ANALYSIS AND SIMULATION OF A TRAFFIC MANAGEMENT CONTROL SCHEME FOR ATM SWITCHES WITH LOOSE COMMITMENTS

Nelson Antunes , Rui Rocha , Paulo Pinto
Instituto Superior Técnico, Av. Rovisco Pais, 1 P-1000 Lisboa, Portugal
INESC, R. Alves Redol, 9 P-1000 Lisboa Portugal
Tel: + 351 1 3100301
Fax: + 351 1 3145843
{Nelson.Antunes,rmr,Paulo.Pinto}@inesc.pt

KEYWORDS
ATM; Traffic Management; Call Admission Control; Measurement based algorithms.

ABSTRACT
This paper presents a heuristic approach to the problem of call admission control. The algorithm is based on traffic measurements and does not guarantee bounded delays or cell losses. It features very low cell delay variation tolerance for CBR traffic and is aware of negotiated, but unused, bandwidth from users. The algorithm does not assume any traffic models for the calls, so composed traffic models for call admission control cannot be used either. Such algorithms have to be tested by simulation. We simulated several specific controversial cases using real pattern traffic (CBR, real MPEG films, and typical IP traffic). Different offered loads (per simulation and during the same simulations) were used. The algorithm proved to deliver the same results as the theoretical approaches (in respect to average cell delay) but with much higher network utilization. It proved to manage well very "ill-behaved" calls in terms of burstiness and grain size.

1. INTRODUCTION

The asynchronous characteristics of ATM networks, allowing connections to vary their bit rate over time, but providing also guarantees for transferring constant bit rate, turn the traffic management control into a challenging problem. Traffic management control has been approached from various angles and a common feature of the proposals is the guarantee of the quality of service (QoS), in terms of delay or cell loss probability. Such approaches give rise to algorithms which are very conservative because they are tuned to the worst case (which seldom happen). We propose a heuristic algorithm, fully based on measurements with the concern of both reducing the cell-by-cell computation to a minimum and setting a decision level for call admission very near to the real traffic. Traffic models for the calls are not required (apart from recognizing the distinction between CBR – Constant Bit Rate, and VBR – Variable Bit Rate) but simply the conformance to the token bucket algorithm (De Prycker 1995), which is hardly a limitation. The higher utilization of the network is achieved by relinquishing strong requirements of the bounded delay for the cells, although the average delay is inherently controlled by the measurements and additional methods. One of the novel features in this approach is the awareness of the negotiated but unused bandwidth from users, that a real network should somehow take into consideration in order to reserve some of it for possible future use. As there are no formal guarantees of quality of service (end-to-end delay) the algorithm cannot be mathematically proved to be correct. Simulation is, then, the unique tool to test and validate the main features of the algorithm. Simulations were performed using real traffic (CBR calls, MPEG videos and data traffic using an ON-OFF model with typical IP length distribution). Stress cases were tested with larger parameters for the data traffic type in order to produce "ill-behaved" traffic which is not handled efficiently by theoretical based algorithms. All simulations were performed with very high levels for the traffic load.

Traditional schemes to the traffic control are based on: (a) deterministic approaches (Knightly and Zhang 1995; Knightly *et al*. 1995), where all cells of a connection are guaranteed to meet the promised QoS, possibly involving some trade-offs between delay and peak cell rate; (b) probabilistic approaches (Abe and Soumiya 1994; Beshai, Kositpaiboon and Yan 1994; Guérin, Ahmadi and Naghshineh 1991; Hui 1988; Murase *et al*. 1991), where methods make use of the statistical multiplexing gain achieving a better network utilization; (c) long range dependence, heavy-tailed distribution and self-similarity (Georganas 1994; Likhanov, Tsybakov and Georganas 1995), where structural similarities across a very wide range of time scales are exploited; and (d) measurement based algorithms to feed theoretical models (Jamin *et al*. 1995; Saito and Shiomoto 1991; Murase *et al*. 1991). Although it is hard to directly compare simulation results, our simulated average delays are compatible with the results reported in the best of these studies, and we achieved a higher switch utilization with a simpler algorithm for equivalent conditions.

2. QUEUE SERVICE DISCIPLINE

A design objective for the queue service discipline was to reduce the cell delay variation as low as possible for CBR traffic. All the other types of traffic were grouped taking into account their sensitiveness to CDV. Medium CDV is appropriate for real-time like traffic and unconstrained CDV is suitable for data traffic. A switch with these characteristics can handle equally well telephone like voice connections, encoded video and bursty data. To achieve this design objective three queues were created:

1. Low CDV queue – used only for CBR, and featuring a nearly-constant average delay;
2. Medium CDV queue – used for VBR traffic wanting a nearly-constant average delay and a medium CDV. Calls in this class have certain restrictions to their burstiness (rate between peak and average) and grain size (rate between peak and link bandwidth).
3. Unconstrained CDV queue – used for traffic that has no strong restrictions on delay or CDV.

The three queues are pictured in figure 1. The lighter part corresponds to the working area and the darker area exists to hold bursts when they happen. It is assumed that cells are never lost due to lack of memory. This is feasible because the admission algorithm works on link capacity and it is assumed that there will be plenty of memory to drive the link near its full capacity. When the link becomes full, delay will start to rise and calls in queues 1 and 2 stop to be accepted, providing a good control of the buffer. For queue 3, delay can be more permissive because there are other flow mechanism to be used (ATM FORUM 1995).

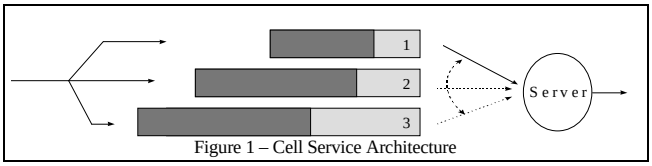

Figure 1 – Cell Service Architecture

The cell service discipline is the following: queue 1 is served whenever there are cells waiting in the queue; queue 2 is served whenever there are cells waiting in the queue, and queue 1 is not being served; queue 3 is served on all other occasions. An extra small feature was tested on the simulations. Cells in queue 1 were deliberately delayed to set their average delay to a similar value as that of queue 2. Applications using ATM networks will have to live with drift. Therefore, it could be advantageous to have similar values for delay in these two queues because it can help the inter-stream synchronization for multimedia applications when the audio and the video have separate circuits (Correia and Pinto 1995) (i.e., smaller synchronization buffers). The introduction of this delay feature on queue 1 involves a minor change to the service algorithm. Cells in queue 1 are only served after a resting time, unless the number of cells in the queue is such that the service time of the last one is already greater than the artificial delay.

