-Range Dependence of Internet Traffic Aggregates. *Lecture Notes on Computer Science*, (2345), May 2002.
11. A. Bak, W. Burakowski, F. Ricciato, S. Salsano, and H. Tarasiuk. Traffic Handling in AQUILA QoS IP Networks. In *QoFIS'01*, September 2001.
12. S. Leinen and V. Reijs. D9.7 - Testing of Traffic Measurement Tools, September 2002.
13. Tijani Chahed. IP QoS Parameters. *TF-NGN*, November 2000.
14. V. Paxson, G. Almes, J. Mahadavi, and M. Mathis. Framework for IP Performance Metrics, IETF RFC2330, 1998.
15. CAIDA Tools. <http://www.caida.org/tools/>.
16. S. Floyd. Comments on measurement-based admissions control for controlled-load services. Technical report, LBNL, 1996.
17. F. Georgatos et al. Providing Active Measurements as a Regular Service for ISPs. In *PAM'2001*, April 2001.
18. C. Dovrolis, P. Ramanathan, and D. Moore. What Do Packet Dispersion Techniques Measure? In *IEEE INFOCOM'01*, 2001.
19. C. Dovrolis and M. Jain. 'End-to-End Available Bandwidth: Measurement methodology, Dynamics, and Relation with TCP Throughput. In *ACM SIGCOMM'02*, August 2002.