Traffic Management Tools for ATM Networks With Real-Time and Non-Real-Time Services

Kalevi Kilkki
Helsinki University of Technology
e-mail: kalevi.kilkki@hut.fi

Abstract

This presentation considers some practical issues of traffic management in ATM (Asynchronous Transfer Mode) networks. Firstly, we illustrate a fundamental dilemma of statistical multiplexing in ATM networks when small loss ratio, small delay and high utilization are expected at the same time. Although there are a lot of solutions to this problem, all of them have some disadvantages. Therefore it is necessary to offer several service types meeting the differing user demands. A minimal amount of service types is desirable because of the additional expenses of nodes and network management when implementing several services. The proposal in this paper is to offer only two basic services: the first one covers real-time CBR (Constant Bit Rare) and VBR (Variable Bit Rate) services and the other non-real-time service is aimed especially for the current Internet applications. The main content of this paper is to present the traffic management tools needed for the real implementation of these services.

1. Introduction

One of the inherent properties of ATM networks is the diversity of services. Firstly, the mean and peak cell rates of connections may vary enormously based on the application used by the customers. Secondly, customers have very different Quality of Service (QoS) requirements. A network operator has a difficult task to meet all the differing requirements and keep the network utilization high at the same time. Let us take a simple example where the link capacity is 130 Mbit/s and there are 6 on/off sources with peak rate 25 Mbit/s and mean rate 5 Mbit/s (Fig. 1). Evidently, the overload probability in this case is so high that the QoS will be somehow deteriorated. One of the main tasks of traffic management in ATM networks is to solve this fundamental problem.

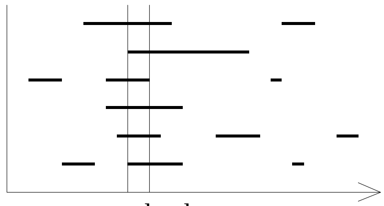

Fig. 1. Traffic demand of 6 sources and an overload situation.

The solutions can be divided into the following groups:

1. *Traffic shaping*. In this approach all (or at least almost all) traffic variations are smoothed before the ATM network. The disadvantage of this approach is apparent: there must be huge buffers at customer interfaces, because the time scale of traffic variations could be several seconds or even minutes and peak rate could be as high 25 Mbit/s as in the example. These buffers are expensive and may bring about an unacceptable long delay and delay variation.
2. *Admission procedure for each connection.* It is possible to avoid overflow situation by rejecting each connection request that endangers the QoS of other connections. In the above case, this means that at most 5 connections (or perhaps 4) will be accepted on the link. Some network operators may consider the average load obtained with this approach far too low. However, with certain applications this scheme might be practical, or even unavoidable.
3. *Admission procedure for each information block*. It is possible to do the capacity checking separately for each information block. This protocol is called ATM block transfer capability (ABT) [1]. There are two different variants of this service: ABT with delayed transmission and ABT with immediate transmission. In the first approach, each information block will be sent to the network without any preceding acceptance procedure, and in the second one there is a CAC type of procedure before the sending of each information block. With both approaches, a sufficient, and probably quite large, buffer capacity is needed at the user interface because of the relatively high probability that there is not enough capacity in the network just when the user wants to transmit the information block. A large buffer means always possibility for long delay.
4. *No admission control and small buffers*. In this approach all connections will be accepted in the network and due to small buffers high cell loss ratios are unavoidable. The advantages of this approach are high utilization and short delay. But at the same time, because of the high cell loss ratio the benefit obtained by customers could be small, or even nonexistent, during an overload situation. One method to improve the situation is to classify cells as important and less important cells.
5. *No admission control and very large buffers*. In the previous approaches it is possible to use relatively small buffers at each network node. We can also use very large buffers in order to store all cells that cannot be immediately transmitted forward. However, because of the high speed of ATM-networks and the properties of data traffic this buffer should be really large if we want to avoid cell losses and at the same

time keep the utilization high. For instance, if in our simple example the overload situation lasts only 1 s, the needed buffer capacity is almost 50 000 cells. Furthermore, if we have huge buffers there will also be very long delays and large delay variation.

