14

Four Standard Control Theory Approaches for the Implementation of RRM ABR Services

M.Ajmone Marsan, A.Bianco, R.Lo Cigno, M.Munafò
Dipartimento di Elettronica, Politecnico di Torino
Corso Duca degli Abruzzi, 24 - 10129 Torino - Italy
{ajmone,bianco,locigno,munafo}@polito.it

Abstract

Four different algorithms are considered for the implementation of RRM ABR services within an ATM switch. The algorithms are inspired by standard techniques in control theory, and are designed aiming at simple implementation in low-cost ATM switches. The four algorithms rely on the periodic measurement of the buffer occupancy, which triggers the congestion detection. ABR feedback to sources can be based on buffer occupancy, on its derivative and integral. The performances of the four algorithms are first compared in a simple scenario, in which 5 greedy TCP connections share a bottleneck link with non-ABR background traffic, in order to obtain a first assessment of their effectiveness. Then a more complex topology, generally known in the literature as 'parking lot', is considered. Finally, we concentrate on the best of the four algorithms to study its ability to efficiently control network performances in different scenarios. The study is based on simulation, using the software tool named CLASS.

Keywords

ATM, ABR, Traffic Management, Best Effort Traffic, TCP over ABR

1 INTRODUCTION

Although ATM networks were originally designed for services requiring a statistically guaranteed quality of service (QoS), like for example videotelephony or real-time video, best-effort services, like multimedia electronic mail, or LAN interconnection, are expected to play a key role in the first wave of ATM. The

*This work was supported in part by a research contract between CSELT and Politecnico di Torino, in part by the EC through the Copernicus Project 1463 ATMIN, and in part by the Italian Ministry for University and Research and the Italian National Research Council.

main difference between the two types of services lies in the impossibility for the latter to provide a detailed description of their traffic characteristics, since their cell generation pattern may depend on unpredictable events.

Fortunately, however, it often happens that cell sources that are not able to properly describe their traffic characteristics are able to adapt their offered traffic to the network load. Because of this situation, ATM standards define such service categories as ABR (Available Bit Rate), UBR (Unspecified Bit Rate), and ABT (ATM Block Transfer) [1, 2], that allow the establishment of connections for which neither the traffic characteristics nor the QoS are completely specified or guaranteed. The failure or success of ABR, UBR, and ABT will largely depend on their ability to provide inexpensive and reliable services. The identification of effective algorithms for the support of these service categories is thus extremely important.

The ABR service category was explicitly designed for sources that can adapt their cell transmission rates to some feedback signal issued by the network. ABR provides two different operating modes: a simpler one called RRM (Relative Rate Marking) where the network feedback can only assume three values, and a more complex one called ERM (Explicit Rate Marking) where the network explicitly notifies sources about their assigned cell transmission rates. In both schemes, the key issue that determines the performance of the network lies in the control algorithm implemented within the ATM switches to decide what feedback must be returned to sources.

In this paper we concentrate on the ABR RRM scheme, and we investigate the influence of different approaches for congestion control within the switch. The study is performed through simulation using CLASS [3, 4]. The ABR connections that we consider carry the traffic generated by sources performing long file transfers using the TCP protocol.

The performance of TCP connections supporting file transfers over ATM networks was already studied by several authors. Results indicated that when the UBR service category is used, performance is generally rather poor [5], mainly due to the TCP protocol dynamics, that cannot be easily adapted to networks with high bandwidth-delay products [6]. When an RRM ABR scheme with a very simple control mechanism is used in the ATM network to support TCP connections, simulation results [7] showed that satisfactory performance can be achieved, provided that the ABR parameters are carefully tuned. In this paper we extend the work in [7] to explore different control algorithms for the generation of the ABR feedback signal within the ATM switches. All algorithms are based on a periodic measure of the number of occupied positions in the link buffer.

The remainder of the paper is organized as follows. Section 2 describes the considered RRM control algorithms, explaining their rationale and their relations to standard control theory schemes. Section 3 describes the network scenario that was considered in the simulation study, as well as some of the key features of our simulation environment. Section 4 illustrates and comments the

numerical results obtained from the simulation runs. Finally, Section 5 ends the paper presenting our preliminary conclusions and indicating the guidelines for the prosecution of this work.

For the sake of brevity, we assume that the reader is familiar with the basic ABR mechanisms as described in [2] and with the TCP transport protocol (see for instance [6, 8]); details about the implementation of TCP in CLASS can be found in [4, 5, 7].

Further information about the current developments of CLASS can be found on the Web at the URL `http://hp0tlc.polito.it/class.html`.

2 THE RATIONALE OF THE CONTROL SCHEMES

In the RRM ABR scheme, feedback to sources is conveyed by the network through the NI (No Increase) and CI (Congestion Indication) bits of the RM (Resource Management) cells. The feedback indication can only assume three values corresponding to "increase the cell transmission rate" (NI=CI=0), "keep the present cell transmission rate" (NI=1, CI=0), or "decrease the cell transmission rate" (CI=1, NI not significant). Sources react to these feedback indications by modifying their cell transmission rates according to their values of the ABR parameters RIF (Rate Increase Factor) and RDF (Rate Decrease Factor), that are negotiated at connection setup.

The NI and CI bits can be set to 1 by ATM switches along the connection path, depending on the switch internal congestion. What type of information is to be used to determine the switch congestion, and what algorithm is to be adopted to set the bits is an implementation choice, and thus it is not and will not be defined by standards. We focus on the algorithms that set the NI and CI bits, and we assume that they are based on the occupancy of the link buffer, which is shared by the ABR connections and the background traffic.

Since the core algorithm on which the RRM ABR scheme is based is very simple, it can be expected that RRM schemes will be implemented in low-cost ATM switches; as a consequence, also the control algorithms implemented within the ATM switches must be very simple. These remarks about the type of ATM switches that we consider lead to the assumption that no sophisticated queueing and traffic control mechanisms are available within the switches; in more detail, we assume that the switches have no per-VC or per-traffic-class separate buffering, and they are not able to compute the instantaneous load offered by the connections. The only available information on link congestion is hence the buffer occupancy Q_l.

