Optimising bandwidth reservation in IP/ATM internetworks using the guaranteed delay service

C. A. Malcher Bastos
Dept[o] de Eng[a] de Telecomunicações,
Universidade Federal Fluminense
Rua Passo da Pátria 156, 24210-240 Niterói RJ, Brazil
telefax: +55 21 620 3935; e-mail: cmbastos@telecom.uff.br

M. A. Stanton
Dept[o] de Informática, Catholic University of Rio de Janeiro
Rua M. de S. Vicente 225, 22453-900 Rio de Janeiro RJ, Brazil
fax: +55 21 511 5645; e-mail: michael@inf.puc-rio.br
http://www.inf.puc-rio.br/~michael

Abstract

The service model proposed by the IETF for integrated services in IP internets, including the Internet, specifies, amongst others, a *guaranteed delay* service, which offers a deterministic delay for packet delivery. An analysis is presented of implementation aspects of this service, when ATM subnets are used. An evaluation is made of the reserved bandwidth necessary to guarantee a given maximum delay for a given application, taking into consideration alternative strategies for implementing IP over ATM, and factors such as traffic profile, end-to-end hop count, packet transmission with or without fragmentation, and bandwidth reservation for individual flows or groups of flows. It is shown that the reserved bandwidth necessary to guarantee a given delay is reduced by interleaving cells from different packets, by making flow group reservations, and by reducing the number of intermediate nodes. In large-scale internetworks implemented over (similarly large-scale) ATM subnets, switched virtual channels (VCs) have to be used. In such a situation, in order to perform flow group

The original version of this chapter was revised: The copyright line was incorrect. This has been corrected. The Erratum to this chapter is available at DOI: 10.1007/978-0-387-35388-3_42

H. R. van As (ed.), *High Performance Networking*

reservations and to fragment packets, it has to be possible to interleave cells from different packets in the same virtual channel. VC management is also discussed, and the need is shown for procedures to optimise bandwidth and simplify the inclusion of flows in a group. A multilink procedure is proposed, which allows gradual alterations to be made to the bandwidth allocated to a flowgroup, thus permitting the inclusion and exclusion of individual flows in groups. An outline proposal is also presented of a SIA (Server for IP/ATM Integration), an ATM network device for setting-up and release of switched VCs used for traffic aggregation for support of IP flows.

1 INTRODUCTION

In the traditional internetworking architecture based on IP (Comer, 1995), the use of routers is essential for communication between two hosts belonging to separate IP networks, and consequently possessing different IP address prefixes, whereas two hosts in the same IP network can communicate directly. Communication between two stations connected to the same IP network obviously use this network's link-level addressing and protocol. It is therefore necessary to determine link-level addresses. In general, an address resolution protocol (ARP) is required to map an internet address to a link-level address. This type of protocol usually requires the use of broadcast communication facilities of the link-level subnet.

Certain subnet technologies, such as ATM and X.25, which provide a native capacity for routing, can be used to build extensive networks, known as "large clouds". A higher-level application can consider such a large cloud as a single link-level subnet. Frequently, on account of the size of the network, or of the technology itself, broadcast communication is unfeasible, making it impossible to use traditional address resolution mechanisms (ARP). Such networks are called NBMA (Non-Broadcast Multiple Access) (Braden et al., 1994b).

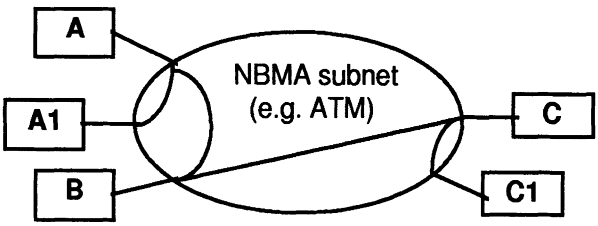


Figure 1 Interconnection of routers in the traditional architecture.

Frequently, such networks correspond to more than one IP network. In such cases, the traditional model for communication between hosts on two such IP networks requires the mediation of one or more routers, even if the communicating hosts are connected to the same link-level subnet. In Figure 1, a traditional IP architecture is assumed, with three routers, A, B and C, connected by IP networks as illustrated. A1 and C1 are hosts which share IP networks with A and C, respectively. Thus A and A1 share a common IP address prefix, as do C and C1. We suppose that these two IP networks are different. We have also supposed that the internet has been configured with an intermediate router, B, between A and C. Thus, should A1 need to communicate with C1, traffic between them will follow the route A1 $\rightarrow$ A $\rightarrow$ B $\rightarrow$ C $\rightarrow$ C1, since routers must be used for communication between hosts (or routers) with different IP address prefixes. However, at the link level, A and C are directly connected, and may communicate without the need to pass through B, in what is usually known as a cut-through or shortcut. (In this case, it is also possible to follow a shortcut from A1 directly to C1.)

ATM has come to be used in both local-area and wide-area networking, and in both public and corporate contexts. Since ATM is a NBMA technology, its use in communication subnets has brought about the development of appropriate standards for interoperating with TCP/IP protocols, even without considering its potential for integrating services.

These standards are RFC 1577 (Laubach, 1994), which describes an address resolution protocol in ATM subnets, RFC 1483 (Heinanen, 1993), describing encapsulation of datagrams when using AAL5, RFC 1626 (Atkinson, 1994), which defines the size of the maximum transmission unit (MTU), and RFC 1755 (Perez, 1995), dealing with set-up signalling for switched connections using UNI 3.1 (ATM Forum, 1995).

The solution based on RFC 1577 is known as "Classical IP over ATM", and had as its primary objective the rapid deployment of IP internets using ATM subnets. In this solution, packets between hosts in different IP networks must pass through intermediate routers, even if the source and destination hosts are connected to the same NBMA subnet. There can only be direct communication between hosts that share the same IP address prefix. Such hosts are said to belong to the same LIS (Logical IP Subnet). Address resolution requires an ATM address resolution server (ATMARP) server, where all stations on the same LIS are registered. This form of packet forwarding completely ignores the communication subnet topology, increasing the hop count between source and destination, and requires packet reassembly in each router, with the introduction of increased delays. For exactly this reason, the Next Hop Resolution Protocol (NHRP) was proposed.

NHRP (Luciani, 1997) allows a host, which needs to communicate over an NBMA subnet, to determine the internet and link-level addresses of the subnet host (or router) nearest to the final destination. If source and destination belong to the same NBMA subnet, then the resulting address will of the destination itself. If they belong to different subnets, the resulting address will be of the exit router to the subnet nearest to the destination. This protocol is not limited only to IP internets, and other groups (such as MPOA of the ATM Forum) are considering its use in other contexts.

These standards currently are designed for best-effort traffic, and make no allowances for real-time traffic or integration of services.