With this kind of service discipline a certain load on queue 1 takes service time from the other queues and the same happens with queue 2 towards queue 3. A major problem in our service discipline happens if CBR traffic is continuously being accepted, stealing time from the other queues. We achieve a balance between the different type of traffic classes by setting traffic quotas to configure the behaviour of the switch (e.g., making it more CBR oriented). The cell service based on a weighted fair queuing service (Demers, Keshav and Shenker 1989) introduces a variable delay on CBR cells because it spends some time serving non real time traffic queues, and the number of queues is unknown.

3. TRAFFIC MANAGEMENT

Traffic Management has two types of control: preventive control, which is basically the call admission control; and reactive control, which tries to solve problems when calls were already accepted but traffic behaves differently than was expected (e.g., less statistical multiplexing gain).

3.1 Call Admission Control

Calls are accepted if the set of conditions hold. The conditions are defined in terms of the following parameters: *load measurements*; *delay measurements*; *unmeasured conditions* – concepts that cannot be measured directly – such as, assessing the ratio of utilization of the network towards the negotiated contracts; the characteristics of the call (burstiness and grain size); and statistical multiplexing gain.

The measurements are based on a scheme showed in figure 2.

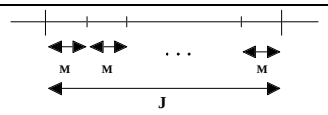

Figure 2 – Measurement scheme

There are periods of measurements of M seconds and a larger period corresponding to *n* M intervals, called J. Measurements are used to maintain estimations for peak, $\hat{P}$, and average, $\hat{A}$, loads, as well as maximum delays, $\hat{D}$. The same procedure is used for the three queues: at the end of each M the queue is measured for load and delay. The peak load measured, P, is the ratio between the number of arrived cells and M. The average load measured, A, is the average of P over J. Finally, the maximum delay measured is the longer duration a cell experienced to be served during M. The expression for P and A are:

$$P = \frac{Number\_of\_arrived\_cells\_during\_M}{M}$$

$$A = \frac{1}{n}\sum P \quad , \quad n = \frac{J}{M}$$

The three estimators maintained by the measurements are updated in the following conditions:

$\hat{P}$	$\hat{A}$	$\hat{D}$
• within J if the measured value is greater than the current value • at the end of J with the greatest value measured, regardless of its current value • when a new call is accepted for this queue (see 3.1.5 below)	• at the end of J regardless of the current value • when a new call is accepted for this queue (see 3.1.5 below)	• within J if the measured value is greater than the current value • at the end of J with the greatest value measured, regardless of its current value

With this kind of algorithm, the estimators freeze the worst situation in a certain period J and reasoning is based on these values when new calls want to be accepted in the following period J. It is assumed that policing mechanisms prohibit streams to use more bandwidth than the negotiated. This is particularly important as the algorithm tends to adapt itself to the traffic that passes. In (Saito and Shiomoto 1991) it is assumed, on the other hand, that calls can pass the negotiated contract (i.e., there is no policing) and his measurement based algorithm still adapts to the situation.

Although our algorithm works on the measured values, the estimators are not updated at the end of J if a call is accepted during this interval based on those values (See 3.1.5 below for the updates when a call is accepted). When a call finishes the estimators are not updated either. The influence of the call will cease to exist and will be measured in future intervals. An important feature of the algorithm is the choice of values for M and J. If M is small, the estimators are more aware of bursts and traffic correlation – the difference between $\hat{P}$ and $\hat{A}$ gets bigger leading to a lower utilization of the link. If M is large, the estimators see smoother traffic and $\hat{P}$ gets near to $\hat{A}$. Regarding J, if it is small, $\hat{P}$ adapts very quickly to the real load and can create instability to the algorithm. If J is large, an interval can last long enough to include long range dependency of certain type of traffic (specially those generated by the MPEG algorithm). If J gets even larger the algorithm is sensitive to increases on traffic, but adapts slowly to lighter loads (only after J). The delay estimator is not very sensitive to the values of M and J. Larger J just makes delay estimators to live a little bit longer.

There are two problems associated with measurement based algorithms which were not properly covered in the literature. First, a temporary lower utilization of the network by some sources in regard to what they had negotiated can produce an adaptation of the algorithm if these periods are long enough (the algorithm assumes that the statistical multiplexing gain is greater than it is). When the sources restart to use the network at the negotiated levels, as they are entitled to, problems with delay and buffers can arise. The adaptation can hardly be considered a feature, as in (Jamin *et al*. 1995), because the network is providing a service and should not infer other parameters from the user´s traffic pattern. The second problem happens when the call has very strange parameters – very high burstiness and/or very high grain size. If the OFF periods are long enough (greater than J) a similar adaptation can happen and the algorithm just forgets that a peak will soon arrive. The third set of call admission conditions are targeted to these problems.

3.1.1 Implementation of the Algorithm. The switch is configured by six parameters. The first, called α, is the maximum link utilization, or load. The load cannot be 100% because the delay could increase too much. Our objective is to reach loads near 100% but the admission control algorithm must be set to a lower value. Traffic in queue 3 will fill the remaining bandwidth if certain conditions are met (see 3.1.3 below). The real value of α is dependent on the traffic characteristics. If the traffic is not very bursty and its grain size is small then a large value can be chosen. If the traffic is very bursty and its grain size is high a lower value is preferable. It is important to note that expressions such as "very bursty" "small grain size" and the value for α can only be meaningful after simulation and field test of the switch. We have chosen a value of 90% for α and the third set of conditions is responsible to avoid the admission of bursty traffic if the switch is already very loaded (or avoid the admission of "normal" traffic if bursty calls were accepted and could need bandwidth).

