INTEGRATED DYNAMIC DISTRIBUTED ROUTING AND ADMISSION CONTROL IN ATM NETWORKS

R. BOLLA[1], A. DALAL'AH[1], F. DAVOLI[1], M. MARCHESE[1] AND M. S. OBAIDAT[2,*]
[1]*Department of Communications, Computer and Systems Science (DIST), University of Genoa, Via Opera Pia 13, I-16145 Genova, Italy*
[2]*Department of Computer Science, Monmouth University, W. Long Branch, NJ 07764, USA.*

SUMMARY

A node-by-node admission control and routing scheme for ATM networks is devised. The scheme is based on the subdivision of traffic into a number of classes, characterized by different performance requirements. At each network node, for all outgoing links, link capacity partitions are periodically assigned to the traffic classes, as the result of an optimization problem over a fixed time interval. Local access control rules compute the maximum number of connections of each class that a link can accept within the assigned capacity. Incoming call connection requests are forwarded in a hop-by-hop fashion. Each node traversed, first checks the presence of resources needed to accept a new connection and guarantee all quality of service (QoS) requirements. This is done by using the local access control rule. Then, it chooses the next node along the path on the basis of a distributed routing strategy. This minimizes a cost function accounting for local instantaneous information, as well as for aggregate information that is passed periodically among adjacent nodes.

Two routing strategies are introduced. In the first scheme, a new call is rejected if, at a certain node along the path, there are not enough resources to guarantee QoS requirements, and no recovery mechanism is implemented. In the second scheme, an alternative path is looked for after the first failure. Simulation results are presented which show a comparison between the two proposed routing strategies. Comparison is also made between the proposed scheme and the other approaches. © 1997 by John Wiley & Sons, Ltd.

Int. J. Commun. Syst., **10**, 215–225 (1997)

No. of Figures: 11 No. of Tables: 1 No. of References: 28

KEY WORDS: ATM traffic control; quality of service; admission control; dynamic routing

1. INTRODUCTION

One of the main problems arising from the statistical multiplexing nature of the asynchronous transfer mode (ATM) technique, is that of guaranteeing Quality of Service (QoS) requirements in the presence of traffic flows that exhibit different statistical and performance characteristics. In this respect, a larger amount of control has to be exerted at the network boundaries (and within the network itself) than that required in other transfer modes (e.g. STM), where the allocation of bandwidth is somehow more structured. This particular aspect of ATM has given rise to a great deal of investigation into bandwidth allocation [see References 1–4, among others], admission and congestion control[5–15] and routing (to a lesser extent).[16–18] Among the possible approaches are control architectures that impose a certain structure on the allocation of the resources considered, where typically the traffic is subdivided into classes that are homogeneous in terms of statistical or performance characteristics.[2,19–25] This often allows the decomposition of a very complex overall control task, that is generally characterized by very different time scales and requirements, according to the level where the system dynamics are considered (e.g. cell and call level) into smaller and somehow independent problems. For instance, an essential decoupling between cell and call level is achieved in Reference 20, through the concept of schedulable region, whereas a hierarchical decomposition has been used by authors in previous papers, such as in References 22–25, by adopting an approach already introduced in the context of TDM systems.[26–27]

In this paper, we devise a new integrated admission control and routing scheme that is based on the above strategy. More specifically, at each ATM network node, several traffic classes share the outgoing links. Each traffic class is characterized by statistical parameters (such as peak and average bandwidth), as well as QoS requirements at the cell level (in terms of cell loss probability and cell delay). On each link, each traffic class dedicates a separate call admission controller which applies a 'local' fixed strategy, designed to maintain the required QoS, given the buffer space and bandwidth (percentage of cells) assigned to the class on the link. The bandwidth shares are periodically recomputed on-line by a bandwidth allocation controller

* Correspondence to: Dr Obaidat, Professor and Chair, Department of Computer Science, Monmouth University, W. Long Branch, NJ 07764, USA.

CCC 1074–5351/97/050215–10 $17.50

Received April 1997
Revised July 1997

that plays the role of a coordinator in a hierarchical dynamic control scheme.