Under the above assumptions, the problem of the definition of the feedback to be returned to sources can be formulated in control theory terms: the system to be controlled can be described as a multi-input single-output system, whose state is described by a single variable Q_l, as depicted in Figure 1, where x_i is the traffic offered by ABR connection i, y is the throughput of the output link, d_{bg} the traffic offered by the non-ABR background traffic, that is assumed to

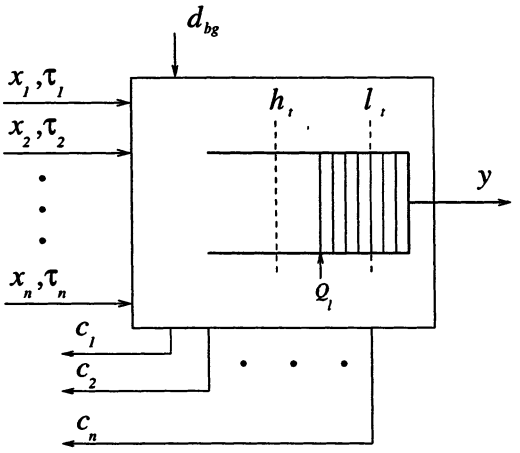

Figure 1 Model of an ATM switch supporting n ABR connections on one output link

be non controllable and thus is, from the control viewpoint, an additive noise. The control feedback returned to ABR source i is c_i, while τ_i is the delay elapsing between the control action within the node and the instant when the rate modification of source i reaches the node. This delay must not be mistaken for the connection round trip time rtt_i, and depends not only on i but also on network congestion, and thus is time-dependent. h_t and l_t are two thresholds on Q_l used by the control algorithms. As noted before, the control signal can only assume the three values IR (increase rate), KR (keep rate), or DR (decrease rate); it must however be remarked that the response of each source to the control signal depends on the parameters RIF and RDF negotiated at connection setup and, if the feedback is DR, also on the actual value of x_i.

The goal of the control of the system in Figure 1 is threefold: i) avoid losses in the buffer, ii) maximize y, i.e., the link utilization, and iii) minimize the buffer occupancy fluctuations to avoid introducing unwanted jitter in the cell delay (remember that the buffer is shared by ABR and background traffics, the latter possibly being delay sensitive).

Given the discrete nature of the feedback signals, the system depicted in Figure 1 is intrinsically non-linear, and the problem of its control cannot be solved analytically, unless drastic simplifications are introduced (see for instance [9, 10]). In addition to the intrinsic nonlinearity of the problem, it should be noted that the ABR control algorithm must operate in a network environment, where a large number of nodes are interconnected and share the same means (i.e., the RM cells) to convey feedback to sources; it is thus possible that feedback signals, issued by one node, are modified by other congested nodes, that set either the NI bit, or the CI bit, or both. This second

aspect makes an analysis even more difficult, since the same control channel is shared by different nodes.

The control of a system like the one in Figure 1 is not addressed in control theory textbooks, but some similarity can be found with the problem of controlling the level of a water reservoir. The main difference between the two problems lies in the linearity of the water reservoir system description, but note also that in the case of the reservoir the delays τ_i are time-independent. From the solution of the water reservoir problem, we know that the water level can be made more stable and the control more robust if the control algorithm makes use not only of the information concerning the water level, but also of its derivative and integral values. Starting from these considerations, and making use of some heuristics, we derive different control algorithms that we call P control, PD control, PD+ control, and PID control, 'P' standing for position (or level) of the buffer occupancy, 'D' for its derivative and 'I' for a measure of its integral. The buffer occupancy is measured at regular epochs, S_r slots apart; between two consecutive samplings the control signals c_i are kept constant. All control algorithms follow the rule that forbids resetting the values of NI and CI bits in RM cells.

P control — This is the simplest control techniques that can be implemented with two thresholds l_t and h_t on the buffer occupancy Q_l. The control feedback is set as follows independently from i:

$$\begin{array}{rcl} Q_l < l_t & \Longrightarrow & c_i = \text{IR} \\ l_t < Q_l < h_t & \Longrightarrow & c_i = \text{KR} \\ h_t < Q_l & \Longrightarrow & c_i = \text{DR} \end{array} \qquad (1)$$

The only parameters that allow the control to be tuned, apart from the buffer sampling interval S_r, are the two thresholds l_t and h_t. It is an easy prediction that Q_l will oscillate, and that the amplitude and period of the oscillations will be proportional to the delays τ_i.

PD control — Let $D_Q(n) = Q(n) - Q(n-1)$ be the two-point derivative of the buffer occupancy Q_l, where n is the n-th sampling instant. The sign of D_Q (dropping n for the sake simplicity) can be used to forecast future buffer congestion or link underutilization. Since the algorithm goal is to obtain the maximum possible buffer stability, the natural mapping of the values of Q_l and D_Q over the control feedback is the following:

$$\begin{array}{rcl} Q_l < l_t \;;\; D_Q \leq 0 & \Longrightarrow & c_i = \text{IR} \\ Q_l < l_t \;;\; D_Q > 0 & \Longrightarrow & c_i = \text{KR} \\ l_t < Q_l < h_t \;;\; D_Q \leq 0 & \Longrightarrow & c_i = \text{KR} \\ l_t < Q_l < h_t \;;\; D_Q > 0 & \Longrightarrow & c_i = \text{KR} \\ h_t < Q_l \;;\; D_Q < 0 & \Longrightarrow & c_i = \text{KR} \\ h_t < Q_l \;;\; D_Q \geq 0 & \Longrightarrow & c_i = \text{DR} \end{array} \qquad (2)$$

Also in this case the control is independent from i and the only parameters available to tune the control are the two thresholds l_t and h_t, but the use of the additional information concerning the buffer filling trend should result in a more stable buffer occupancy.

PD+ control — This technique is derived from the previous one, giving more importance to buffer occupancy variations: if $|D_Q(n)| > \beta$ then the value of Q_l is ignored. The goal of these modifications is to detect important traffic variations as soon as possible and, as a consequence, to keep the buffer as empty as possible to minimize cell losses. The mapping of the values of Q_l and D_Q over the control feedback is the following:

$$
\begin{array}{rcccccccccccc}
 & & \forall Q_l & & & ; & & & D_Q & < & -\beta & \Longrightarrow & c_i = \mathrm{IR} \\
 & & \forall Q_l & & & ; & \beta & < & D_Q & & & \Longrightarrow & c_i = \mathrm{DR} \\
 & & Q_l & < & l_t & ; & -\beta & \leq & D_Q & \leq & 0 & \Longrightarrow & c_i = \mathrm{IR} \\
 & & Q_l & < & l_t & ; & 0 & < & D_Q & \leq & \beta & \Longrightarrow & c_i = \mathrm{KR} \\
l_t & \leq & Q_l & < & h_t & ; & -\beta & \leq & D_Q & \leq & \beta & \Longrightarrow & c_i = \mathrm{KR} \\
h_t & \leq & Q_l & & & ; & -\beta & \leq & D_Q & < & 0 & \Longrightarrow & c_i = \mathrm{KR} \\
h_t & \leq & Q_l & & & ; & 0 & \leq & D_Q & \leq & \beta & \Longrightarrow & c_i = \mathrm{DR}
\end{array}
\qquad (3)
$$

The control is as usual independent from i, but in this case also the parameter β, together with the two thresholds l_t and h_t, can be used to tune the control algorithm.