The inefficiency of Classical IP over ATM, doubts about the stability of NHRP (which we do not discuss here), and the large signalling overhead introduced by the use of Q.2931 (ITU-T, 1994) for setting up ATM virtual circuits, has led to the introduction of a new paradigm for routing IP packets in ATM subnets, known as *IP switching* (Callon, 1997). The basic idea is to establish a direct, albeit temporary, relationship between the internetwork and link levels, allowing packet forwarding based only on virtual channel identifiers, and using IP routing. In this fashion, the ATM switch assumes IP routing functions, making packet reassembly and NHRP unnecessary.

The TCP/IP architecture is evolving from a model of point-to-point delivery, with best-effort service, to multipoint delivery, with guarantees of quality of service (QoS) (Braden et al., 1994a). In order to offer performance guarantees, the flow profile of application traffic should be characterised by a token bucket, whose parameters are declared to the network routers, so that the required network resources may be reserved. Among the consequences of QoS guarantees are admission control and the introduction of flow state information in routers, which represent a considerable departure from the traditional model of IP internetworking.

The single most important service parameter to be guaranteed by the network is packet delay. Services currently being standardised include *guaranteed delay* and *controlled load* (Shenker, 1997; Wroclawski, 1996). The former provides deterministic guarantees of packet delay, and to implement this the router must implement an appropriate packet scheduling algorithm. The WFQ (Weighted Fair Queuing) (Demers, 1989; Partridge, 1994; Parekh, 1992) has been much studied and discussed by the working groups involved in developing these standards, and several variants have been proposed to support guaranteed service. This is the service which makes the greatest demands on network resources. There are also voice and video applications currently using the Internet, and these applications perform adequately when network load is low. The controlled load service was designed to cater for these applications, which do not need deterministic guarantees. This service supposes that applications will accept some loss of fidelity, in order to readjust their playback points as a function of the traffic.

The service model maintains the use of unreliable IP datagrams, and introduces dynamic resource reservations in routers which need to be continuously refreshed. Should there occur a routing alteration during a flow transmission, a reservation will be required for the new route, and this may cause a temporary interruption to the service. This is a consequence of the kind of routing used, which seeks the shortest route (in some metric) for each datagram. Current discussions include the use of QoS routing, which could result in offering as a solution a viable route, not necessarily the shortest one.

Today it is almost certain that it will soon be possible to implement services with controlled performance using TCP/IP. However, further study is needed to answer the following questions:

- Can we be sure that the model being proposed will allow the setting up of QoS control for a network as large as today's Internet?
- Supposing an affirmative answer to the first question, will the proposed model make efficient use of network resources, or will it be possible to find ways of further improving resource utilisation?

The present article is a contribution to answering this second question, when an ATM subnet is used for implementing an IP internet.

1.1 Organisation of this article

In section 2, we discuss the basic issues related to integrated services and to the model proposed by the IETF, which are relevant for our needs. Section 3 analyses the network bandwidth that must be reserved for a given application, in order to achieve a given delay when using the guaranteed delay service. The results of this analysis are applied to IP over ATM, and an alternative is proposed for the management of virtual circuits. Finally, section 4 presents the conclusions.

2 ISSUES RELATED TO THE SERVICE MODEL

2.1 The WFQ scheduling algorithm, and Parekh's thesis

The Weighted Fair Queuing (WFQ) algorithm was proposed by Demers (1989), based on a proposal of Nagle (1987), and uses bit-by-bit round robin scheduling, in order to guarantee an application fair access to the transmission bandwidth of a communications link.

Clearly it is not feasible to transmit the packets bit by bit, but, given the number of active flows and the size of a packet, it is easy to calculate the moment when the last bit of the packet would have been transmitted, if the transmission had been done bit by bit. Then the packets should be scheduled for transmission in accordance with the result of this calculation, and this would approximate asymptotically bit by bit round robin. One should also recognise that there will be occasional departures from the bit by bit model, caused by the arrival of a packet after the beginning of the transmission of another, which ought to have been transmitted afterwards.

Finally, it should be recognised that not all flows need to be allocated the same fraction of the transmission bandwidth. Supposing that there are N transmission queues, it is sufficient to subdivide the transmission bandwidth into cycles of M bits, where $M >> N$, and allocate to each flow a bandwidth which is equivalent to the number of bits required. This particular variant is called WFQ.

Parekh's contribution (Partridge, 1994; Parekh, 1992) was to prove that, given a flow *i* whose traffic is shaped on the network boundary using a token bucket, rate-limited by a leaky bucket, and that WFQ is used in all nodes, the overall queuing delay of a packet belonging to flow *i* has a deterministic upper bound given by

$$D_i = \beta_i / g_i + (h_i - 1) \times l_i / g_i + \sum_{m=1}^{i} l_* / r_m \tag{1}$$

where
D_i = maximum delay experienced by a data packet belonging to flow *i*,
β_i = bucket size,
g_i = flow transmission rate, which should be greater than the rate of token generation
h_i = hop count between source and destination,
l_i = maximum size of a packet of flow *i*,
l_* = maximum packet size allowed in the network, and
r_m = transmission rate for hop *m*.

WFQ is not the only kind of scheduling algorithm which can offer guarantees (Bennett, 1996; Georgiadis, 1996), but it is one of those that allows us to guarantee one of the smallest deterministic delays (Partridge, 1994). Recently, several variants of WFQ have been proposed, which are designed to reduce the cost of implementation, which is generally high.

2.2 The guaranteed delay service

The basic idea behind the guaranteed service is that a data flow can be described using a token bucket, and, with this description, a network element can calculate several parameters which will specify how this flow should be handled. Computing and combining the parameters from the different network elements along the route between the source and destination makes it possible to determine the maximum delay that a packet belonging to this flow will suffer when transmitted by this route (Shenker, 1997).

In order to offer delay guarantees, it is necessary to reserve network resources. To specify and make the reservation, an admission control mechanism is required, but the service specification does not deal with this. Additionally, making end-to-end guarantees requires the participation of all intermediate elements.

This service does not deal with the minimum or average delay, just with the maximum delay, not including the path latency, which must be added to the calculated queuing delay. If the application does not exceed the declared traffic parameters, the service also guarantees that packets will not be dropped because of buffer overflow.

The network element should make certain that the service offered is approximated by the fluid model. The fluid model of a service at rate R is equivalent to that offered by a dedicated link at rate R between the transmitter and the receiver, that is, the transmission is considered to be a continuous stream of bits. In this model, the service rendered is completely independent of all other flows.

Such a continuous stream of bits can be modelled by a token bucket with parameters (r, b), where r is the rate of generation of tokens, and b is the bucket size, which is served by a link with transmission rate R.. The definition of guaranteed service is based on the fact that the "fluid delay" of a flow, considered as a continuous stream of bits, is bounded by b/R , provided that $R \geq r$. Each network element must therefore assure that the queuing delay of any packet is bounded by $b/R + C/R + D$, where C and D describe the maximum local variation from the fluid model, since on a shared link it is not possible to transmit packets one bit at a time, simulating exactly the fluid model.