The next three parameters are the quotas for each queue, β_1, β_2 and β_3. These parameters are important to prevent traffic from one queue to excessively use the switch. When traffic from queue 1 is accepted, the other queues can suffer longer delays. These quotas can be exceeded on certain circumstances (e.g., when the switch has a light load. See 3.1.2 below). This mechanism of quotas is important because it preserves bandwidth to be used by other types of traffic. If such a mechanism did not exist the switch would invariably reject calls for other queues if one queue monopolized the switch making it unsuitable for a large range of traffic types. An opposite example happens in (Jamin *et al*. 1995) were high priority calls are systematically rejected after a certain value of load.

The last three parameters are not independent. They are the maximum delay for the queues. d_1 is obtained from β_1, d_2 is derived from β_1 and β_2, and d_3 is set to a desired value for maximum delay. There are no actual expressions for derivation but only hints from simulations. The values for d_i will actually set the lighter zones for the buffers in figure 1. The meaning of d_3 is slightly different from the others because there are not strong commitments in this queue. d_3 is an indication of the filling level of queue 3.

The negotiated parameters needed to establish a call were adopted from the ATM Forum standard (ATM FORUM 1995). For CBR traffic the PCR (Peak Cell Rate) is the only parameter needed. For the other two classes the terminal must provide the PCR, the MBS (Maximum Burst Size) and the SCR (Sustainable Cell Rate). It is assumed, without loss of generality, that the streams conform to the Generic Cell Rate Algorithm (GCRA) (De Prycker 1995). Assuming this algorithm the maximum number of cells in M, N(M), is

$$Maximum\_number\_of\_cells\_in\_M = N(M) \le \min\left(\left\lfloor 1 + \frac{M + \tau_s}{T_s} \right\rfloor, \left\lfloor 1 + \frac{M}{T} \right\rfloor\right)$$

where τ_s is the burst tolerance; T_s is the average inter-arrival time between two cells and T is the minimum inter-arrival time between two consecutive cells. The first expression is used when M is greater than, or equal to *MBS*T*, otherwise the second is used.

The following paragraphs describe how the estimators, configuration parameters and negotiated parameters are used in the expressions for the call admission control.

3.1.2 Load Measurements. The first set of conditions is related to the load measurements. For the first queue (CBR traffic) the following expressions are used

$$\hat{P}_1 + \hat{P}_2 + \hat{P}_3 + \frac{N(M)}{M} < \alpha \ \wedge \begin{cases} \hat{P}_1 + \dfrac{N(M)}{M} < \beta_1 \\ \vee \\ \hat{P}_1 + \dfrac{N(M)}{M} > \beta_1 \ \wedge \ \hat{D}_2 < 0.75 d_2 \ \wedge \ \hat{D}_3 < 0.75 d_3 \end{cases}$$

The equation on the left checks if the load relative to the maximum number of cells in M can still fit in the overall load of the switch (*N(M)* is divided by *M* because the unit of the expression is load). If it can, then the quota of the queue, β_1, is checked in a similar way. If both are valid, the load condition is passed with success. If β_1 is exceeded then the other queues are checked for available resources (75% of the maximum delay was used as a threshold value during simulation). Although traffic in queue 1 is not disturbed by the other queues, the algorithm does not allow the quota to be exceeded unless the traffic in the other queues has still some spare "space" for their own new calls. The expressions for queues 2 and 3 are similar.

3.1.3 Delay Measurements. The rational behind the delay analysis is to check if the maximum delay was experienced during the interval. This algorithm does not impose a guaranteed value for maximum delay and allows it to fluctuate in order to maximize the utilization of the link. It is important to note that the average delay will be much smaller (see section 4). The equations for the delay are the following:

First queue: $\hat{D}_1 < d_1 \wedge \hat{D}_2 < d_2$ *Second queue*: $\hat{D}_2 < d_2$ *Third queue*: $\hat{D}_3 < d_3$

Traffic on queue 1 is only accepted if the situation in its queue and queue 2 is normal. Traffic for queue 2 is accepted if situation in its queue is normal. It is useless to test delay in queue 1 because traffic in queue 2 will never disturb queue 1. Traffic for queue 3 tests only its queue, as well. Queue 3 is never tested when calls for queues 1 or 2 appear because delay in queue 3 is mainly used for reactive control. Its traffic is also used to fill the remaining bandwidth and the maximum delay value exists to prevent buffers from filling up dangerously.

3.1.4 Unmeasured Conditions. High burstiness and/or high grain size can produce traffic with large fluctuations on the number of cells emitted. The peaks can be so far apart that an interval J can fit entirely between them leading to incorrect measurements. Sub-utilization of the negotiated conditions can also lead to wrong measures. The problem with burstiness and grain size is that it is more damaging to the buffers and to the delay to have just a few number of calls with high values for burstiness and grain size than a larger number of more "well behaved" calls**.**

The algorithm becomes aware that the measurements can be wrong if the $\hat{A}$ is reasonably lower than the negotiated average over a period of J (either because of the burstiness or the sub-utilization). This scheme has the advantage of taking into account the potential problem of bursts in regard to the measurement interval (J) used for the entire algorithm. I.e., it is not an absolute measure of burstiness. When $\hat{A}$ is reasonably lower than the entire negotiated sustainable cell rate, a new condition for the *potential real traffic* of the calls has to be used (assuming a certain statistical multiplexing gain). This value must be somewhere between the values from the expressions above and the peak values of the calls. Once again, the interval M is used to provide a dependency to the measured unit and N(M) gives an indication of the worse case in terms of burstiness and grain size. We considered a difference of 10% on the load of any queue (measured versus average) to become aware of a problem in the measurements, and the top value of α, for the negotiated SCR.

$$\Sigma \frac{SCR_i}{Link} - \hat{A}_i > 0{,}1 \qquad \Sigma \frac{SCR}{Link} < \alpha$$

When this happens, $\hat{P}_i$, in the load expression above is replaced by $\tilde{P}_i$, the sum of the load of each call

$$\tilde{P}_i = \Sigma\, Potential\_real\_traffic_i$$

As this expression is only valid if the measures fail, its scope of utilization is not so important for the algorithm, i.e., in normal circumstances the real measures are an indication of the load on the switch. During these special cases it is better to fail by being conservative than the other way around. The potential real traffic of each call is given by

$$\begin{cases} \dfrac{N(M)}{M} & if \quad M > MBS * T \\ \left[\dfrac{1}{T_s} + \left(\dfrac{N(M)}{M} - \dfrac{1}{T_s}\right) * \dfrac{MBS * T}{MBS * T + T_s + \tau_s}\right] & if \quad M \le MBS * T \end{cases}$$

The idea behind the expression is to derive an equivalent ON-OFF stream from the negotiated parameters and assume that during the ON periods the call is transmitting at peak rate followed by a sufficiently long OFF period to conform to the average. The upper expression is used when the duration of the ONs is smaller than M (there is at least one burst followed by some portion of an OFF period inside M). If the duration of the ONs is bigger than M then N(M) represents the PCR of the connection, and the estimation could be very conservative (specially if the call is bursty). So, in this case, we try to measure how far apart the bursts are. We add the difference between the peak load and the mean load weighted by the frequency of the burst, to the mean load of the connection.