PID control — This last technique is based also on the system history; we are interested in the "recent" past, hence we should compute the integral of the buffer occupancy over a moving or jumping window. The same information, however, can be obtained in a more convenient way by computing the following recursive equation

$$I_Q(n) = \alpha Q(n) + (1-\alpha) I_Q(n-1)$$

which, if $0 < \alpha < 1$ defines a single-pole digital IIR low-pass filter whose impulse response is exponential with decay parameter $\theta = -(1-\alpha) \cdot S_r \cdot T_s$, where T_s is the duration of one slot.

The parameters available to tune the control now comprise, besides the two thresholds l_t and h_t, also the value α that defines the memory of the filter. I_Q and Q_l can be compared against different sets of thresholds, but, since I_Q is bounded by Q_l, we will use just two thresholds for the sake of simplicity. Recalling that the goal of the control algorithm is first of all to avoid losses, it can be argued that when congestion is building up, i.e., $D_Q > 0$, the control algorithm should be based on the information conveyed by the value of Q_l, which is faster to react to traffic changes; when congestion is relaxing ($D_Q < 0$), instead the algorithm can be based upon the value of I_Q, which is slower to react to traffic changes, thus helping

in smoothing the oscillations of Q_I. Based on these heuristics, the following mapping of c_i is proposed:

$$\begin{array}{lllllllll}
D_Q \leq -\epsilon & ; & & I_Q & < & l_t & \Longrightarrow & c_i & = & \text{IR} \\
D_Q \leq -\epsilon & ; & l_t \leq & I_Q & < & h_t & \Longrightarrow & c_i & = & \text{KR} \\
D_Q \leq -\epsilon & ; & h_t \leq & I_Q & & & \Longrightarrow & c_i & = & \text{DR} \\
D_Q > -\epsilon & ; & & Q_I & < & l_t & \Longrightarrow & c_i & = & \text{IR} \\
D_Q > -\epsilon & ; & l_t \leq & Q_I & < & h_t & \Longrightarrow & c_i & = & \text{KR} \\
D_Q > -\epsilon & ; & h_t \leq & Q_I & & & \Longrightarrow & c_i & = & \text{DR}
\end{array} \tag{4}$$

where ϵ is a small positive quantity (typically 1 or 2 cells) that avoids considering cell-level bursts as a misleading indication of congestion build-up or relaxation. As a matter of fact, the mapping defined by (4) considers small variations in D_Q as a signal of "potential congestion buildup".

3 THE SCENARIO UNDER STUDY

For the analysis of the behavior of the four RRM control algorithms, we concentrate upon two network topologies. The first one, depicted in Figure 2 defines a very simple network scenario, comprising just two ATM switches connected by one link that is the system bottleneck. The choice of such a simple topology stems from the desire to isolate phenomena due to the control algorithms from those due for instance to topology. Moreover, this topology can be considered an approximate model of any network where only one link is congested between the source and the destination: apart from the buffer of the congested link, all other buffers along the connection can be assumed to be almost empty, so that delays are dominated by the propagation delay and can be considered as almost constant. These assumptions justify long delays between the source and the control point since congestion may arise in any switch, not only near the sources. The bit rate on all links equals 150 Mbit/s; n ABR connections, each one using a different link to reach the first switch, converge upon the only link between the two switches, creating a potentially highly congested situation. The link between the two switches and the associated buffer are shared also with some background non-ABR traffic. The second topology we study is depicted in Figure 3 and is generally known in the literature as 'parking lot'. We consider N nodes and $N-1$ TCP connections together with some non-ABR background traffic; the non-ABR traffic on each link is statistically independent from the one on the other links. This topology allows us to study how the control algorithms in different nodes interact with one another, and check if ABR connections crossing more nodes are penalized with respect to those crossing less nodes.

The ABR connections serve greedy unidirectional sources implementing an ftp file transfer that lasts for the whole simulation experiment. File transfers

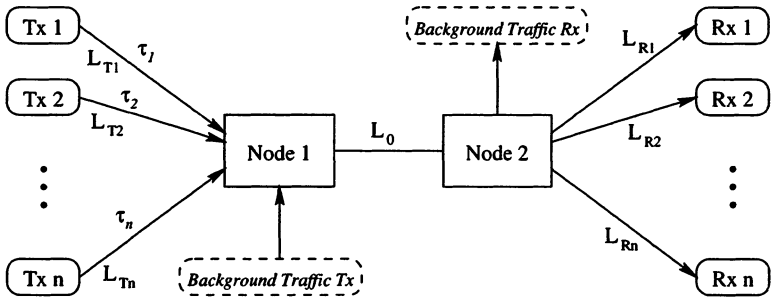

Figure 2 Bottleneck network topology

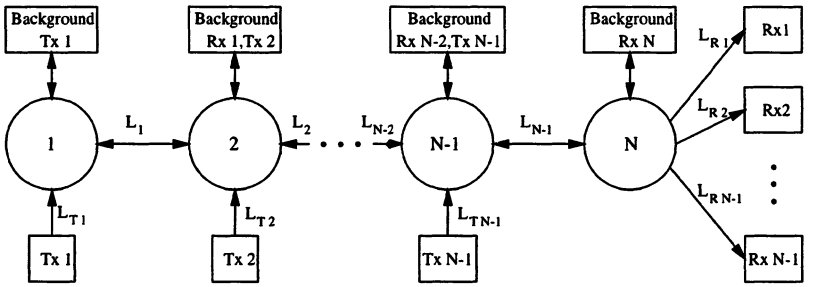

Figure 3 Parking lot network topology

exploit TCP connections that are provided by the officially distributed BSD 4.3 TCP-reno release [11], that was adapted to run above CLASS, as described in [4]. The TCP parameters are set so as to allow sources to grab all of the available transmission resources. The MSS (Maximum Segment Size) for TCP connections is set to 9140 bytes and all the TCP connections are assigned a maximum window size equal to 20 MSS if not otherwise stated.