3 ALTERNATIVES FOR IMPLEMENTING IP OVER ATM

In order to implement the guaranteed delay service, it will be necessary to reserve transmission bandwidth along the entire route. To achieve this, the network elements should implement a packet scheduling algorithm which allows such guarantees to be offered.

In this section, a number of alternatives will be described for interconnecting routers using ATM virtual circuits, and these will be analysed in terms of the bandwidth which must be reserved in order to guarantee a particular maximum packet delay, supposing the use of the guaranteed delay service. These alternatives will be compared and related to proposals for IP over ATM.

The work reported in this section was motivated by discussions on the mailing list of the *intserv* working group of the IETF, following a contribution by Keshav (Intserv, 1996). Our principal contributions include the systematic analysis performed for the guaranteed service over ATM, the presentation of quantitative results, and the relationship between these results and some protocols proposed for ATM.

3.1 Guaranteed delay service and packet fragmentation

By packet fragment, we mean a data unit capable of identification and individual forwarding, which may be transmitted interleaved with fragments of other packets. Analysing Figure 2, we note that non-interleaved transmission is equivalent to transmission without fragmentation, in the sense that an entire packet must be transmitted, before the transmission of another one may begin.

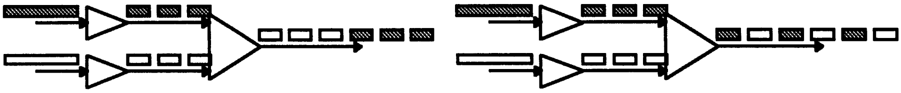


Figure 2 Packet transmission with and without interleaving of fragments.

It is important to emphasise that in an ATM subnet a packet is always fragmented before transmission, since ATM uses fixed length cells. However, in this article, a packet is only considered to be fragmented when its cells may be interleaved with those of other packets in the same virtual circuit. In this sense, AAL5, for example, does not permit packet fragmentation at the VC level.

Equation 2, derived from equation 1, is used to calculate the bandwidth which should be reserved to guarantee a delay d as a function of the transmission parameters, when using WFQ.

$$g = (b + (N-1)P_1)/(d - (N.P_2/r)) \qquad (2)$$

In equation 2, g is the bandwidth which must be reserved, b the token bucket size, N the end-to-end hop count, P_1 the maximum packet size for the application using bandwidth g, P_2 the maximum packet size on the network belonging to other applications, d the desired maximum delay and r the transmission rate of the physical link.

The token generation rate does not appear explicitly in equation 2. For this equation to be valid, g must be at least as large as the application token generation rate. All hops are considered to support the same transmission rate.

The following subsections describe several different alternatives for the use of virtual circuits in implementing IP over ATM, relating them to equation 2. In a later section, this equation will be used to compare the bandwidth requirements of these alternatives for a given guaranteed delay.

3.1.1 Guaranteed traffic aggregation and equivalent flows

Independently of packet fragmentation, which will be analysed below, a further factor for reducing the required bandwidth for guaranteed service is flow grouping. Rampal (1996) has shown that, in certain cases, the required bandwidth for a flow group may be less than the sum of the individual bandwidths. In order to make a group reservation, it is necessary to choose a set of parameters which will be sufficient to cater for the whole group. Thus, considering the example of two flows we have

$$b_g = b_1 + b_2 \quad , \tag{3a}$$

$$\rho_g = \rho_1 + \rho_2 \quad , \tag{3b}$$

$$P_g = \max (P_1, P_2) . \tag{3c}$$

In the equations 3 above, b is the bucket size, ρ the token generation rate, P the maximum packet size, and g indicates the value to be used for the flow group (flow 1 and flow 2). It can be shown that in several cases the bandwidth to be reserved for the flow group with index g is less than the sum of the total bandwidth required for the flows individually (indices 1 and 2). For this result to hold, the bandwidth reserved for the flow group must not be less than ρ_g, the sum of the individual token generation rates.

3.1.2 Interconnection of routers using a single virtual channel

This is the traditional implementation of IP over ATM, proposed for best-effort traffic, with reassembly of packets in each router. If we suppose the use of this option for guaranteed traffic, the resource to be reserved is part of the bandwidth allocated to the VC. The VC must possess capacity sufficient for several guaranteed flows. Alternatively, one can maintain a small number of VCs between pairs of routers. This option is equivalent to interconnecting routers with a dedicated link. In this case, fragmentation of packets will not be caused by the ATM network, as the packets will be reassembled in each router, and will be transmitted serially, without interleaving (Figure 2(a)). Nevertheless, traffic can be aggregated, as proposed in (Rampal, 1996), which can contribute to reducing bandwidth requirements, as will be seen below.

To analyse this case, which corresponds to no fragmentation, we consider $P_1 = P_2$ in equation 2, that is, maximum application packet size is equal to the maximum network packet size.

3.1.3 Interconnecting routers using dedicated VCs for each flow

In this case, an intermediate router receives the cells of a packet, analyses its header, makes a routing decision, and forwards the cells by a specific VC. Packets are encapsulated using AAL5, which implies in reassembly and refragmentation before forwarding. This case is identical to the previous one, except for the fact that cells from different packets can be interleaved, since each flow uses a separate VC. Nevertheless, all cells of a packet must be received by a router before retransmission can occur, and, in spite of the use of fragmentation, there is a reassembly delay at each network router. Since virtual circuits, resource reservation and traffic policing in ATM are performed end-to-end, there is no opportunity here for traffic aggregation or reserved bandwidth sharing, since the virtual circuits begin and end at each router.

To simulate this case, in equation 2, P_1 was chosen to be equal to the maximum application packet size, and P_2 equal to 48 bytes, the payload of an ATM cell. This takes into consideration the fact that a packet needs to be received entirely before being retransmitted, but that it is only necessary to await the transmission of one cell, before retransmission by the router, since cell interleaving is possible (using different VCs). The cell headers have not been considered, as they will affect equally all the cases we are considering.

3.1.4 Interconnecting routers using virtual paths

In this case, bandwidth can be allocated for a virtual path (VP), and this can be shared between different flows, which are multiplexed using different VC identifiers. thus aggregating traffic. From the point of view of fragmentation, this case is equivalent to the previous one. The disadvantage of this alternative is the upper bound of 256 on the number of VPs in a UNI, which reduces its scalability.

3.1.5 Interconnecting routers using a cell-based interface

Another alternative is an interface which functions without packet reassembly, instead processing packets cell by cell. The interface accepts cells from a number of incoming virtual circuits, analyses their content, checks their destination IP address, and takes a routing decision without packet reassembly. Cells are processed and forwarded to outgoing virtual circuits, whenever these are available. Reassembly delay is eliminated and all three alternatives previously mentioned can be considered: VPs, a single shared VC, and separate dedicated VCs.