3.1.5 Acceptance of a Call. When the parameters of a new call make the various expressions valid, then the call is accepted and the estimators of the peak and average load are updated using the worst cases (It is assumed that the new call starts with a peak):

$$\hat{P}_i' = \hat{P}_i + \frac{N(M)}{M} \qquad \hat{A}_i' = \hat{A}_i + \frac{SCR}{Link}$$

The following J intervals will measure the real effects of the new call.

3.2 Reactive Control

Reactive control is used when calls were accepted but the measures indicate a violation of the configuration parameters (basically the delays). It can have various reasons. For queue 1, a long delay means that there is a strange correlation among the cells. For queue 2, maximum delay can be large due to bursts or correlation of bursts. The limitation of the burtiness of the calls for this queue, stated above, has to do with this mechanism. If queue 2 rejects very "ill-behaved" calls then d_2 is a good indication of the load. Otherwise, d_2 would start triggering problems just because of bursts. For queue 3 the ATM Forum has standardized the rate flow mechanism of ABR (Available Bit Rate) (ATM FORUM 1995). It is then possible to slow down some calls in order to keep the delay within the configuration bounds (remember that the estimator is the maximum delay and the average delay is significantly lower). The configuration parameter d_3 could be set at such a level to let the delay increase in order to use the link in a more store and forward way. No reactive control was used in the simulations.

4. PERFORMANCE ANALYSIS

In order to prove the suitability of our approach and also to study the impact of certain conditions on the performance of our algorithm, a set of simulations was performed. For that, we developed an ATM network model which consists of a given number of terminal multiplexers connected to a switch as depicted in Figure 3.

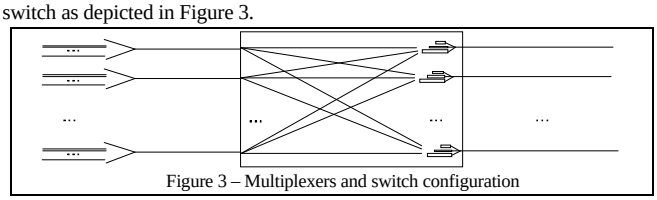

Figure 3 – Multiplexers and switch configuration

In this model, we have considered an 8x8 non-blocking switch with output buffering. The incoming and outgoing links have a capacity of 155 Mbit/s. Each inlet is fed with a stream of cells coming from a multiplexer. With this configuration it is possible to accommodate a large number of connections into a single inlet, enforcing the token bucket filter and allowing a very flexible loading pattern.

To represent the traffic of the different type of calls, models for CBR and ON-OFF were used and real MPEG films were used for VBR. The CBR used a deterministic model and the ON-OFF was composed by three ON states and one OFF state, with geometric distributions and batch arrivals (Bonomi *et al*. 1994; Kontovassilis, Tsiligaridis and Stassinopoulos 1995).

Before each simulation starts the time of arrivals and departs for each type of calls is generated based on an exponential distribution. The choice of the mean time between arrivals and the mean duration of calls is calculated by the mean load weight of a call and the load we want to obtain (called ρ). The basic set of characteristics for each type of call is: CBR (64 Kbps, 424 Kbps, 1.6 Mbps); MPEG (21 films), ON-OFF (404 Kbps, 809 Kbps, 2 Mbps). This set was used in all simulations but the one in section 4.2.3 where extra "ill-behaved" ON-OFF sources were added. All ON-OFF calls have burstiness of 2. During the simulation, when a call is due to begin, two decisions are made randomly: its characteristics and the multiplexer it will use. The call is then targeted to outlet 1, to simplify the simulations.

The purpose of the CBR calls was to represent plain telephone calls, video conferences and CBR films; real VBR MPEG films were available and one of a set of 21 is chosen randomly; different parameters were chosen for data traffic calls (with some extra in section 4.2.3), as stated above.

All simulation runs have a sufficient long duration to yield statistical significant results enhanced by the realization of a warm up period during which no observations are collected. In the next sections we present 2 different types of simulation results. The first one relates to the performance of the queue service discipline whereas the second one addresses the behaviour of the Call Admission Control.

4.1 Queue Service

The main goal of this set of simulations is to show the behaviour of the queue service discipline when the enforced delay mechanism for queue 1 cells is active. This happens when a significant part of the load is conveyed through the queue 2 and tries to replicate a situation where typically multimedia connections are involved. To achieve this, the system was tested under high loads and experiencing highly unbalanced conditions regarding the traffic distribution through the 3 queues of the outlet 1 (Figure 3).

These simulations were ran for 4 different load values: ρ=0.6, ρ=0.7, ρ=0.8 and ρ=0.9 with a load distribution of : ρ_1=25%, ρ_2=65% and ρ_3=10%. During these simulations, we tagged one of the several multimedia connections (a CBR and an MPEG film) and observed the delay performance of its components for a period of 100 seconds. Figure 4 shows the delay experienced by audio and video components in queues 1 and 2 respectively, for the different load values.