The RRM ABR implementation in CLASS conforms to the ATM Forum Traffic Management Specification 4.0 [2]. It monitors the status of the buffer in the forward direction (i.e., on the link going from the transmitter to the receiver) and sets the feedback signal c_i on the RM cells flowing in the backward direction (i.e., on the RM cells going from the receiver to the transmitter). This operating mode minimizes the values of τ_i, thus making the control easier. The ABR parameters vector is kept constant in all the simulations: the most significant parameter values are summarized in Table 1. ICR is the rate at which sources start to transmit after a long silence period, PCR is the maximum allowed transmission rate, MCR is the minimum guarranteed transmission rate, RIF is the additive rate increase factor, RDF is the multiplicative rate decrease factor and TBE is the transient buffer exposure. For

Parameter	Value
ICR	10 Mbit/s
PCR	150 Mbit/s
MCR	0.1 Mbit/s
RIF	1/256
RDF	1/16
TBE	500 cells per connection

Table 1 ABR parameter values used in the simulation runs

what concerns TBE, we assume that its value is selected so as to accommodate all the cells transmitted by the source at ICR during a round trip time. We do not claim that the chosen parameter vector is the best possible choice for the considered situation, but the values are reasonable (we are not interested here in finding the best possible ABR parameters combination, rather in studying control algorithms for ABR) and they are among those recommended for use in [2]; moreover they have been proved to perform quite well in [12].

When present (and when not otherwise stated), the background traffic is generated by one ON-OFF source with average duty-cycle 2/3: on the average the source is transmitting for 2/3 of the time and silent for the other 1/3. The ON and OFF periods have exponentially distributed random durations, and the cell transmission rate during the ON periods is constant. Even if the burstiness of this background traffic source, defined as the ratio between the peak and the average cell transmission rates, is only 3/2, it is nevertheless very demanding for a control algorithm, probably much more demanding than the superposition of a number of more bursty sources.

In order to assess the effectiveness of the control algorithms we consider two sets of performance indices: i) the steady state link utilization and the background traffic delay jitter measured as the standard deviation of the cell delay, (these indices measure how well the network resources are exploited, and how much the background traffic is affected by ABR traffic), and ii) the goodput and efficiency of the ABR TCP connections, measuring the performance of ABR services from the user point of view. Moreover some examples of the dynamic behavior of the buffer occupancy are shown to gain better insight in the control schemes behavior.

4 NUMERICAL RESULTS

In this section we present some of the numerical results obtained from the simulation of the four considered control algorithms. The discussion of the results is divided into five subsections; first the four control algorithms are compared on the bottleneck topology with a given set of simulation parameters

	Parameter					
Control	S_r	B	h_t	l_t	β	α
P	100	5,000	2,500	100	N.A.	N.A.
PD	100	5,000	2,500	500	N.A.	N.A.
PD+	100	5,000	2,500	500	10	N.A.
PID	100	5,000	2,500	100	N.A	0.001

Table 2 Parameter values used for the different control algorithms

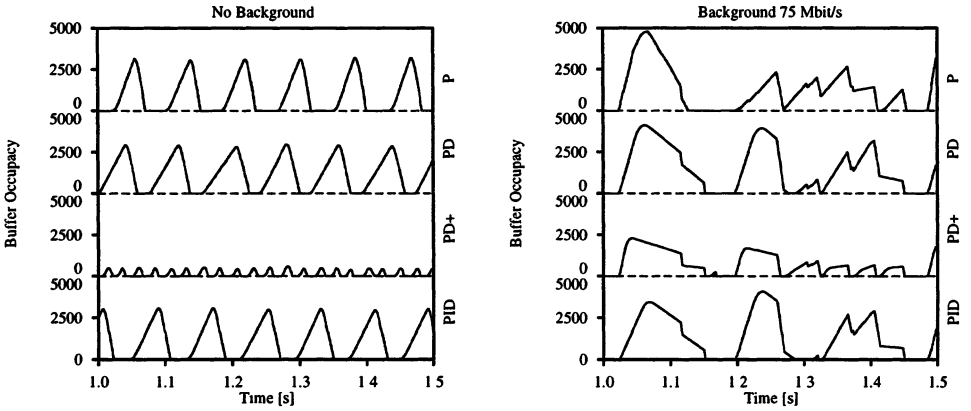

Figure 4 Time-dependent buffer occupancy, without background traffic (left), and with 75 Mbit/s background traffic (right), in the reference scenario with 5 TCP connections

which identifies the reference scenario. Then, a lower latency for the same topology is considered. Later, the parking lot topology is considered in order to compare the control algorithms in a more complex scenario. Finally, the performance of the most promising control technique is more deeply examined, looking at networks where connections span over different lengths, or where connections with different PCR are active.

4.1 The Reference Scenario

We take as a reference scenario a network covering the size of a European State, assuming a round trip time propagation delay of 20 ms, which roughly corresponds to a 2,000 km network span one way. With this scenario we have set $L_{Ti} = L_{Ri} = 500$ km and $L_0 = 1,000$ km (see Figure 2), which implies

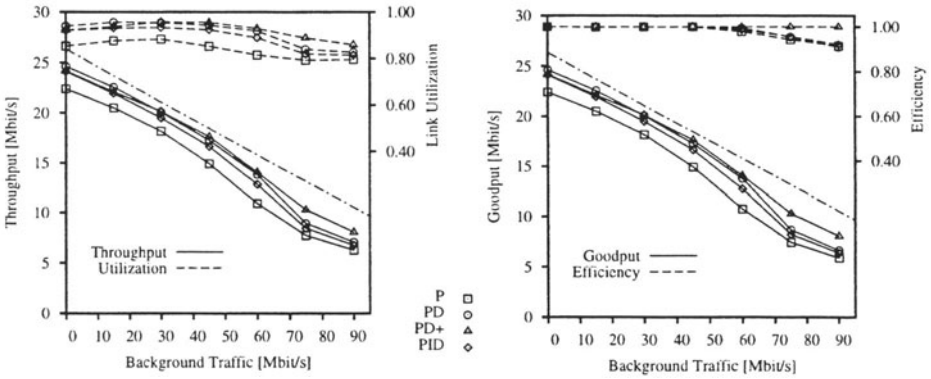

Figure 5 Throughput, link utilization (left), goodput and efficiency (right) averaged over the 5 TCP connections in the reference scenario

$\tau_i \simeq 5$ ms and $rtt_i \simeq 20$ ms. The number of TCP connections is set to 5. Table 2 summarizes the values of the main parameters used by the different control algorithms.