- The use of VPs permit traffic aggregation, and fragmented transmission of packets. However the upper bound on the number of VPs available continues to hold.
- The use of a single shared VC also permits traffic aggregation. To benefit from fragmentation, packets should be transmitted with interleaving. Cells would be received, processed and forwarded directly to outgoing virtual circuits with interleaving, without awaiting the arrival of all the cells of each packet. Since cells are forwarded with interleaving, without packet reassembly and subsequent segmentation, sharing of a single VC requires a multiplexed identifier field (MID), as used in AAL3/4, which allows the receiver to separate cells from different packets.
- The use of a dedicated VC for each flow allows packet fragmentation without a MID, but, on the other hand, does not support traffic aggregation. An IP switch, based on ATM technology, and which does not use *VCmerge* operations (Callon, 1997), may be considered to operate with dedicated VCs.

From the standpoint of equation 2, all the cases discussed in this subsection are similar, and for the purpose of comparison the size of a cell payload is used, since the ATM header is the same in all cases. To take fragmentation into account, we consider that all network packets have the size of an ATM cell payload, that is, $P_1 = P_2 = P = 48$ bytes. In reality, the application packet, for which we reserve bandwidth, is composed of several fragments (in the traditional sense), which correspond to ATM cells. Thus the delay calculated using equation 2 using the above parameters is that affecting the transmission of just one packet fragment from source to destination. We have therefore to introduce a correction, to include the time required for all fragments to reach the destination, and make a small modification to Keshav's original equation (Intserv, 1996).

Suppose that a packet of size F, composed of fragments of size P, is transmitted at a reserved rate g. This packet will take F/g seconds to be transmitted. If the first fragment takes d' seconds to reach its destination, then for the maximum delay for this packet to be d, the delay for each fragment must be $d = d' + (F-P)/g$, which may be rewritten

$$d' = (d - (F-P)/g) . \tag{4}$$

F is the maximum packet size produced by the application which needs to reserve bandwidth g in order to guarantee a maximum total delay d. The factor $(F - P)$ supposes that d' already includes the arrival of the entire first fragment, that is, after d' seconds we must still await the arrival of the remaining fragments of the packet, except for the first fragment, which will already have arrived. If we now replace d in equation 2 by d', and use equation 4, then, solving for g, we obtain

$$g = (b + (N-2)P + F)/(d - (N.P/r)) . \tag{5}$$

This guarantees that after d seconds, all fragments will have arrived, and the packet will be available for the destination application. Therefore, when considering the case of fragmented transmission, we will use equation 5 instead of 2.

3.2 A quantitative comparison of some alternatives

This section compares a number of alternative approaches for implementing the guaranteed delay service ATM, in terms of their required bandwidth requirements. Using equations 2, 3 and 5, the following pairs of alternatives are studied:

- individual flow reservation, with and without fragmentation;
- individual and group reservations without fragmentation;
- individual and group reservations with fragmentation;
- group reservations, with and without fragmentation.

Comparisons are made through the use of the gain, G, which is the ratio of the bandwidth reservations in the two alternatives considered for each case. In the case of fragmentation, we consider that we have an ATM subnet, that is, the fragment is always of the same constant size. Thus, parameters such as bucket size and packet size can be specified as an integer number of fragments. For purposes of comparison, this same approach is used for the case of no fragmentation. Thus we have:

fragment size = P
application packet size = $P_1 = mP$
network maximum packet size = $P_2 = wP$
bucket size = $b = kP_1 = kmP$

hop count = N
transmission rate = r
required delay = d
reserved bandwidth = g

Based on these definitions, equation 2 can be rewritten as:

$$g = \frac{kmP + (N-1)mP}{d - NwP/r} \tag{6}$$

and equation 5, which deals with the case of transmission with fragmentation, becomes:

$$g = \frac{kmP + (N-2)P + mP}{d - NP/r} \tag{7}$$

The denominators in equations 6 and 7 are similar, except for the factor w, the maximum cell count for network packets. Supposing a large-scale network using a 34 Mbps transmission rate and an end-to-end hop count of 10, the value of the fraction NP/r is 0.11 ms. If we consider an admissible delay of 100 ms, which is considered a reasonable value in discussions on IETF lists, we conclude that this factor is negligible in ATM subnets, where the largest cross packet size is a single cell (w = 1). If, in the case without fragmentation, the size of a cross packet is equivalent to 100 cells, about 4.7 Kbytes, this fraction corresponds to 11 ms, still only 11% of the admissible delay. With higher transmission rates, the influence of this factor will be smaller. In any case, it will only not be negligible in cases of networks without fragmentation and with sufficiently large packets.

In order to simplify the analysis, and for the reasons just discussed, the denominators of equations 6 and 7 will be considered identical. Whenever this approximation is invalid, the effect will be to increase the advantage of fragmentation compared with transmission without fragmentation.

3.2.1 Individual flow reservation with and without fragmentation

The comparison is made using equations 6 and 7. Defining the gain G_{cs} as the ratio of the two left hand sides, after rearranging the terms and division of numerator and denominator by mP we obtain:

$$G_{cs} = \frac{k + N - 1}{k + (N-2)/m + 1}$$

This expression allows some immediate conclusions. For single-cell packets, $G_{cs} = 1$ (a trivial conclusion, since no fragmentation occurs). As the packet size increases ($m >> 1$) G_{cs} tends to $(k + N - 1)/(k + 1)$, that is, the gain is directly proportional to the hop count, N, corrected by the bucket size, k. The larger the bucket size, the smaller the influence of the hop count. Thus, the effect of using a token bucket large enough to contain several packets ($k >> 1$) is to reduce the gain due to fragmentation.

For $N \geq 2$, we note that $G_{cs} \geq 1$. If packets are large ($m >> 1$), or if there are several intermediate nodes, the reserved bandwidth with fragmentation is always less than without. If an application uses a large bucket, the advantages of fragmentation are reduced, but we still have $G_{cs} > 1$, and fragmentation is always advantageous.

Figure 3 shows the dependence of the gain on the hop count, for packets of 22 and 171 cells (1 and 8 Kbytes), with buckets holding one or two packets. In the case of ATM, fragmentation can be achieved by one of the following alternatives:

- use of a VC for each flow,
- use of small packets, corresponding to a kind of pre-fragmentation, or
- use of an adaptation layer permitting cell interleaving in a single VC.

The influence of the hop count is clearly shown.