In this context, in order to check if all nodes to be traversed from source to destination have enough resources to serve an incoming connection, while still guaranteeing QoS to the connections in progress, and possibly optimize the usage of network resources, we have defined a combined routing and admission control scheme. The two proposed routing algorithms are implemented in a hop-by-hop fashion at call set-up time; a call request is forwarded from node to node, each time dynamically choosing the next hop. Upon traversing a node and after checking the availability of the resources for admission, an outgoing link is chosen according to the routing strategy. The request is forwarded to the next node on the path and the same action is repeated. The difference between the two routing algorithms is related to the rejection decision. In the first algorithm, if a node cannot guarantee enough resources for the new call on any outgoing link, the request is refused and the resources already allocated are released. In the second algorithm, if there are no available resources in a node, a new attempt is made to route the new call by looking for an alternative path. In both cases, if the destination node has been reached, the event is notified to the source node, the corresponding virtual circuit (VC) is created, and the resources temporarily assigned along the path are fixed for the duration of the connection. Routing decisions are made on the basis of a distributed strategy, whereby every node decides on the output link which the connection request packet must be sent to. The output link is chosen by minimizing a cost function composed of two terms; a local and global term. The local term takes into account the situation of the node by using a simple function; its value depends on the residual number of connections that every link could accept (maintaining QoS) at the moment the decision is made. The global term holds aggregate information about the situation of the network and it is updated periodically by each node and passed along to its neighbours.

In the next section, we briefly review the structure of the ATM node and the admission control and bandwidth allocation strategies. The overall admission control and routing scheme and the details of the two proposed routing algorithms are described in Section 3. Section 4 presents and discusses the performance simulation results. Section 5 contains the conclusions and directions for further research.

2. ADMISSION CONTROL AND BANDWIDTH ALLOCATION

We assume the traffic on the network to be divided into H classes; each one is characterized by statistical parameters such as peak and average transmission rates, as well as by the QoS requirements such as cell loss probability and cell delay.

The general structure of a node is depicted in Figure 1, where incoming links from other nodes, a local traffic input link and outgoing links (over which the incoming traffic is partitioned) are shown. A resource allocation scheme and a local control access rule of the type presented in Reference 24, are implemented for every outgoing link and their structure, also shown in Figure 1, will be briefly described in this section.

Each traffic class is assigned a separate buffer (service separation),[28] its output is statistically multiplexed on the outgoing link by a schedule which substantially divides the global channel capacity C among classes, according to proportions that last a certain period. Connection requests which can come from the users, directly connected to the node or from other nodes by means of special call request packets, are also processed on a class basis. Two controls are exerted on the system by using a two level hierachical control scheme; one acting on the scheduler and the other on the admission of the connections.

We pose our problem in discrete time, where the duration of a cell transmission corresponds to that of a slot and represents the time unit. At the higher level of the hierarchical control scheme, a bandwidth allocation controller periodically reassigns capacity partitions to every class. The scheduler receives the values of the partitions and must assure that every buffer is assigned a percentage of slots equal to the ratio between the total capacity and the capacity assigned to its class. The new capacity partitions $V_m^{(h)}$, $h=1,2,\ldots,H$, are computed by the controller at discrete time instants, $m=0$, K, $2K$, $\ldots$(where K is the length of the reallocation period in slots), based on the minimization of a cost function that takes into account an evaluation of the expected overall cell loss probability over the next interval. The partitions are computed independently link-by-link by the controller.

At the lower level, H access controllers decide the acceptance of a connection request independently for each class; the maximum number of calls for each class is temporarily fixed, depending on its capacity share at the moment of the decision. The cell loss rate, P_{Loss}, and the delayed cell rate, P_{delay}, for the connection of one class over an outgoing link can be computed as in Reference 23. The acceptance decisions are taken on the basis of the two above mentioned quantities, and, therefore, depend on the capacity currently allocated to the class, the current number of connections in progress, and the statistical and performance characteristics of the specific traffic. As for the latter, class h traffic is assumed to be made up of bursty connections with identical and independent statistical characteristics. The symbols $b^{(h)}$, $B^{(h)}$, and $P^{(h)}$ represent the burstiness (peak bit rate to average bit rate ratio), the average burst length, and the peak bit rate of the h^{th} class. Connected sources are assumed to be of the on-off type, and are modelled simply as a