The buffer size should clearly be bigger than the sum of the TBEs of all the connections and we (arbitrarily) set it to twice that value: for the considered scenario we have a buffer $B = 5,000$ cells; the high threshold of the buffer is set to 2,500 cells with the idea that the TBE quota should be kept free under all possible circumstances.

From Table 2 it can be noticed that l_t is set to a higher value in the case of PD and PD+ controls: since the conditions to allow ABR connections to increase their rate are stricter for these controls, we have increased the value of the low thresholds so as to compensate for the potentially slower increasing rates of TCP connections.

All of the selected parameter values do have a large influence on the overall system performance, and each of them can be the subject of an optimization procedure; however, like in the case of the ABR parameter vector, we use fixed, reasonable values for all parameters, without any claim about the optimality of such values.

Figure 4 reports the measured buffer occupancy as a function of time, when no background traffic is present in the left-hand side picture, and when a 75 Mbit/s background traffic loads the bottleneck link in the right-hand side picture. Each picture contains the curves referring to the four considered control algorithms.

Even with such a long delay between sources and control point, which should make the task of the control algorithm hard, all of the considered algorithms attain quite similar performances. If no background traffic is present, the

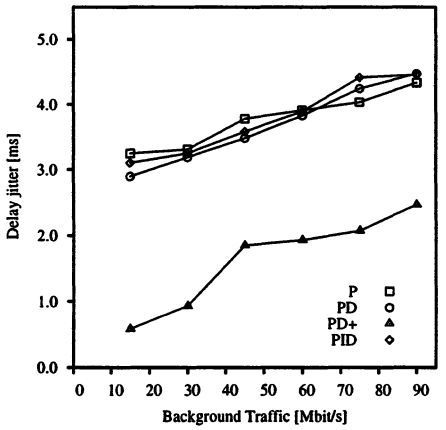

Figure 6 Delay jitter of the background traffic in the reference scenario for $\tau_i \simeq 5$ ms, $rtt_i \simeq 20$ ms, with 5 TCP connections

buffer occupancy oscillates rather slowly and smoothly (except for the case of PD+ control), which is an acceptable behavior since it seems almost impossible to achieve stability with just ternary-valued c_i's.

The PD+ control algorithm, which reacts to the quick variations of the buffer level, presents more frequent and smaller oscillations with respect to the other controls. This phenomenon is more evident in the left-hand side pictures, but it can be observed also when a 75 Mbit/s background traffic is present. Even if the PD+ control keeps the buffer occupancy at small values, it is able to provide a good network utilization as we shall see later; the ability of the PD+ algorithm to tightly control the buffer occupancy will become a significant advantage in most of the considered scenarios.

More complete indications on the performance of the four control schemes are given by the results reported in Figure 5. In the left-hand side picture, solid lines represent the average throughput obtained by the TCP connections, i.e., the overall number of TCP segments correctly delivered by the network divided by the simulation time. Retransmitted segments are considered as good, since we are interested in the overall network performance, and retransmissions due to higher level protocols cannot be ascribed to the network behavior. The dot-dashed line represent the "fair bandwidth share", which is the amount of capacity available for each TCP connection. The dashed lines report the overall link utilization, an index that shows how well network resources are being exploited.

The left-hand picture gives an idea of the raw performances of the control schemes, as viewed by the network: to have an indication of the performance

perceived by the users we have to consider the right-hand picture, in which the average "goodput" and efficiency of the TCP connections are plotted. The average "goodput" of the TCP connections is similar to the measured throughput but does not take in account the retransmitted segments. The efficiency is the ratio between the goodput and the offered load, and is an indication of the waste of resources introduced by retransmissions.

It can be noticed that the throughput and link utilization performance of all the schemes are quite similar; the P control behaves slightly worse, providing, on the average, only a 80% utilization of the bottleneck link. If we concentrate on the TCP performance indices, we first observe that also from the "user" perspective the P control behaves worse than the others. Moreover, the PD+ scheme can avoid TCP losses also with very high background traffic, as indicated by the efficiency close to 1.0 even with 90 Mbit/s background traffic load. In this high latency network we are close to, but we do not reach, the limit represented by the dot-dashed line, i.e. the fair bandwidth share that should be assigned to ABR controlled TCP connections; still, avoiding any losses in the TCP connections is quite a remarkable results.

We focus now on the delay jitter curves shown in Figure 6; this is the jitter of the delay suffered by the cells of the background traffic. The delay jitter is an indication of the ability of the algorithms to control the variability of the buffer occupation. As we should expect by the buffer occupation behaviour previously examined, the PD+ control algorithm provides significantly better performance with respect to all other algorithms; this is an important reason to prefer the PD+ control. Delay jitter is an important performance parameter by itself since the background traffic can have real-time characteristics, but it provides also an indication of the ability of the control mechanism to tightly control the network dynamics, which are prone to oscillating behaviours whose amplitude should be reduced to obtain an efficient control.

4.2 Lower latency network

We reduce here the network span with respect to the reference scenario, moving to an ATM network with the span of a Metropolitan Area Network. We assume a round trip time equal to 2 ms, that is one tenth of the one considered before. We have therefore set $L_{Ti} = L_{Ri} = 50$ km, $L_0 = 100$ km, values that result in $\tau_i \simeq 0.5$ ms and $rtt_i \simeq 2$ ms (see Figure 2). The other parameters remain the same as before, and we still consider 5 TCP connections.

The results obtained when the four considered control schemes are applied in this lower latency network are reported in Figure 7. The left-hand side picture reports the average goodput and efficiency of the TCP connections, while the right-hand side picture shows the delay jitter of the background traffic cells.

With the tighter control loop induced by the reduced round trip time, the

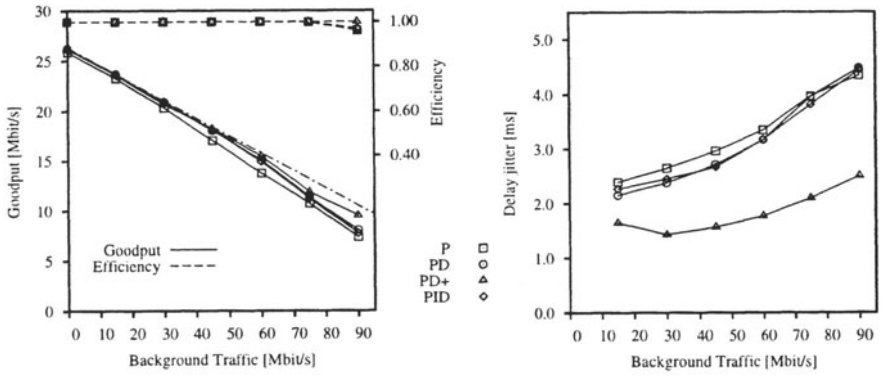

Figure 7 Average goodput and efficiency (left) and delay jitter (right) for $\tau_i \simeq 0.5$ ms, $rtt_i = 2$ ms, with 5 TCP connections

average performances obtained by the TCP connections are good for all four control schemes, but the PD+ scheme still performs slightly better than the others.