6. *No admission control, large buffers and intelligent scheduling.* Because the buffer capacity will in practice be restricted there should be rules for the acceptance and rejection of cells during overload situation. This selection should be efficient on the network point of view and, above all, fair for all users. In our example, all users will lose some cells when the buffer occupation exceeds certain limit. It is the responsibility of each user to react properly on the increased cell loss ratio.

7. *Reactive control.* All the above approaches are based either on preventive control or on the assumption that users can react properly on the impaired QoS caused by a network overload. A further approach is to construct detailed protocols for regulating the traffic process sent to the network, as in the Available Bit Rate (ABR) service [1]. It should be noted that this approach requires large buffers both at the network nodes (because of high bit rates and round-trip delays), and at user interfaces since we can hardly suppose that all applications can immediately adapt to any changes of available bit rate. The main advantage of this approach is that it might be possible to keep the cell loss ratio low and utilization high, but again, at the expense of long delays.

What conclusions can we draw from the above general considering? First of all, Fig. 1 shows that it is impossible to attain at the same time small cell loss ratio, short delay and high network utilization for all connections if there is a considerable amount of data traffic with high burstiness and high peak rate. This conclusion can be illustrated by a tetrahedron shown in Fig. 2.

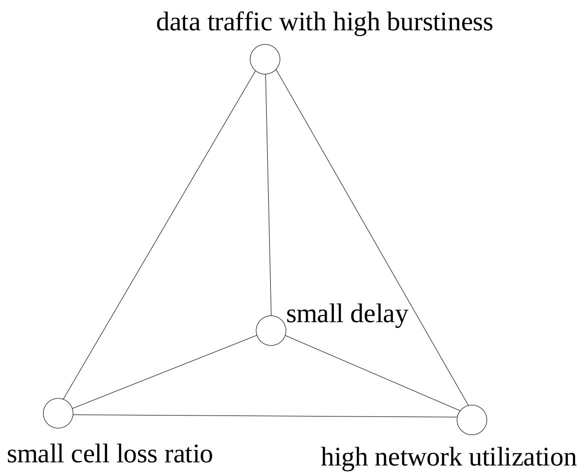

Fig. 2. Four targets of ATM services [2].

By adjusting the above 7 service items in the framework of Fig. 2 we can find 4 basic services:

1. Constant bit rate service with small cell loss ratio, small delay, high utilization, but without the possibility to transmit bursty data traffic as such (item 1).
2. Variable bit rate service for bursty data traffic with small cell loss ratio and small delay, but without high network utilization (item 2).
3. Unspecified bit rate service with small delay and high utilization but without any hope for small cell loss ratio (item 4).
4. A network service with small cell loss ratio and high utilization, but long delay for the application during overload situation. In item 3 the delay arises at user interface (because all information blocks cannot be transmitted immediately) and in items 5, 6 and 7 mainly at network nodes but also at user interfaces.

In consequence, the network operator must abandon either small cell loss ratio, small delay, high utilization or bursty data traffic with high bit rate. In other words, any network service forms a plane which goes through 3 edges of the tetrahedron but misses one of the edges. Of course, it is always possible to offer several network services at the same time and meet all the four targets for different connections: small delay and small cell loss probability for CBR connections and high utilization with UBR (Unspecified Bit Rate) type of service offered to data traffic. One of the main questions of the network operator is what is the best selection of the ATM services to be offered to customers. The rest of this paper is dealing with this question.