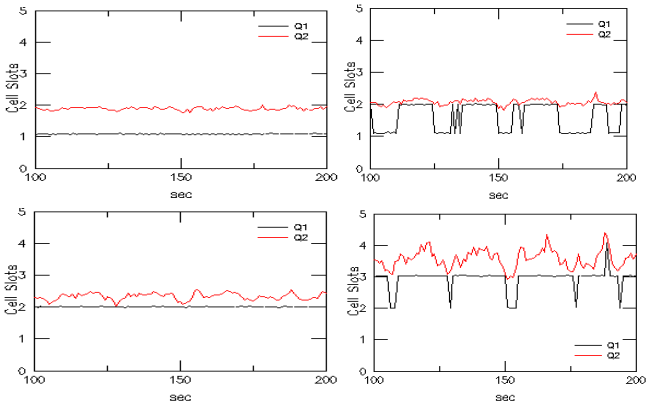

Figure 4 - Cell service discipline

It is clear from the figure that two types of behaviour are observed for the delay pattern of queue 1: for ρ=0.6 and ρ=0.8 the pattern apparently shows no influence from the cell delay in queue 2, apart from the induced delay, whereas for ρ=0.7 and ρ=0.9 the pattern displays several spikes due to the influence of queue 2. The explanation for this kind of behaviour is the following: during each measurement window J, the average delay (for all calls) of queue 2 is measured. This delay, in cell slots, is computed as an integer number. For ρ=0.6 the measured delay is stable throughout the observation period shown (equal to 1 cell slot). The same situation occurs for ρ=0.8 but now with a 2 cell slot delay (therefore, the average delay for queue 1 is 2 cell slots). However, as the delay in queue 2 approaches an integer number boundary, the enforced delay mechanism becomes unstable. This happened for ρ=0.7 and ρ=0.9 where the delay in queue 1 varies rapidly each time the delay in queue 2 crosses the integer threshold. Note however that the magnitude of such variation does not exceed 1 cell slot.

Figure 5 shows the histogram of the cell delay in queue 1 and 2 for loads ranging from 0.6 to 0.9. They are representative of how the delays are distributed and how much CDV can be expected. It is clearly observed that queue 2 exhibits the major delay variations whereas queue 1 only shows a slight spreading of its cell delays. However, for ρ=0.7and ρ=0.9 the variation of cell delay increases and the delay dispersion around the average value is quite noticeable. Nevertheless, if we compute the 95th percentile for all the histograms, the maximum dispersion is about 4 cells for queue 1 and about 7 cells for queue 2. These values represent a small variation specially when

compared with the corresponding median which is of 3 cells for both cases. This means that a possible value for the peak delay is not more than 4 cells slots away from the central value of the delay distribution in the case of queue 2 and not more than 1 cell slot in the case of queue 1.

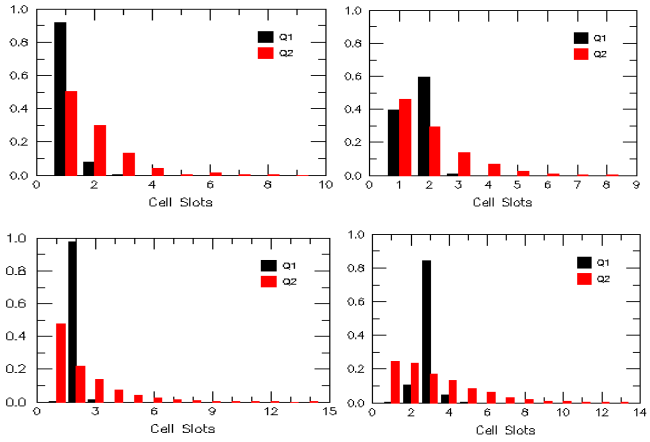

Figure 5 - Distribution of the delays in queue 1 and 2 for different offered loads.

4.2 Traffic Management

The simulations described below correspond to five different situations a switch can be confronted to. The first situation highlights the response of the algorithm to sudden variations of the load offered to the switch with special emphasis on the algorithm tracking performance. The second situation reflects the case when the sources negotiate more bandwidth than they will use during the lifetime of the connection (only the sources for queues 2 and 3 are over-negotiated). The third situation covers the case of sources with high grain-size and burstiness, which can produce congestion in the network. The fourth simulation results show the behaviour of the algorithm when the offered load represents a higher value than the system can carry. Finally, the last one depicts the response of the Call Admission Control mechanism to unbalanced traffic conditions and particularly the impact on the call rejection rate and delay performance.

4.2.1 Load Variation Tracking. The simulation model was driven with a load of ρ=0.7 evenly distributed through all the queues. In order to verify the tracking performance of the algorithm, ON-OFF periods characterize the overall call generation profile. Each ON period with a duration of 10 minutes is followed by an OFF period with the same duration. During ON periods, calls with an average duration of 5 minutes were randomly generated whereas no call generation takes place during the OFF periods. The α was 0.9, all β_i were 0.3, d_1 was 10, d_2=20 and d_3=50.

An overall time of 50 minutes was simulated. Figure 6 shows two details of the simulation results where the average load estimator $\hat{A}$ and the peak load estimator $\hat{P}$ are represented.

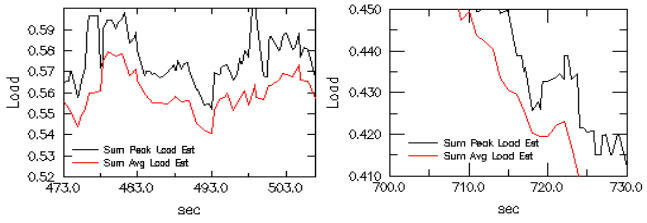

Figure 6

As Figure 6 depicts, the variations on the offered load are smoothly tracked by the average load estimator. The degree of tracking is, of course, dependent of window size J. Additionally, the two estimators react in response to either increases or decreases on the load offered to the switch. When the load increases, the peak load estimator immediately follows this variation while it is accompanied by the average load estimator in a more smooth way. However, a load decreasing is handled more conservatively which is denoted by the slower updating of the load estimators. This is due to the fact that, after each new accepted call, the value of $\hat{A}$ and $\hat{P}$ are immediately updated with the negotiated parameters ($\hat{P}$ is updated as if the call starts with the peak traffic). When no call arrives but the traffic has a burst, $\hat{P}$ hangs to the maximum value. For the descending part, both in the case of a call tear-down (there are no updates) or in the case of a decrease on the load, $\hat{P}$ gets smaller only if the traffic features no spikes during an J.