The tightness of the control loop does not significantly affect the delay jitter introduced on the background traffic with the P, PD and PID schemes, since this delay jitter is dominated by the buffer occupancy oscillations induced by the control algorithms; the measured jitters for these schemes are slightly smaller than the ones obtained in the reference scenario. For the PD+ scheme, instead, we can observe a somewhat higher jitter in this scenario for low background traffic, when compared to the reference scenario; in fact, in this low latency network, due to the tighter control on TCP sources, it is very unlikely that the node triggers the enhanced control based on sudden buffer occupancy increases. For this reason the buffer oscillates between the high and low thresholds instead of being tightly controlled to smaller values as in the previous scenario. Of course, higher buffer occupancy has a negative effect on the delay jitter. When the background traffic is increased this phenomenon quickly tends to disappear and we obtain results similar to those obtained in the reference scenario. This behavior is better explained by the time dependent buffer occupancy reported in Figure 8. With no background traffic, the buffer occupancy with the PD+ control oscillates just like in the PD case. With 75 Mbit/s background traffic load the qualitative behavior of the buffer occupancy is similar to the reference scenario.

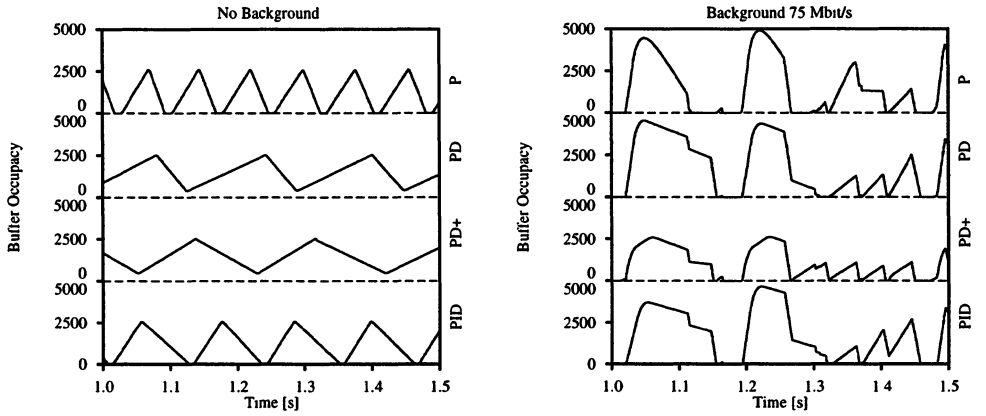

Figure 8 Time-dependent buffer occupancy, without background traffic, and with 75 Mbit/s background traffic, in the reference scenario with 5 TCP connections

4.3 Parking Lot Topology

We now consider a parking lot topology like the one depicted in Figure 3 with 7 nodes and 6 TCP connections. All connections have the same physical length (1000 km one way), but queueing delays are now different for each connection, depending on the number of crossed nodes. With reference to Figure 3, we have $L_i = 100$ km, $\forall i$, $L_{Ri} = 200$ km $\forall i$ and $L_{Ti} = 1000 - L_i - L_{i+1} - \cdots - L_{n-1}$ km. The round trip propagation delays are thus $rtt_i \simeq 10$ ms for all connections, when all buffers are empty. The τ_i differ from node to node; however, if we consider only node number 6, which controls the link that should be the bottleneck of the system, then the distance between the control point and the sources is 700 km and, when the network is lightly loaded $\tau_i \simeq 7$ ms.

We are now interested in the behavior of each single connection, since it can be expected that crossing a different number of nodes may introduce differences in performance due to the interaction of different, non coordinated control points.

Figure 9 reports the goodput and efficiency curves referring to the 6 TCP connections with the four considered control algorithms; they are plotted on different charts, one for each control algorithm. The connections are labeled from 1 to 6: connection 1 crosses all nodes, while connection 6 crosses only the last two, between which the most congested link is located.

The most remarkable feature that stems from these results is that the performances obtained by different connections are practically the same provided that they cross more than one control point. Instead, the connection that only crosses the last congested link always obtains a different goodput, which is,

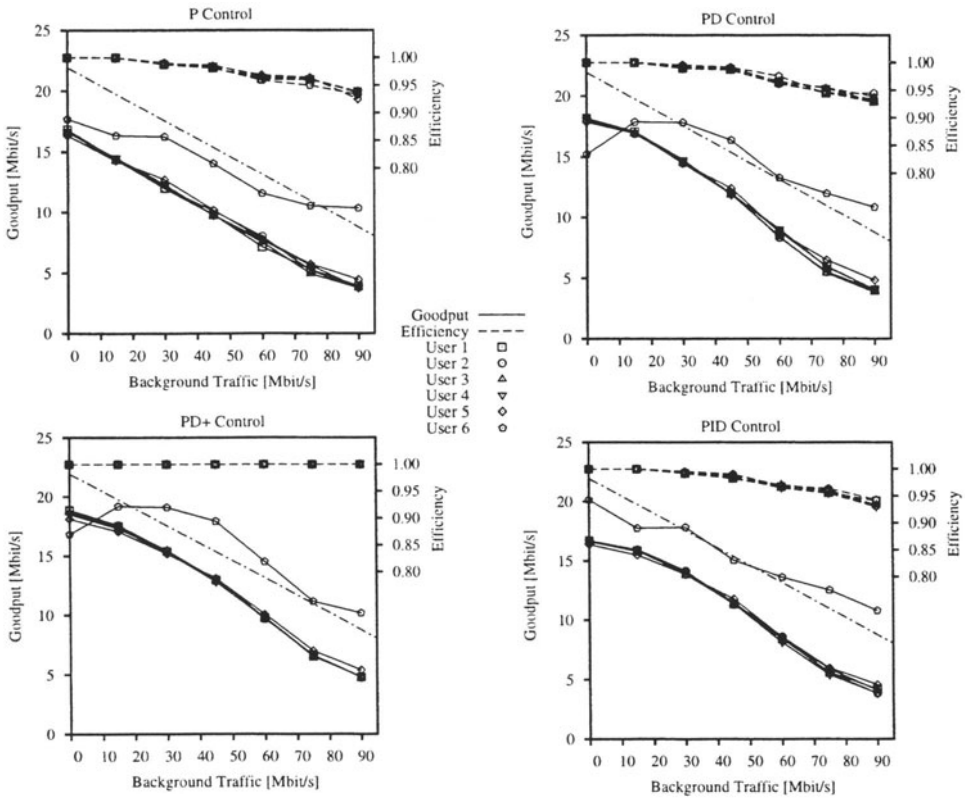

Figure 9 Goodput and efficiency of the each TCP connection in a parking lot topology with 7 nodes, $\tau_i \simeq 2.5$ ms, $rtt_i \simeq 10$ ms and 6 TCP connections