2. Service Combination

Internet and its explosively increasing traffic demand is clearly one of the most important elements as regards the near future of ATM networks. The applications running in the current Internet have to adapt to the service characterized by long delay and high frame (or packet) loss ratio. It is assumed that a higher protocol level can adapt to this, somewhat poor, QoS by re-sending the lost frames and by decreasing the bit rate after frame losses. It is very probable that this service model will prevail several years although the ATM technology will give new opportunities and a lot of additional capacity for Internet. Therefore, in an ATM network there has to be a service which meets this actual service demand. As to the above presented service items the most suitable service model for this demand is item 6: no admission control, large buffers and intelligent scheduling. This approach will be addressed more thoroughly later in this paper.

However, as Scott Shenker has stated in his article [3], this UBR service is not sufficient but, in addition, an ATM network operator should offer a service which satisfies the requirements of real-time applications like voice and video. Many real-time applications are intrinsically variable bit rate sources. The basic problem is how we can exploit the statistical multiplexing of VBR connections when we have a very stringent requirement for cell loss ratio and we only know the traffic process approximately. This is a hard problem but after extensive studies there are at least promising solutions (although the final assessment cannot be made until we have experience in real networks). One possible scheme is presented in the following chapter.

Thus, our suggestion is that a basic ATM node should have two service types: a service for real-time CBR and VBR traffic with high QoS requirements, and an inexpensive UBR service for applications capable to adapt into changing capacity and poor QoS. Although there certainly are applications not very suitable for these two service types, most of the current user demand can probably be satisfied with these services. The main advantage of this approach is that the integration of the two service types in an ATM node is quite straightforward. The only requirement is that the node must separate the services by a strict priority: if there is any cell in the buffer belonging to the real-time service class, UBR cells are not transmitted forward. As for the traffic management of real-time services, the ATM node looks (almost) like a simple switch with only one service class. The only complication emerges if the operator wants to guarantee a minimum cell rate even for some UBR connections; this cell rate should be taken into account in the CAC procedure of the real-time services.

3. Realization of Real-Time Service for CBR and VBR Connections

The basic requirements for the realization of a real-time service are small delay, (usually) small cell loss ratio, and possibility to exploit statistical multiplexing. Because of the small delay requirement preventive control approach is the most feasible one. In order to realize this scheme the following traffic management components are needed: traffic descriptor for the characterization of each connection, usage parameter control (UPC) for controlling the traffic sent to the network by each user, and connection admission control (CAC) to decide whether a new connection can be accepted into the network.

The properties of a connection are characterized by four traffic parameters: peak cell rate (PCR), sustainable cell rate (SCR), cell delay variance tolerance (CDVT) and intrinsic burst tolerance (IBT) [1]. These parameters are then controlled by UPC which accepts certain traffic patterns are rejects some others. Although it is not quite clear on the network performance point of view which one of the acceptable traffic patterns is absolutely the worst one, the pattern presented in Fig. 3 is at least near the worst and due to the simplicity suitable for practical performance evaluation.

Traffic parameters and UPC are well defined but what has been missing is a simple and efficient CAC that guarantees high QoS. The fundamental problem of this task is that the aggregate traffic process is very difficult to model with any simple formula. The solution presented in [4] is to firstly divide the problem into three time scales: cell, burst and rate-variation scale.

Fig. 3. A near worst case traffic pattern with given PCR, SCR, SDVT and IBT;
M = the maximum number of consecutive cells; *L* = the maximum burst size in cells.

The cell scale process is composed of the short term fluctuations in the traffic process. As to the (near) worst case traffic, cell scale takes into account the maximum number of consecutive cells (M). Now we can neglect temporarily the other traffic parameters and attempt to calculate the allowed load (denoted by ρ_{cell}) taking into account only this traffic process. A feasible approach is to apply an $N*D/D/1/K^*$ model where the real buffer capacity (K, in cells) is replaced by $K^* = K/M$. This approach offers a good basis for practical calculations, and for the use of pre-calculated tables. The superposition of cell scale variations can modeled by using the effective bandwidth method.