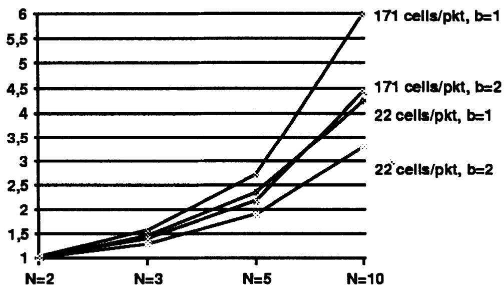


Figure 3 Individual reservations with and without fragmentation. (Delay = 100 ms, Transmission rate = 34 Mbps)

3.2.2 Individual or group reservations without fragmentation

To analyse this case, we must compare the reservation required for z individual flows with that required for a single flow with parameters corresponding to the flow group. We shall analyse the case in which all the individual flows are described by the same parameters. In this case, the equivalent single flow uses a token bucket z times the size of the individual flow (see 3.3.1). The reservation for this equivalent flow is therefore obtained from equation 6, using a bucket size of zb. To obtain the total reservation for z individual flows, we merely multiply equation 6 by the factor z. The resulting gain, G_{zis}, is given by

$$G_{zis} = \frac{zkmP+(N-1)zmP}{zkmP+(N-1)mP}$$

and, dividing numerator and denominator by zmP, we obtain

$$G_{zis} = \frac{k+N-1}{k+(N-1)/z}$$

From this expression, we conclude that group reservations, in the case of a set of similar flows, always reduces the bandwidth reservation, when compared with individual flow reservations ($G_{zis} \geq 1$). Increase in bucket size tends to cancel the advantages of group reservations. The gain depends on the relation between N and z. For increasing values of z, G_{zis} tends to $(1 + (N - 1)/k)$, and G_{zis} is proportional to the hop count, N, when $z >> N$. When $N/z >> k$, the gain tends to z. The influence of the hop count, N, increases in proportion to the increase in z, the number of flows in the group. The only restriction in this case is that the sum total of token generation for all flows must be less than the reserved bandwidth.

3.2.3 Individual or group reservation with fragmentation

In this case, similar reasoning is applied to 7, and the resulting gain, G_{zic} is given by:

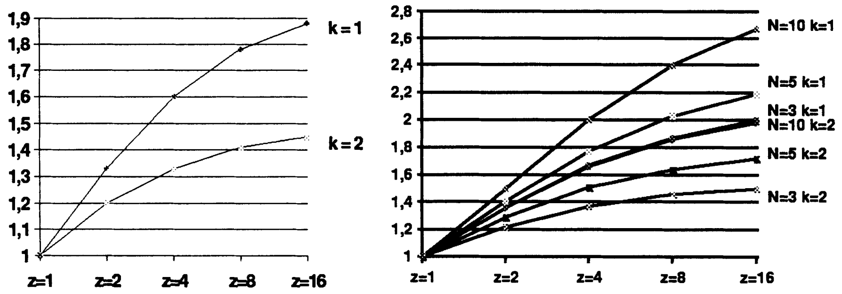
$$G_{zic} = \frac{zkmP+z(N-2)P+zmP}{zkmP+(N-2)P+mP}$$

and, dividing numerator and denominator by zmP, we obtain

$$G_{zic} = \frac{k+1+(N-2)/m}{k+(N-2)/zm+1/z}$$

Note that $G_{zic} > 1$. In this case, we must consider the influence of the packet cell count, m. For small packets, the results are similar to the previous case. As packet size increases, the gain tends to $(k + 1)/(k + 1/z)$, a very different result from the previous case. The value of the gain is always between $(k + 1)/(k + ½)$, for two flows, and $(k + 1)/k$, for a large number of flows ($z >> 1$). For large packet sizes and $k = 1$, the gain varies between 1.33 and 2. The effect of increasing the bucket size is to reduce the gain. The gain also tends to z, with increase in the hop count. The main difference between this case and the case without fragmentation is the dependence of the gain on the packet size.

When the packet cell count is large (Figure 4(a)), the hop count has no importance. The larger the size of the flow group, the greater the advantages of fragmentation, but the influence of the bucket size is also increased. The figure shows buckets of one and two packets. For small packets (Figure 4 (b)), the hop count has a greater influence. The larger the hop count, the greater the advantage of group reservation. With increase in bucket size, the advantages of fragmentation are reduced.



(a) cell count = 171 (8 KB packets) (results independent of N = 3,5,10)

(b) cell count = 8 (384 byte packets)

Figure 4 Individual and group reservations with fragmentation. (Delay = 100 ms, Transmission rate = 34 Mbps)

3.2.4 Group reservations with and without fragmentation

In this case, the gain can be calculated from equations 6 and 7, using a bucket size equivalent to the number of flows in the group, that is, *zkmP*. We thus define the gain G_{zcs}

$$G_{zcs} = \frac{zkmP+(N-1)mP}{zkmP+(N-2)P+mP}$$

and, dividing numerator and denominator by *mP*, we obtain

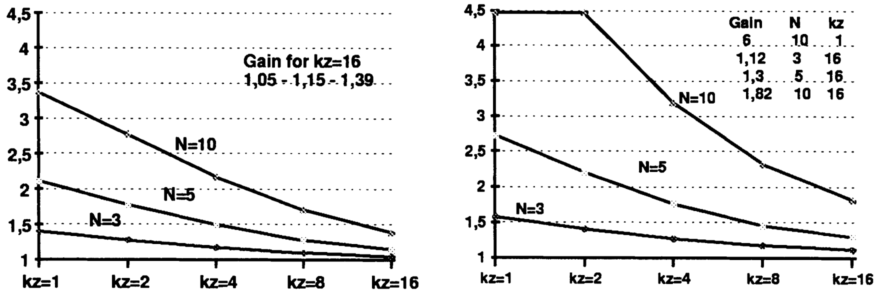
$$G_{zcs} = \frac{zk + (N-1)}{zk + (N-2)/m + 1}$$

The expression for G_{zcs} is identical to that for G_{cs}, the gain in the case of transmission with and without fragmentation, if we substitute *zk* for *k*. This is unsurprising, since group reservation is equivalent merely to increasing the bucket size. All the previous observations are directly applicable to this case. It is important to point out that, as the number of flows increases, the gain tends to 1. This means that for applications in which it possible to group together a large number of flows, it is irrelevant, from the standpoint of bandwidth reservation, whether or not the network deals with packet fragmentation.

On the other hand, for applications with a limited number of transmitters, fragmentation can represent a considerable economy in bandwidth. For example, for $m = 10$, $k = 1$ and $z = 4$ (or $k = z = 2$), we have for $N = 5$, 7 and 10, respectively $G_{zcs} = 1.5$, 1.81 and 2.24, which represents in all cases a significant reduction in bandwidth reservation, even for cell counts of 10 (packets of 480 bytes). When $m >> N$, the dependence on the hop count is linear, whilst if $m <<$

N and $N/m >> zk$, the gain tends to m. It should be remembered that increasing the cross packet size increases the advantages of fragmentation, by making the influence of network packet size non-negligible.

Figure 5 shows the dependence pointed out. For packets of 171 cells and a hop count of 10, we observe the great advantage of fragmentation, even for kz = 16, corresponding to 4 flows and a bucket size of 4 packets.