in most cases, significantly higher than thoee obtained by other connections. The only case when the goodput of connection 6 is smaller than those of the other connections, is when no background traffic is present and a derivative control is used. It is remarkable that the same behavior was observed in simulations of the parking lot topology with a different number of nodes and TCP connections (results for these topologies are not reported for the sake of brevity), indicating that the conclusions drawn here have a rather general validity. Also in this case we can observe that the PD+ control scheme outperforms the others, since it is the only one that guarantees no losses within nodes, regardless of the background traffic level. The other three algorithms, instead, show an efficiency degradation starting from quite light background traffic levels (20–30 Mbit/s), indicating buffer overflows within the nodes.

If we consider the delay jitter of the background traffic cells, shown in Figure 10, we observe a behavior similar to the bottleneck topology, since, once

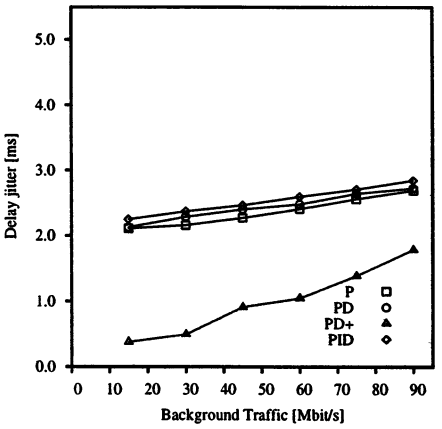

Figure 10 Delay jitter of the background traffic in the parking lot topology with $\tau_i \simeq 2.5$ ms, $rtt_i \simeq 10$ ms and 6 TCP connections

again, the PD+ control algorithm induces a smaller impact on the background traffic delay. It must be noticed that in this case the delay jitter of the background traffic cells is computed taking into account all the background traffic contributions, not only the one going from node 6 to node 7, and crossing the most congested link. Since the load on the other links is smaller, the delay of the other background traffic flows is less affected by the ABR traffic; this implies that the real difference in the background delay jitter is in fact greater than the one observed in Figure 10.

4.4 Variable Length Connections

We focus now on the case of variable length connections, considering only the PD+ control scheme since it has been shown to perform better and to be more reliable than the others. We turn our attention back to the simple bottleneck network with 5 TCP connections and ON-OFF background traffic. The distances of the five TCP sources from the bottleneck link are 2, 10, 50, 200, and 250 km respectively. We present results for this configuration in Figure 11; the span of the network between the first switch and the TCP receivers is 1500 km in the left-hand side picture, and 15 km in the right-hand side picture. Remember that the congestion control mechanism of TCP, being based on an adaptive window algorithm that reacts to the network status with a delay proportional to the round trip time, is prone to unfairness against connections that experience higher round trip times. Moreover, connections

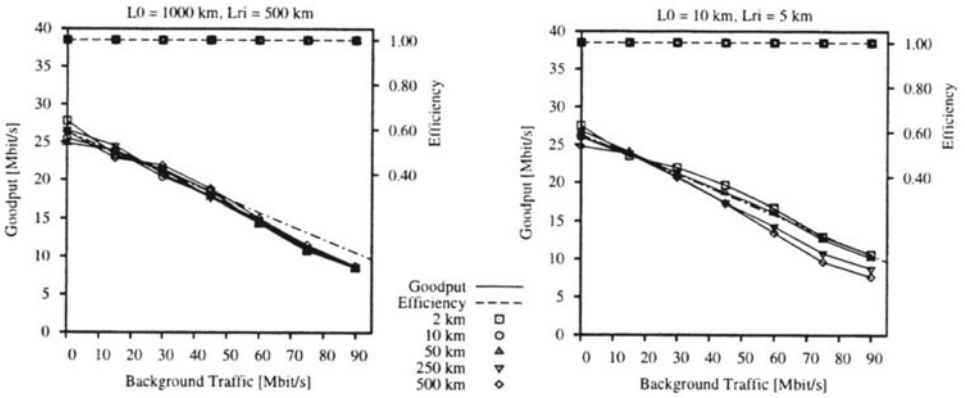

Figure 11 Average goodput and efficiency with PD+ control with 5 TCP connections of variable length

with different lengths experience different delays on the control signal received by nodes.

The overall goodput and efficiency figures are quite satisfactory in this scenario. A reasonable fairness is obtained among different connections when a network span of 1500 km is considered; the differences in the connections lengths are hidden by the network span so that the TCP biased behaviour is not striking. When a smaller network is examined, as in the right-hand side picture, unfairness arises among connections, although it must be noticed that the ABR mechanism allows a good control over the biased behaviour.

4.5 Connections with Different PCRs

In Figure 12 we present results for 5 TCP connections with different PCR in the bottleneck topology; two connections have PCR equal to 25 Mbit/s, two connections declare a 50 Mbit/s PCR, and the last connection is characterized by a PCR equal to 100 Mbit/s. In spite of the differences in PCR, we have used the same RIF and RDF parameters for all TCP connections. The left hand plot in Figure 12 refers to a round trip delay equal to 2 ms, while the right hand plot refers to a 20 ms delay.