4.2.2 Unused Negotiated Bandwidth. In this simulation the model was driven with: ρ=1.02, ρ_1=0.29, ρ_2 =0.3, ρ_3=0.43, α=0.9, $\beta_1=\alpha/3$, $\beta_2=\alpha/3$, $\beta_3=\alpha/3$, and d_1=10, d_2=20, d_3=50. The warm up period considered was 500 seconds. The overall negotiated offered traffic is higher than the switch capacity, and the connections for queue 2 and 3 will always negotiate a value superior to what they will actually send. Figure 7(a) shows the distribution of calls per queue. Each call had 5 seconds for the mean between arrivals and 300 seconds for the mean duration. This gives a mean of 60 calls per queue.

Figure 7(b) represents the difference between the average load estimator $\hat{A}$ (lowest line), the sum of the SCR (middle line), and the equivalent load (potential real traffic) for queue 3, during the whole simulation. It can be seen the more conservative value of the potential real traffic when the measures are not trustworthy. Queue 2 has a similar pattern to queue 3 and is not showed. For queue 1, all curves converge in one, since the sources are deterministic and the negotiated parameters represent the load really sent.

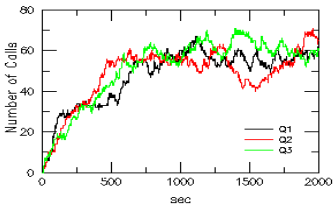

Figure 7 (a)

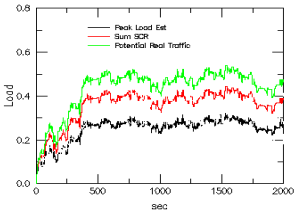

Figure 7 (b)

Figure 8(a) shows the total load for all queues with the total SCR and the total peak load estimator, $\hat{P}$, plotted. It is seen that the sum of all SCR (the negotiated bandwidth) never exceeded α, which is a condition of our algorithm. Finally, figure 8(b) shows the actual number of calls accepted (figure 7(a) referred to the number of calls offered). A comparison between the two figures shows that there were some rejected calls for queue 2 and 3, even for load levels below the quota. The reason for that was the fact that the measures have failed, and the more conservative value for the used bandwidth was considered. The expression led to a value greater than α. This procedure recalls the negotiated parameters and tries to protect the current users.

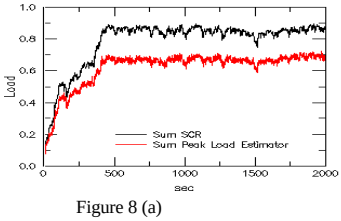

Figure 8 (a)

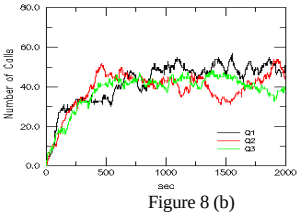

Figure 8 (b)

4.2.3 High Bursty and Grain Size Sources. To simulate high bursty calls the model was driven with: ρ=0.9, ρ_1=0.3, ρ_2 =0.3, ρ_3=0.3, α=0.9, β_1=0.3, β_2=0.3, β_3=0.3, and d_1=10, d_2=30, d_3=100. Notice that the values d_i were changed from the ones in the previous simulations (see Conclusions). The warm up period considered was 500 seconds. The ON-OFF calls were augmented with two extra calls: one with 5 Mbits/s of mean traffic and the other with 10 Mbits/s. The burstiness of all these calls remained equal to 2. We have also permitted ON-OFF connections in queue 2, with a probability of 25%. The offered load, ρ, was equal to the switch capacity.

Figure 9 (a) exhibits the overall peak load estimator and average estimator, as well as the individual peak load estimators for each queue. The difference between the peak and average load estimators is very narrow and the lines are almost coincident. For the individual queue lines it is visible the effect of the bursty calls on queues 2 and 3 (lighter lines in the figure).

Figures 9 (b) and (c) indicate the variation of the maximum delay estimator for each queue. The results have shown that queue 3 can suffer long delays, when "ill-behaved" connections are considered, even when the conditions of the call admission control are satisfied. The complete algorithm would require an action from the reactive control in order to reduce the delays. Even without it, maximum delays in queue 3 are not dramatic. Cells in queue 2 do not suffer so long delays because its service priority is higher but the 25% of the ON-OFF traffic is noticeable. In terms of violation of the maximum delay, queue 1 was always below. Queue 2 had some loaded periods where the delay never pass 30. Queue 3 had some periods with maximum delay over 100. During the stress periods the average delay for queue 2 was 2 cell slots.

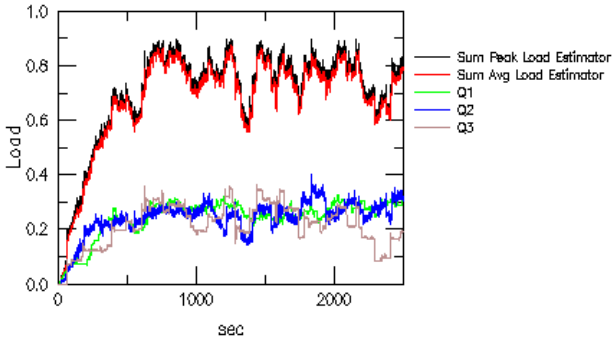

Figure 9 (a)

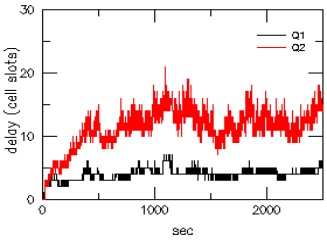

Figure 9 (b)

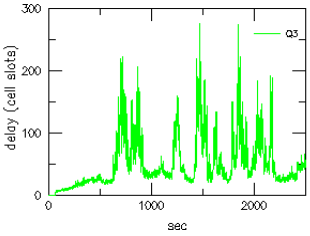

Figure 9 (c)

Figures 10 (a), (b) and (c) show the number of calls offered and rejected per type of calls (the third figure has all the data calls even if some went to queue 2). Rejections occurred during the more stressed periods. Queues 1 and 2 suffered the most. The main reason for rejections was the traffic type quota because the conditions on delay were not met. Rejections for queue 3 were due to the delay on the queue and not its quota (if the ABR control scheme was active more calls would have been accepted because the existent ones would decrease their bit rate). A minor number was rejected because the switch capacity, α, was exceeded. If the values of d_i were not changed from the values of the previous simulations, more calls would have been rejected.