(a) cell count = 8 (384 byte packets) (b) cell count = 171 (8 Kbyte packets)

Figure 5 Group reservations with and without fragmentation. (Delay = 100 ms, Transmission rate = 34 Mbps)

3.2.5 Conclusions from the analysis

Individual reservations:

- For a hop count of two, there is no advantage in fragmentation, independent of packet size - the gain is close to one. The reserved bandwidth depends strongly on the hop count in the absence of fragmentation. The reserved bandwidth depends directly on packet size for packets transmitted with or without fragmentation. This result can be obtained directly from equations 2 and 5;
- for a hop count of 10, the bandwidth reservation without fragmentation is at least five times as much as for a hop count of two, independent of packet size. Using fragmentation, the reserved bandwidth depends much less on the end-to-end hop count, but for smaller packets this dependence is still important;
- the larger the bucket size, the greater the bandwidth required for a given delay. The influence of bucket size does not depend on the hop count, and is greater when fragmentation is used. Large buckets reduce the benefits of fragmentation, but these benefits are still important;
- the larger the packet size, the greater the influence of fragmentation. For small packets, the gain is around 4 for a hop count of 10, whilst for large packets (cell counts of 171), the gain is around 6 for the same hop count (Figure 3). Thus we can conclude that packets should be fragmented, or the hop count be reduced. As in general we cannot reduce the hop count to 1 (possibly due to administrative restrictions which do not allow a shortcut from

source to destination), fragmentation should always be considered for the case of guaranteed traffic.

- The gain resulting from fragmentation is marginal, when packet reassembly is required in the router. This will only be advantageous when packets are larger than 4 Kbytes and the end-to-end hop count is at least 10. This case was not discussed here, but can be found in (Malcher Bastos, 1997).

Group reservations

As resource reservation and traffic policing in ATM subnets are based on virtual circuits, one should evaluate the benefits of grouping several flows using the same virtual circuit. In this case, we have to consider an equivalent flow, whose parameters are obtained from equations 3. The only restriction is that the reserved bandwidth must exceed the sum of the separate token generation rates for the flows in the group.

- For flows with identical parameters, group reservation is always beneficial, from the standpoint of economising bandwidth, independently of packet size, hop count, flow group size or the use of fragmentation; it is important to compare reserved bandwidth for a group of flows, for the cases of transmission with and without fragmentation. This comparison indicates whether there is any advantage in transmitting packets with cell interleaving, when using a single VC. If the required bandwidth is similar in the two cases, then there will be little point in using fragmentation and, consequently, cell interleaving. In this case, flow aggregation provides the same benefits as fragmentation;
- group reservations reduce the advantages of fragmentation, since the gain due to fragmentation is reduced in proportion to the increase in token bucket size. If the group is large, it may be unnecessary to fragment, but group reservations are always beneficial;
- from the equation for gain, we see that the gain for a hop count of 2 is equal to 1. From Figure 5 we observe that the gain increases with increasing hop count, and decreases when more flows are included in a flow group. The gain also depends on the packet size. For instance, for a hop count of 10 and kz = 8, the gain is around 2, both for large packets (2.32) and small packets (1.71). But for kz = 4, the variation is between 3.2 and 2.2, As a rough approximation, we may say that, independently of packet size, for hop counts between 3 and 5, fragmentation is beneficial for flow groups with $kz \leq 8$. For hop counts between 5 and 10, fragmentation is still advantageous for $kz > 16$.

3.3 Consequences for IP over ATM and the Guaranteed Service

From the standpoint of bandwidth reservation, there are advantages in the use of small packets, if we ignore the effect of increased header overhead. If an application generates large packets, these should be fragmented before

transmission. Small packets at the IP level can be produced by adequate definition of the MTU (maximum transmission unit). If the underlying subnet is cell-based , as in the case of ATM, then the packets can be fragmented at the subnet level, and this taken into account in calculating resource reservations.
An important result is that the interface of the router with the ATM subnet should be capable of processing and transmitting packets cell by cell. Such an interface initiates and terminates virtual circuits, processing packets cell by cell, and making possible the benefits of fragmentation. The technique of IP switching (Callon, 1997), which can associate a dedicated virtual circuit to a specific data flow, when based on ATM, automatically allows packet fragmentation. The IP switching techniques were not developed with this in mind, but it is an performance argument in their favour (Malcher Bastos, 1997). Should the VCmerge operation (Callon, 1997) be used, in order to support fragmentation one must use VPs (VPmerge), or a form of relating cells to a packet or to a transmitter, equivalent to the AAL3/4 MID.

It is important to reduce the end-to-end hop count, as the reserved bandwidth depends directly on this factor. This favours the use of NHRP, which is important for reducing transmission bandwidth, as well as reassembly latency at intermediate routers, which was one of its original objectives. It is interesting to note that we have so far not encountered a single IP/ATM document from the IETF or MPOA document from the ATM Forum, which refers to this matter or points out such benefits for ATM-based IP switching.

Transmitting packets belonging to different flows on the same VC or VP brings already mentioned benefits of bandwidth reduction through traffic aggregation. If VCs are used, fragmentation requires cell interleaving, and we need to know to which packet each cell belongs. One could use AAL3/4, or an equivalent scheme. This will increase the transmission overhead, but even so fragmentation is beneficial.

3.4 Management of virtual circuits

Several documents relating to IP over ATM (Berson, 1997; Borden, 1997; Berger, 1996), produced by the ISSLL (integrated services over specific link layers) working group of the IETF, refer to the management of virtual circuits and traffic aggregation as related items having high priority for further study. One aspect of virtual circuit management is traffic aggregation in a single virtual circuit. The use of a single, high capacity virtual circuit for different flows has the advantage of reducing the problem to one previously solved: the interconnection of routers by high capacity links (Berson, 1997). The problem of bandwidth management would be dealt with at the internet level, which would distribute bandwidth between different service types and specific applications.

Figure 6 shows qualitatively some aspects of bandwidth management of a 34 Mbps link between two network elements, such as used in connecting two routers using several ATM virtual circuits. We consider one VC of capacity 16 Mbps, which is subdivided as follows: 4 Mbps are being shared for flows of type *f*, through a group reservation for the flows *f1*, *f2* and *f3*. Individual reservations have been made for flows *g1*and *h1*. Flows *f4* and *f5*, in spite of being of type *f*, have received individual reservations of 2 Mbps. From the figure it should be noted that, if a group reservation had also been made for *f4* and *f5*, it would still be possible to include in the group a further flow of type *f*. In such a situation, cell interleaving would have to be used to maintain packet fragmentation in the VC. Another option would be to use a dedicated VC for each separate flow, but in this case group reservations would not be possible.

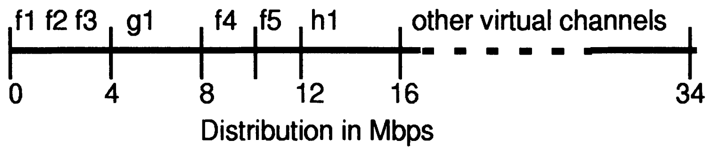


Figure 6 Distribution of data flows in a virtual channel.

In order that a virtual circuit (VC or VP) be set up in an ATM subnet, an end-to-end signalling procedure is required to negotiate parameters and reserve resources in network switches. One advantage of traffic aggregation in a shared virtual circuit is to eliminate, or at least reduce, the amount of signalling overhead.