The other extreme case is the rate-variation scale where the traffic fluctuations are slow. In practice this assumption is valid always when the maximum number of cells in a burst (L) is larger than the buffer capacity (see [4]). In this case we can ignore the effect of buffer capacity and consider only peak and sustainable cell rates (e.g., [5]). Large deviation approximation is very useful when evaluating homogeneous cases, and effective variance method gives good approximation for the superposition of different sources, see e.g. [6] and [7].

The most difficult area exists between the cell and rate-variation scales. Unfortunately, it is very difficult to solve the allowed number of connections with the exact deterministic process presented in Fig. 3. One simpler approach is to replace the deterministic process by a Markovian one with the same mean and peak rates and to neglect the cell clumping due to cell delay variation. This Markovian model is solvable although the solution needs a lot of numerical processing. Fortunately, our recent studies indicate that it is possible to develop a simple approximation for this burst scale process.

The remaining problem is to combine the results of the three scales. Our solution is a simple combination of effective bandwidth and effective variance models where a new connection is accepted if the following condition is valid [4]:

$$\sum_i m_i + \sqrt{\left(\sum_i \sigma_i^*\right)^2 + \sum_i v_i^*} \le \rho_{max} C \qquad (1)$$

where: m_i = sustainable rate, C = link capacity,

$$\sigma_i^* = \sigma_{i,cell}^* + \sigma_{i,burst}^*,$$

$$\sigma_{i,cell}^* = m_i\left(\frac{1}{\rho_{i,cell}} - 1\right),$$

$v = v_{i,burst}^*$ or $v = v_{i,rv}^*$ depending mostly on the maximum burst size to buffer size ratio, and

$$v_{i,rv}^* = SCR(PCR - SCR)\left(c_1 + \frac{c_2}{e^{c_3 SCR/PCR} + (PCR/C)^{c_4} + c_5}\right).$$

The tuning of coefficients c_1, c_2, c_3, c_4 and c_5 may be done by using a genetic optimization method [7]. The approximate formulae for burst scale parameters ($\sigma^*_{i,burst}$ and $v^*_{i,burst}$) are still under study but interim simulation studies have shown that an appropriate accuracy is obtainable. In consequence, it is possible to develop a relatively simple CAC method that takes into account the four traffic parameters, all time scales and gives good possibility for statistical multiplexing.

Our performance evaluation has covered the following parameter region: PCR/C = 0.0002 … 0.25, SCR/C = 0.0001 ... 0.20, SCR/PCR = 0.001 ... 1, M = 1 ... 20, L/K = 0.01 ... 3. The most difficult parameter combination is when PCR/C > 0.1 and SCR/PCR < 0.1. Then the real cell loss ratio depends in a complicated way on the parameters of other connections and even (1) fails in some cases to catch the real behavior of the aggregate traffic process. However, the overall results presented in Table 1 are very promising. As the total number of cells in the simulation is 16 400 000 000, the results provide a good assessment of the performance of the CAC algorithm. Yet, it should be noted that the worst case represents 22% of the total amount of lost cells, and there may be even worse cases not occurred in our simulations.

Table 1. Results of performance evaluation of CAC algorithm, the target cell loss ratio = 10^{-5}, K = 100, in each case 5 000 000 time slots are simulated using one randomly selected source type and CBR load (0 … 99%).

selection of cases	number of cases	number of lost cells	average cell loss ratio
$P_{loss} > 2\cdot10^{-4}$	1	1939	$3.95\ 10^{-4}$
$2\cdot10^{-4} > P_{loss} > 10^{-5}$	41	6028	$4.83\ 10^{-5}$
$10^{-5} > P_{loss} > 2\cdot10^{-7}$	116	933	$2.50\ 10^{-6}$
$P_{loss} = 0$	4582	0	0
all	4740	8900	$5.42\ 10^{-7}$

4. Realization of a Fair UBR Service

The main aspect of UBR service is fairness. In order to provide bandwidth fairness among competing UBR connections, the buffer capacity at each switch needs to be allocated fairly among the competing connections. Otherwise a connection that gets more than its fair share of the buffer space, also gets more than its fair share of the bandwidth. The allocation scheme presented in this paper allows a fair share of link capacity using only a pure FIFO buffer [8].