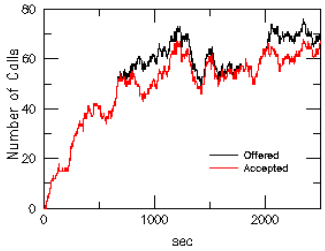

Figure 10 (a)

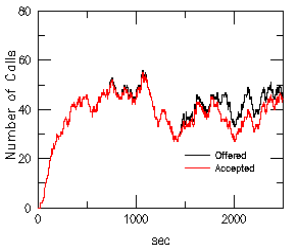

Figure 10 (b)

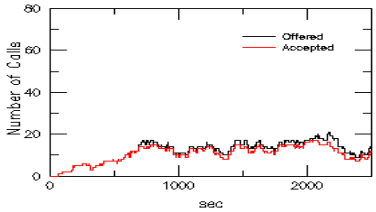

Figure 10 (c)

4.2.4 Heavy Load Conditions. In the case of heavy load, the same configuration parameters for the switch as those in section 4.2.3 have been used. The offered load had different ON periods as depicted in figure 11(a). The ON periods toggled between 1 and 0.7 and the load distribution per queues was the same. Figure 11(a) plots the total load transported (peak load estimator) and the individual peak load per queue. The lines are much smoother than in the previous case as the calls are more "well-behaved". The aggregated value remains just below α when the ON period has load 1 due to rejections. Figure 11 (b) and (c) show the maximum delay for the three queues. It is interesting to see that the values for queue 2 are higher than in the previous simulations and the values for queue 3 are lower. The higher values for queue 2 were due to higher offered traffic and some calls were accepted over its load quota. Another reason is the high load in queue 1 (it reaches 0.4 at near 600 seconds) inducing higher delays in the system. The lower maximum delays for queue 3 were a consequence of the characteristics of the calls (more "well-behaved"). The violations for queue 3 were much fewer than for the previous case. This result is interesting because under stronger conditions of load the switch reacts better and the call admission phase is sufficient to drive the switch smoothly. In the previous simulation, the "ill-behaved" calls passed the call admission control and cause some problems afterwards.

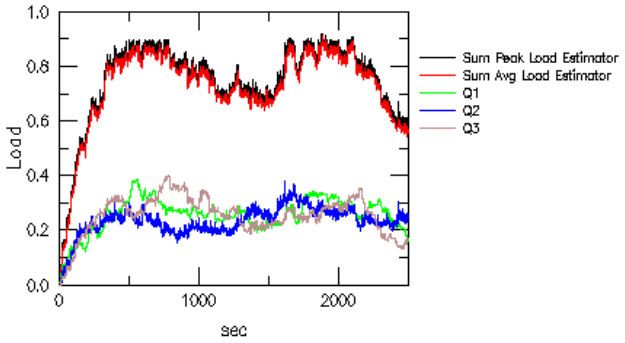

Figure 11 (a)

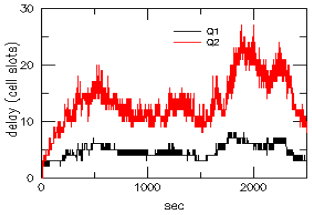

Figure 11 (b)

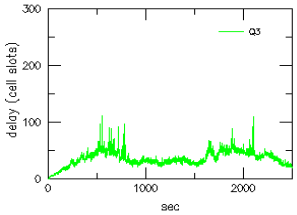

Figure 11 (c)

Rejections happened for all queues and were caused mainly by an excess of the overall quota (during the stress periods). As we used the same values for the maximum delay as in the previous simulation, there were no rejections due to the delay. However, this fact could only be possible due to the nicely balanced offered traffic. If it was not so, the queue with the most traffic load would overpass its load quota without problems because the delay conditions in the algorithm would always validate the call. If the situation persisted different delays would be felt for the lower priority queues and the switch could be working under the optimal load conditions.

4.2.5 Unbalanced Traffic Conditions. Finally, the last simulation is, in fact, composed of two different traffic profiles both with an overall load value

of ρ=0.8. The first traffic profile uses the following load distribution: $\rho_1=\rho/4$; $\rho_2=\rho/2$; $\rho_3=\rho/4$; this profile is similar to the one described in section 4.1. In the second traffic profile we swapped the load distribution between queue 1 and 2, to verify if a different load characteristic (CBR contributes now significantly to the total load) could modify the delay performance and the load accepted.

In addition, the maximum delay values and the load quotas were chosen so that similar conditions could be achieved in both cases. The quotas chosen were derived from the simulation of section 4.1 and have the following values:

- profile 1 $d_1=8$, $d_2=25$, $d_3=70$ $\beta_1=0.225$, $\beta_2=0.45$, $\beta_3=0.225$
- profile 2 $d_1=12$, $d_2=35$, $d_3=70$ $\beta_1=0.45$, $\beta_2=0.225$, $\beta_3=0.225$

The simulation results, represented as the maximum delay experienced by the cells in each queue, in each measurement interval, are shown in Figures 12 (a) and 12 (b), for profiles 1 and 2, respectively. The load graphics are similar to what one would expect and are not shown.

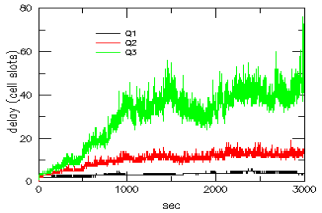

Figure 12 (a)

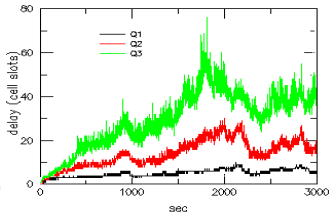

Figure 12 (b)

The first conclusion that can be drawn from the figures is that the selection of appropriate delay values for each profile is of paramount importance as wrong parameters could lead to significant call rejection due to delay quota exceeded situations. If for example, we have chosen, for profile 2, a value for d_2 equal to the one used in profile 1, a non-negligible call rejection rate would have been found. As the delay performance of queue 2 is not only dependent on the fraction of load handled by this queue but is also related with the load offered to queue 1, one can conclude that the choice of d_2 is directly related with ρ_1 and ρ_2 and, hence, with β_1 and β_2.