A disadvantage of use of a single high capacity VC, as described here, is that its management may be complex. If a large bandwidth were allocated to a given channel, this might be underused, producing exactly the opposite of the desired effect. On the other hand, if the reserved bandwidth of a shared VC is already totally used up, and a new flow were to be set up, this flow would not be able to use the existing channel, as the resources allocated would be insufficient. Two solutions are available: either we make an individual allocation for the new flow, with loss of the capacity to exploit traffic aggregation, or we allocate a larger bandwidth for the now larger flow group, and transfer the flows of the old group to this new channel, freeing the existing one.

One difficulty here is the possibility that resources are not available for reallocating a channel for the larger flow group. To avoid undesirable service interruptions, we should first allocate the new channel, and then transfer to it the existing flow group, before releasing the old channel. Obviously, this would not be possible, if the new channel's capacity exceeds the available bandwidth. The alternatives left open to us include: an individual reservation for the new flow; the use of a new route, refusing a flow which could be handled on this one; or the release of the existing shared channel, prior to establishing the new channel, with consequent service interruption, unacceptable here.

Thus we see that traffic aggregation on a single virtual circuit is by no means as simple as perhaps appears at first.

3.5 Multilink procedure for ATM: an alternative for VC management

A further alternative solution to the problem posed at the end of the previous section would be renegotiation of virtual circuit parameters during the phase of data exchange. If it were possible to alter these parameters, to cater for the new flow, the problem would be solved. However, this facility is not currently supported in ATM (ATM Forum, 1995a, 1995c, 1996). The relevant ITU-T recommendations (ITU-T, 1996[a], 1996b) do not deal with renegotiation during the phase of data transmission.

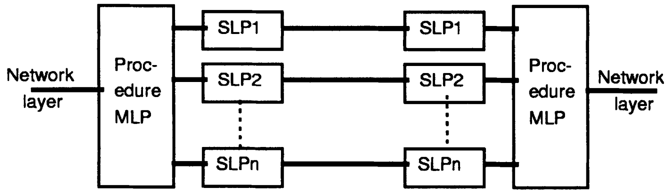


Figure 7 Multilink Procedure.

Another alternative to solve this problem would be a procedure analogous to the multilink procedure proposed for the link level of the X.75 protocol (ITU-T, 1993a). In this procedure, illustrated in Figure 7, there may exist several parallel physical transmission links, each one of which using as link-level protocol an instance of LAP-B (ITU-T, 1993b), called Single Link Procedure (SLP). Between the network layer and the SLP, a Multilink Procedure (MLP) is used to distribute among the different alternative links the packets to be transmitted. The MLP encapsulates these packets in a multilink frame, which includes a sequence number field, to allow the MLP layer to deliver the packets in the correct order.

In the case of ATM, a set of VCs connecting the same endpoints could be used in a similar way by a multilink procedure. Suppose, for example, that an 8 Mbps shared channel were implemented by a multilink group of four 2 Mbps channels. If it were necessary to increase the capacity of this multilink channel to 10 Mbps, it would suffice to add a further 2 Mbps channel. The difference between this procedure and the use of separately allocated channels is in the form of packet transmission. In the case of separately allocated channels, all cells of the same packet follow the same VC. In the multilink procedure, each time a packet is transmitted, its cells will be distributed amongst all channels of the multilink group, fully utilising the capacity of the group. As new flows are included or excluded, new virtual channels can be established or old ones released, permitting operations of traffic aggregation, which are not possible using dedicated channels.

The cells of a packet will need to be identified with a MID and a sequence number, and then distributed among the different VCs for transmission. On arrival, cells will be regrouped by MID and reordered by sequence number. Among the duties of the multilink procedure will be the management of MIDs. Since cells of a single packet will be transmitted in different VCs, the MID will have to be unique within a VC group and not just within a single VC, as is traditional in AAL3/4, for example.

The multilink procedure is equivalent to transmission in a single channel, in the sense that the final bit of a group of packets sent by a group of k channels of bandwidth g/k will arrive at the same instant as would be the case if sent by a single channel of bandwidth g. This is easy to see, if we recognise that, although the transmission rate is reduced by a factor of k, the number of cells in each channel is similarly diminished.

In (Malcher Bastos, 1997), a description is given of a segmentation and reassembly sublayer, which allows the implementing of a multilink procedure. There is also a discussion of the feasibility of such an implementation, including the cost of performing the necessary additional functions, such as management of multiple VCs, distribution of cells on transmission, and their reordering on arrival. For the multilink procedure to be feasible, there must also exist a simple way to allocate bandwidth for a channel group, and that this effort be rewarded in terms of bandwidth economised. This point is also discussed, and a proposal is presented for the introduction of a connectionless service in ATM subnets, incorporating the mechanisms described in this section, which may be utilised for implementing the guaranteed delay IP service over ATM (Malcher Bastos, 1998).

This connectionless service may be implemented with the assistance of a device which we call a SIA (Server for IP/ATM Integration), which is also discussed in (Malcher Bastos, 1997). This device manages the setting up and release of switched virtual channels, permitting traffic aggregation in a single virtual channel, simplifying network resource management and permitting the reduction of bandwidth required for guaranteed service. A SIA only concerns itself with those matters related to the communication subnet, relegating to the IP layer those matters traditionally handled by connectionless servers.

A SIA is an ATM network device, and can simultaneously attend to several end systems and routers, belonging to different LISs. There may be several SIAs in the same ATM subnet, and each of these can operate independently of other SIAs connected to the same network, exercising only local functions. Alternatively, several SIAs may cooperate, communicating through shared SVCs, using cell interleaving and shortcuts, and mapping IP integrated services parameters onto ATM parameters.

A SIA can operate under the control of a router, setting up virtual circuits of specific IP flows, and mapping input channels into output channels, without the necessity for header analysis of each packet processed. This procedure is analogous to an IP switch, which *redirects* IP flows, binding them to specific virtual channels. In a SIA this redirect operation can also be performed locally, avoiding the unnecessary passage through a router of certain packets.

Thus, in order to carry out its functions, a SIA needs to resolve IP addresses using classical IP over ATM, use shortcuts, based on NHRP, which reduces the number of hops between source and destination, perform traffic aggregation on a single virtual channel, using cell interleaving, based on a SAR sub-layer developed specifically for this purpose, and perform redirection of IP flows, as well as traditional redirection, operating in this case like an IP switch.

4 CONCLUSION

We have analysed some aspects of the interoperation of IP internets with ATM subnets, in relation to optimising the bandwidth allocated to an application using the guaranteed delay service in the integrated services model for IP internets. For such a service, the network elements should implement a packet scheduling algorithm based on some variant of WFQ. It is shown here that, in the case of the guaranteed service, if packets are fragmented before transmission, a considerable economy can be made in the amount of network resources which must be reserved, for a given maximum delay. The reserved bandwidth also depends on the end-to-end hop count, and although this dependence is stronger if fragmentation is not used, it still exists even with fragmentation. This result points out clearly the importance of NHRP, even if the network utilises IP switching techniques, which were originally proposed as an alternative to NHRP. Flow group reservations, instead of for individual flows, are another factor which can substantially reduce the reserved bandwidth.