The basic idea of the proposed scheme is that the buffer implementation should be as simple as possible, whereas we may allow a relatively complex algorithm to decide whether an incoming cell should be accepted or rejected. Let us suppose that the incoming cells to an ATM node are generated by several sources, all of which send AAL5 frames (then we know when a frame ends). Firstly, it is important to know whether the arriving cell is the first cell of a frame, and whether some cells of the same frame have already been delivered forward. If there is an impending danger of buffer overflow, whole frames should be dropped instead of individual cells. Therefore, if the buffer occupancy exceeds a certain limit, the first cell and all the following cells of a frame should be dropped. Consequently, if the first cell of a frame is accepted into the buffer, all the following cells of the frame will be accepted, provided that the buffer is not fully occupied.

Now the first algorithm (*A1*) can be defined as follows:

- The first cell of an AAL5-frame is dropped if $X > R$, where X is the number of cells in the buffer and R is a limit for buffer occupancy.

Unfortunately, this buffer allocation scheme does not guarantee a fair share between different type of connections if some connections have exploited more resources than the other ones. If the network operator's intention in an overload situation is to share the link capacity evenly among all active connections, then the operator should drop cells from those connections that have exploited the largest part of the link and buffer capacity. One way to do this is to consider the number of cells in the buffer. Let us denote the number of cells connection i has in the buffer by Y_i and the number of connections which have at least one cell in the buffer by N_a. If during an overload situation a connection has more cells in the buffer than the average value (i.e., $Y_i > X/N_a$) , we may conclude that the connection is to some extent responsible for the overload situation. Let us denote

$$W_i = \frac{Y_i N_a}{X} . \qquad (2)$$

Parameter W_i can be used as a measure of the exploitation of network resources by connection i provided that the buffer capacity is sufficiently large in comparison with the typical burst size. Now we can define an advanced algorithm by applying parameter W_i. In this algorithm (*A2*) the first cell of an AAL5-frame is dropped if

$$(X > R) \text{ and } (W_i > 1).$$

Although this simple scheme levels down in some degree the differences in link capacity used by connections, the result is not quite satisfactory. When X exceeds R all connections which have more cells in the buffer than average (i.e., $W_i > 1$) experience roughly the same frame loss ratio, independently of the instantaneous rate of the connection.

In order to improve the performance of the algorithm we can replace the on/off type rejection function by a smoother one. In the third algorithm (*A3*) the first cell of an AAL5-frame will be dropped if:

$$(X > R) \text{ and } W_i > Z\left(1 + \frac{K - X}{X - R}\right), \qquad (3)$$

where K is the buffer capacity in cells and Z is a free parameter (typically less than 1).