The second conclusion has to do with the behaviour of the queue 2 delay in the figures shown – queue 2 experience more maximum delay in profile 2 and the maximum delay for queue 3 is less stable. Since the average total load for queue 1 and 2 is the same (75% of the total load), one would expect to observe lower delays for the queue 2 for profile 2 as the traffic in queue 2 is lower. The first answer is that the delays in the graphics are the maximum delays. When the traffic in queue 1 is greater, cells in queue 2 see cells in queue 1 very often. Traffic in queue 2 is eminently bursty and it will be served at a lower pace in profile 2. When more than one burst arrive at the same time, the delay increases much more than in profile 1. In the latter case it is likely that the low utilization of queue 1 will improve the service of queue 2 draining the bursts quicker. In terms of accepted calls and load, both profiles (with the configuration parameters as they were) performed equally.

5. CONCLUSIONS

The main conclusion of this paper is the achievement of a higher utilization of the network due to a relaxation on the guarantees of QoS. Simulations have shown that the quality of service obtained is high so the relaxation simply moved away the excessive conservative nature of the theoretical approaches.

A measurement based method with some conservative adjustments can be the ideal solution for the call admission control in ATM networks, because it is not dependent on the connection traffic characteristics (making it future-safe for the new services, such as ATM intends to be); it is very simple with a straightforward implementation; and can provide answers to new calls within a very short time. Measurement based algorithms work better when there are multiple calls. The requirement of high network utilization fits entirely the characteristics, so the approach is tuned with the problem. When the utilization is low, some errors might occur but it is not critical as the resources are available.

One of the main characteristics of the algorithm is its awareness of unused bandwidth or the existence of high bursts. This approach seems inevitable to a commercial network and does not reserve too much bandwidth. The cell service discipline seems highly adjusted to the traffic classes defined, as the simulations proved. There is a natural balance between the existent classes and the objective of a very low CDV for the CBR traffic was achieved.

The configuration parameters are essential for a good utilization of the network. To achieve the best utilization possible the load quotas must be tuned to the percentage of the traffic types the switch is going to handle. Concerning the maximum delays, the main factors to take into consideration are the burstiness of the traffic (specially for d_3) and the balance of the traffic types. If CBR traffic is going to be the major type of traffic, d_2 and d_3 should have slightly greater values.

In terms of further work there are some interesting areas to explore. We did not simulate the algorithm exhaustively. Important issues are the consideration of an entire network and see how bursts in one switch propagate through the network.

A thorough study of the ABR reactive control on queue 3, including its latency over the network is also an important issue. The distinction of service classes between ABR and UBR could also lead to a clearer definition of the reactive control queue 3.

6. REFERENCES

Abe, S. and T. Soumiya. 1994. "*A Traffic Control Method for Service Quality Assurance in an ATM Network.*" IEEE Journal on Selected Areas in Communications, Vol. 12, No. 2, February, pages 322-331.

The ATM FORUM Technical Committee. 1996. "*Traffic Management Specification.*" Version 4.0, April.

Beshai, M.; R. Kositpaiboon; and J. Yan. 1994. "*Interaction of Call Blocking and Cell Loss in an ATM Network.*" IEEE Journal on Selected Areas in Communications, Vol. 12, No. 6, August, pages 1051-1057.

Bonomi, F.; J. Meyer; S. Montagna; and R. Paglino. 1994. "*Minimal ON/OFF Source Models for ATM Traffic.*" ITC´94, pages 387-399.

Correia, M. and P. Pinto. 1995. "*Low-Level Multimedia Synchronization Algorithms on Broadband Networks.*" ACM Multimedia´95, San Francisco, pages 423-434.

Demers, A.; S. Keshav; and S. Shenker. 1989. "*Analysis and Simulation of a Fair Queueing Algorithm.*" Proc. of ACM SIGCOMM'89, pages 3-26, September.

Georganas, N. 1994. "*Self-Similar ("Fractal") Traffic in ATM Networks.*" Proc. 2nd Internacional Workshop on Advanced Teleservices and High-Speed Communication Architectures (IWACA´94), Heidelberg, Germany, September, pages 1-7.

Guérin, R.; H. Ahmadi; and M. Naghshineh. 1991. "*Equivalent Capacity and Its Application to Bandwidth Allocation in High-Speed Networks.*" IEEE Journal on Selected Areas in Communications, Vol. 9, No. 7, September, pages 968-981.

Hui, J. 1988. "*Resource Allocation for Broadband Networks.*" IEEE Journal on Selected Areas in Communications, Vol. 6, No. 9, December, pages 1598-1608.

Jamin, S.; P. Danzig; S. Shenker; and L. Zhang. 1995. "*A Measurement-based Admission Control Algorithm for Integrated Services Packet Networks.*" SIGCOMM '95.

Knightly, E. and H. Zhang. 1995. "*Traffic Characterization and Switch Utilization using a Deterministic Bounding Interval Dependent Traffic Model.*" IEEE INFOCOM'95.

Knightly, E.; D. Wrege; J. Liebeherr; and H. Zhang. 1995. "*Fundamental Limits and Tradeoffs of Providing Deterministic Guarantees to VBR Video Traffic.*" Proceedings of ACM SIGMETRICS '95.

Kontovassilis, K.; J. Tsiligaridis; and G. Stassinopoulos. 1995. "*Buffer dimensioning for delay - and loss-sensitive traffic.*" Computer Communications, Vol. 18, No 5, May, pages 315-328.

Likhanov, N.; B. Tsybakov; and N. Georganas. 1995. "*Analysis of an ATM Buffer with Self-Similar ("Fractal") Input Traffic.*" Proc. IEEE INFOCOM´95, Boston, Abril.

Murase, T.; H. Suzuki; S. Sato; and T. Takeuchi. 1991. "*A Call Admission Control Scheme for ATM Networks Using a Simple Quality Estimate.*" IEEE Journal on Selected Areas in Communications, Vol. 9, December, pages 1461-1470.

De Prycker, M. 1995. *Asynchronous Transfer Mode: Solution for Broadband ISDN.* 3rd Ed. Prentice Hall.

Saito, H. and K. Shiomoto. 1991. "*Dynamic Call Admission Control in ATM Networks.*" IEEE Journal on Selected Areas in Communications, Vol. 9, No. 7. September.