In both the 2 ms and 20 ms scenarios, we can observe that connections obtain a goodput roughly proportional to their PCR, since, on average, all connections receive the same number of increase and decrease rate messages. Connections with higher PCR reach quickly a higher transmission rate with respect to other connections when increase messages are sent by nodes; when congestion is detected and nodes start issuing decrease signals, all connections

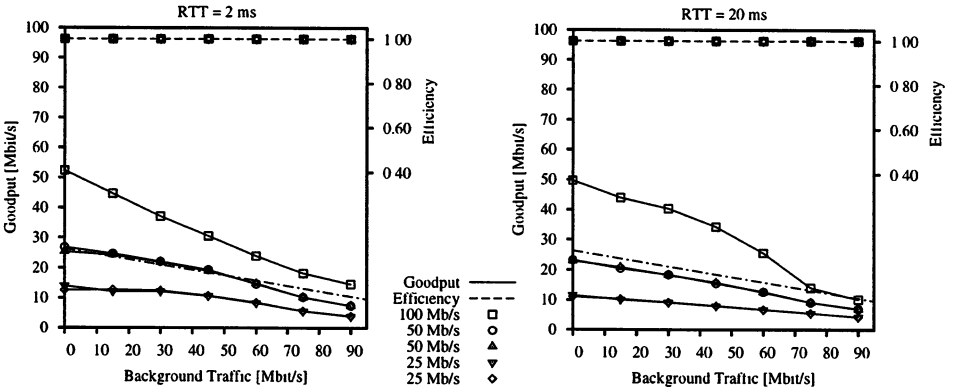

Figure 12 Average goodput and efficiency (left-hand side) with PD+ control with 5 TCP connections of variable PCR

reduce their rate. A small reduction in the transmission rates is sufficient to reach a non congested state, so that connections with higher PCR still keep a significantly higher transmission rate at the end of the decrease phase.

This is an interesting behaviour, in our opinion close to what we would like to obtain when connections with different PCRs share a link; it is important to observe that also in this scenario the efficiency of TCP connections is always equal to 1.

5 CONCLUSIONS

In this paper we presented a simulation analysis of the effectiveness of different control algorithms for the implementation of RRM ABR services within an ATM switch. The algorithms are inspired by standard techniques in control theory, and are designed for simple implementation, requiring only the periodic measurement of buffer occupancy.

The performances of the algorithms were compared in two different scenarios considering ABR connections that transport the traffic generated by sources performing long file transfers using the TCP protocol. The first scenario refers to a very simple network with only two switches, while the second one is somewhat more realistic, with 7 nodes arranged in the configuration generally known as 'parking lot'.

Numerical results show that, at least in the considered environments, the performance differences among the four control algorithms are not striking, but PD+ control tends to reduce the amplitude of the oscillations of the buffer

occupancy; this property is quite beneficial, especially when the delay jitter of the background traffic is considered.

We have studied the ability of the PD+ control to effectively control network behaviour in different scenarios, comprising TCP connections spanning different lengths and TCP connections with different PCR. In all the considered scenarios, the PD+ control yields a high utilization of the link capacity, a good control of the delay jitter, and a fair sharing of network resources.

REFERENCES

[1] ITU-TSS Study Group 13, Recommendation I.371 "Traffic Control and Congestion Control in B-ISDN", Geneve, Switzerland, July 1995

[2] ATM Forum/af-tm-0056.000, "Traffic Management Specification", Version 4.0, April 1996

[3] M.Ajmone Marsan, R.Lo Cigno, M.Munafò, A.Tonietti, "Simulation of ATM Computer Networks with CLASS", 7th Int. Conference on Modeling Techniques and Tools for Computer Performance Evaluation, Vienna, Austria, May 1994

[4] M.Ajmone Marsan, A.Bianco, T.V. Do, L.Jereb, R.Lo Cigno, M.Munafò, "ATM Simulation with CLASS", Performance Evaluation, Vol.24, 1996, pp.137-159

[5] M.Ajmone Marsan, A.Bianco, R.Lo Cigno, M.Munafò, "Some Simulation Results about Shaped TCP Connections in ATM Networks", in: D.Kouvatsos (editor), Performance Modeling and Evaluation of ATM Networks – Vol.2, Chapman and Hall, London, 1996

[6] A. Romanow, S.Floyd, "Dynamics of TCP Traffic over ATM Networks", ACM SIGCOMM'94, London, UK, September 1994

[7] M.Ajmone Marsan, A.Bianco, R.Lo Cigno, M.Munafò, "TCP over ABR: Some Preliminary Simulation Results", *1st Workshop on ATM Traffic Management*, Paris, France, December 1995

[8] R.Stevens, "TCP/IP Illustrated", Vols. I & II, Addison Wesley, 1994

[9] S.Mascolo, D.Cavendish, M.Gerla, "ATM Rate Based Congestion Control Using a Smith Predictor: an EPRCA Implementation", IEEE INFOCOM'96, S.Francisco, CA, USA, March 1996

[10] Y.Zhao, S.Li, S.Sigarto, "A Linear Dynamic Model for Design of Stable Explicit-Rate ABR Control Schemes", ATM Forum/96.0606, April 1996

[11] V.Jacobson, "Berkeley TCP Evolution from 4.3-tahoe to 4.3-reno", *Eighteenth IETF*, Vancouver, BC, Canada, August 1990

[12] M.Ajmone Marsan, A.Bianco, R.Lo Cigno, M.Munafò, "TCP Over ABR in ATM Networks with Variable Topology and Background Traffic", *IEEE ATM Workshop 1996*, San Francisco, CA, USA, August 1996

6 BIOGRAPHIES

Marco Ajmone Marsan is a Full Professor at the Electronics Department of Politecnico di Torino, in Italy. He holds a Dr. Ing. degree in Electronic Engineering from Politecnico di Torino, and a Master of Science from the University of California, Los Angeles. He has coauthored over 150 journal and conference papers in the areas of Communications and Computer Science, as well as the two books "Performance Models of Multiprocessor Systems" published by the MIT Press, and "Modelling with Generalized Stochastic Petri Nets" published by John Wiley. His current interests are in the fields of performance evaluation of data communication and computer systems, communication networks and queueing theory. M. Ajmone Marsan is a Senior Member of IEEE.

Andrea Bianco is a Research Assistant at the Dipartimento di Elettronica of the Politecnico di Torino. He holds a Dr. Ing. degree in Electronic Engineering and a Ph.D. in Telecommunications Engineering both from Politecnico di Torino. His current research interests are in the fields of access protocols for all-optical networks, performance analysis of ATM networks, simulation of communication protocols, and formal description techniques.

Renato Lo Cigno is a research engineer at the Electronics Department of Politecnico di Torino. He received a Dr. Ing. degree in Electronic Engineering from Politecnico di Torino in 1988. Since then he has been with the Telecommunications Research Group of the Electronics Department of Politecnico di Torino, first under various research grants and contracts, then as a staff member. His research interests are in communication networks simulation and performance analysis.

Maurizio Munafò is a research engineer at the Electronics Department of Politecnico di Torino. He obtained a Dr.Ing. degree in Electronic Engineering in 1991, and a Ph.D. in Telecommunications Engineering in 1994, both from Politecnico di Torino. His research interests are in simulation and performance analysis of communication systems.