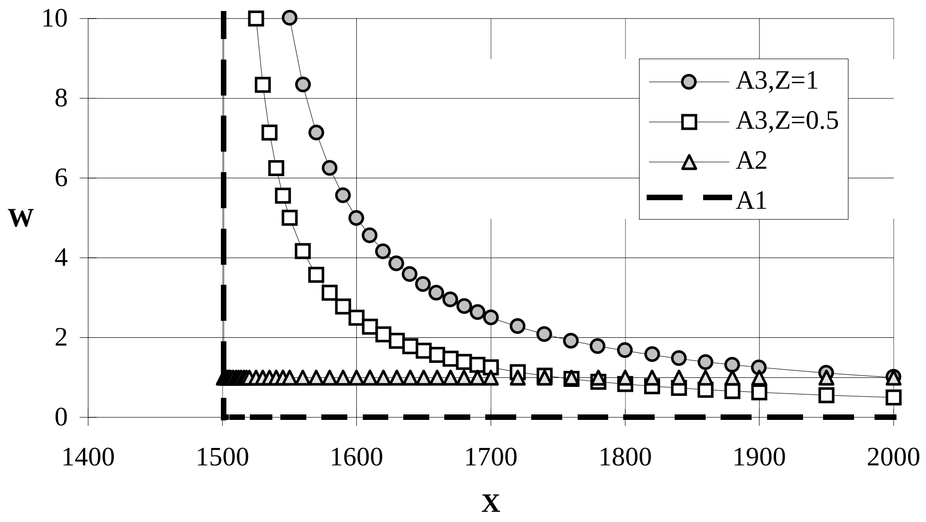

Fig. 4. Rejection functions of Algorithms 1, 2 and 3; K = 2000, R = 1500.

The shapes of the rejection functions of different algorithms are presented in Fig. 4. If we apply Algorithm 3 and the buffer occupancy exceeds R, cells are rejected from a connection only if it has a considerably amount of cells in the buffer. When X approaches K, the allowed number of cells may decrease eventually below 1. As a consequence, finally most of the first cells of AAL5-frames will be dropped, because the scheduling algorithm tends to level down the differences in the number of cells while X is increasing. This property is desirable since some of the buffer capacity should be left for the remaining cells of accepted frames. An appropriate behavior of algorithm can be obtained by a proper selection of parameters R and Z.

The results presented in [8] have shown that the proposed algorithm makes it possible to offer high fairness with a simple queuing principle. Especially important is that the connections with moderate bit rate may remain totally undisturbed as regards the cell losses although the delay variation is similar to all connections. If the higher layer protocols work properly, the overload situation finishes after some delay and the highest bit rates are cut to a certain value.

5. Conclusion

The goal of this paper has been to show that it is possible to meet most of the real user demand by a simple ATM switch with two service classes: a real-time service class for CBR and VBR connections and UBR service class for applications with loose QoS requirements. The main advantage of this simple approach is that the integration of the two service types in an ATM node can be realized using a simple priority scheme.

Real-time services need a feasible preventive control method. This control method can be composed of three parts. Two parts are well defined in ITU's specifications, namely, traffic descriptor and usage parameter control. This paper presents the outlines of the third part, Connection Admission Control. Our investigations, although yet in preliminary stage, show that our framework forms a good basis for practical implementation with good performance.

The other necessary part of traffic control is a fair scheduling algorithm for UBR service. This paper shows that it is possible to attain high fairness by using FIFO buffers and a simple allocation scheme. The advantage of the proposed allocation algorithm is that it reacts accurately and quickly during overload situations and reduces the bandwidth of only those connections that are using an excessive amount of the link bandwidth. At the same time, connections with small cell rate do not experience any cell losses. By using weighting coefficients it is possible to allocate link capacity according to pre-defined values. However, further simulations with several nodes and upper protocol levels are needed to assess the real performance of the buffer allocation schemes.

References

[1] ITU-T, Recommendation I.371, Traffic Control and Congestion Control in B-ISDN, Geneva, May 1996.

[2] J. Virtamo, Inauguration speech, 18.9.1995, Otaniemi, Espoo, Finland.

[3] S. Shenker, Fundamental Design Issues for the Future Internet, IEEE J. on Selected Areas in Communications, Vol. 13, No. 7, Sept. 1995.

[4] K. Kilkki, Traffic Characterisation and Connection Admission Control in ATM networks, Doctoral thesis, Espoo 1994.

[5] J. Roberts (ed.), Performance Evaluation and Design of Multiservice Networks, COST 224 Final Report, Luxembourg, 1992.

[6] K. Kilkki, Connection Admission Control Methods in ATM networks, International Teletraffic Seminar, Bangkok, 1995.

[7] S. Saranka, K. Kilkki, Optimization of Effective Variance Based CAC Algorithms, to be appeared in Globecom'96, London, 1996.

[8] J. Heinänen, K. Kilkki, A Fair Buffer Allocation Scheme, submitted for publication, 1995.