Aspects of VC management were also discussed. Traffic policing in ATM can be performed for VCs or VPs. Using a VP to multiplex flow groups in a UNI introduces scaling problems, since one VP is required for each flow group. Alternatives include sharing a single high capacity VC, or renegotiating the traffic parameters of a connection. Both of these have drawbacks, and we introduced as another alternative the multilink procedure, proposed in (Malcher Bastos, 1997). This procedure allows a finer granularity to be used in bandwidth reservations, without losing the possibility of being used for flow groups. Finally, an outline proposal has been presented of a SIA (Server for IP/ATM Integration), an ATM network device for setting-up and release of switched VCs used for traffic aggregation, typically under control of an IP router, drawing together in one place various of the topics discussed in this article.

5 REFERENCES

Atkinson, R. (1994) *Default IP MTU for use over ATM AAL5.* RFC 1626.
ATM Forum (1995a) UNI Specification 3.1. Prentice Hall, New Jersey.
ATM Forum (1995b) *Traffic Management Specification V. 4.0.*
ATM Forum (1995c) *Lan Emulation 1.0.*
ATM Forum (1996) *User Network Interface (UNI) Signalling Specification V 4.0*

Bennett, J. and Zhang, H. (1996) Hierarchical Packet Fair Queueing Algorithms. *Proc. ACM SIGCOMM 96.*

Berson, S. and Berger, L. (1997) *IP Integrated Services with RSVP over ATM.* Internet Draft draft-ietf-issll-atm-support-03.txt.

Berger, L. (1996) *RSVP Over ATM: Framework and UNI 3.0/3.1 Method.* Internet Draft draft-berger-rsvp-over-atm-00.ps.

Berson, S. and Vincent, S. (1997) *A "Classy" Approach to Aggregation for Integrated Services.* Internet Draft, draft-berson-classy-approach-00.txt.

Borden, M. et al. (1996) *Issues for RSVP and Integrated Services over ATM.* Internet-Draft draft-crawley-rsvp-over-atm-00.

Braden, B., Clark, D. and Shenker, S. (1994a) *Integrated Services in the Internet Architecture: an Overview*, RFC 1633.

Braden, B. et al. (1994b) *Internet Architecture Extensions for Shared Media,* RFC1620.

Callon, R. et al. (1997) *A Framework for Multiprotocol Label Switching.* Internet-draft, draft-ietf-mpls-framework-00.txt.

Comer, D. (1995) Internetworking with TCP/IP, Prentice Hall, New Jersey.

Demers, A et al. (1989) Analyses and Simulation of a Fair Queuing Algorithm. Proceedings. ACM SIGCOMM.

Drury, D. (1996) ATM traffic management and the impact of ATM switch design. *Computer Networks and ISDN Systems* **28**, 471-479.

Garrett, M. (1996) *Interoperation of Controlled-Load and Guaranteed-Service with ATM.* Internet-draft draft-ietf-issll-atm-mapping-01.txt.

Georgiadis, L. et al. (1996) Efficient Support of Delay and Rate Guarantees in an Internet. *Proc. ACM SIGCOMM 96.*

Heinanen, J. (1993) *Multiprotocol Encapsulation over ATM Adaptation Layer 5.* RFC 1483.

Intserv (1996) From the IETF Integrated Services (intserv) discussion list, after the publishing on the list of the paper, "A study of bandwidth requirements for guaranteed service traffic with WFQ scheduling" by S. Keshav

ITU-T (1993a) Recommendation X.75, *Packet-Switched Signalling System Between Public Networks Providing Data Transmission Services.*

ITU-T (1993b) Recommendation X.25, *Interface Between D C E and D T E for Terminals Operating in the Packet Mode on Public Data Networks.*

ITU-T (1994) Recommendation Q.2931 *B-ISDN User-Network Interface Layer 3 Specification for Basic Call/Bearer Control.*

ITU-T (1996a) Recommendation Q.2962 *Digital Subscriber Signalling System No. 2-Connection characteristics negotiation during call/connection establishment phase.*

ITU-T (1996b) Recommendation Q.2963.1, *Digital Subscriber Signalling System No. 2-Connection modification: Peak cell rate modification by the connection owner.*

Laubach, M. (1994) *Classical IP and ARP over ATM.* RFC 1577.

Luciani, J. et al. (1997) *NBMA Next Hop Resolution Protocol (NHRP).* Internet draft draft-ietf-rolc-nhrp-11.txt.

Malcher Bastos, C.A. (1997) Connectionless service for Guaranteed Traffic in IP internets over ATM (in Portuguese). Doctoral thesis. Dept° de Informática, Catholic U. of Rio de Janeiro, Brazil.

Malcher Bastos, C.A. and Stanton, M.A. (1998) Efficient Support for Guaranteed Service in IP over ATM, accepted for presentation at 9th IEEE Workshop on LAN/MAN, Banff, Canada.

Nagle, J. (1987) On Packet Switches with Infinite Storage. *IEEE Trans Com,* **35** No 4.

Partridge, C. (1994) Gigabit Networking. Addison Wesley Pub. Co.

Parekh, A. (1992) *A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks,* Technical Report LIDS-TR-2089, Laboratory for Information and Decision Systems, Mass. Inst. of Tech.

Perez, M. et al. (1995) *ATM Signalling support for IP over ATM.* RFC 1755.

Rampal, S. (1996) *Flow Grouping for Reducing Reservation Requirements for Guaranteed Delay Service.* draft-rampal-flow-delay-service-00.txt,

Shenker, S. et al.(1997) *Specification of Guaranteed Quality of Service.* Internet-draft draft-ietf-intserv-guaranteed-svc-07-txt.

Wroclawski, J. (1996) *Specification of the Controlled-Load Network Element Service.* Internet Draft draft-ietf-intserv-ctrl-load-svc-04.txt.

Zhang, L. et al. (1993) RSVP: A New Resource Reservation Protocol, *IEEE Network,* September 1993.

6 BIOGRAPHY

Carlos Alberto Malcher Bastos

Associate Professor in the Telecommunications Engineering department of the Universidade Federal Fluminense, where he has occupied several technical and administrative positions, and where he has been active for several years in data communications and computer networks. He holds a doctoral degree in computer science, and his current research interests are high-speed networks, ATM, integrated services and IP over ATM.

Michael Anthony Stanton

Originally from Manchester, England, gained his B.A and Ph.D. at Cambridge University, before settling in Brazil. Presently holds the positions of Associate Professor at the Catholic University of Rio de Janeiro, and Full Professor at the Universidade Federal Fluminense, both in the field of computing. Current research interests are in computer networking and distributed systems, with emphasis in high-speed networking, network management and security.