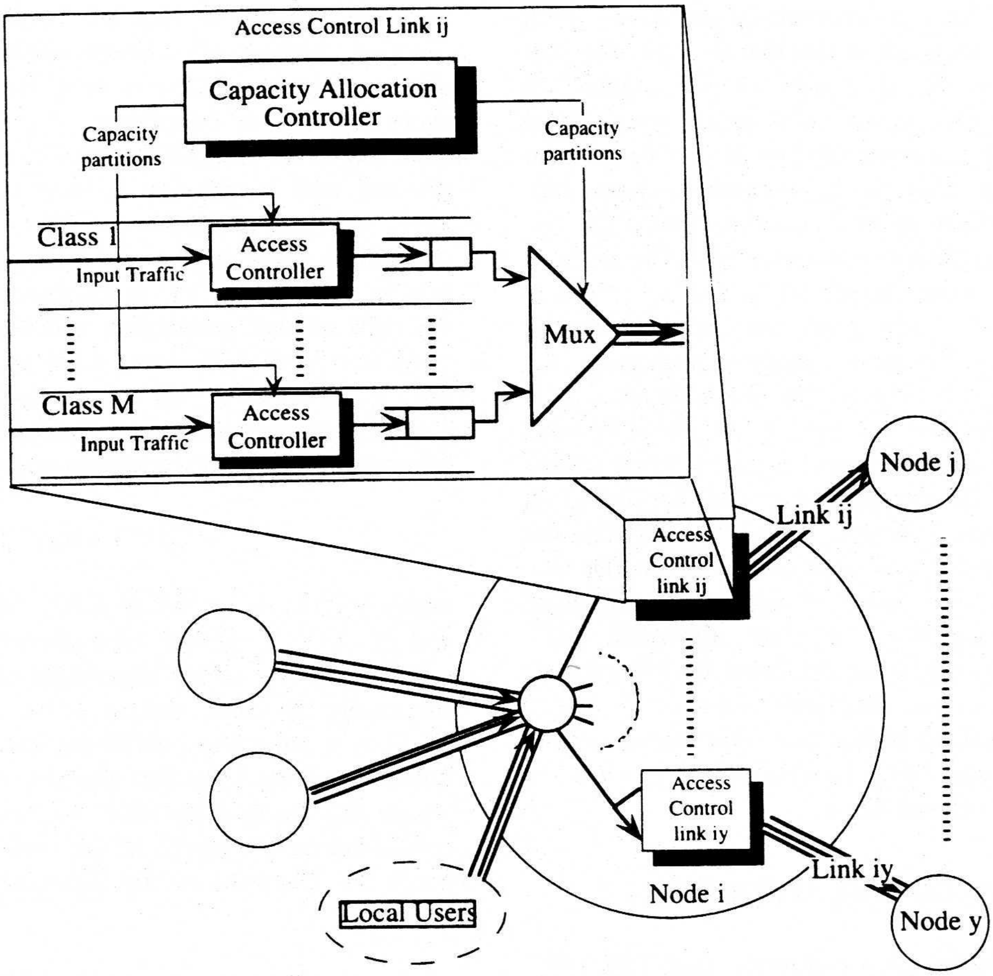


Figure 1. Structure of a node and overall control architecture of a link

two state (idle and active state) Markov chain. When a connection is in the active state, it can generate a cell per slot with a certain probability (equal to $P^{(h)}/C$), it does not generate cells when in the idle state.

After receiving its capacity assignment from the bandwidth allocation controller, each access controller computes the maximum number of connections of the h^{th} class that can be supported on link ij as shown below:

$$N_{ij}^{(h)}(m) = \min \{N_{ij,L}^{(h)}(m); N_{ij,D}^{(h)}(m)\}, \qquad h = 1,\ldots,H \tag{1}$$

where $N_{ij}^{(h)}(m)$ is the maximum number of connections on the link capable of maintaining the cell loss rate below a given upper bound $\epsilon^{(h)}$. $N_{ij,D}^{(h)}(m)$ is the maximum number of connections computed by imposing a similar limit on cell delay, namely, the probability of exceeding a delay of $D^{(h)}$ slots, be less than an upper bound $\delta^{(h)}$ (the details of this computation can be found in Reference 23). Summing up, the local acceptance rule is: a new connection of the h^{th} class arriving at time slot k, $m+\tau \leq k \leq m+\tau+K-1$ (where τ indicates the number of slots required for the computations), can be accepted on link ij if

$$N_{ijA}^{(h)}(k) + 1 \leq N_{ij}^{(h)}(m) \tag{2}$$

where $N_{ij,A}^{(h)}(k)$ is the number of connections of the h^{th} class in progress (i.e. previously accepted on the link and not yet terminated) at time slot k.

The task of the allocation controller is that of balancing the bandwidth distribution among the classes by setting the values $V_m^{(h)}$ at the beginning of a new K-slot interval, on the basis of a performance measure which should be capable of reflecting changes in the offered load of all traffic classes. We measure the offered load [call/s] of a traffic class over a K-slot period as the time average of the total number of service requests in that period, including both accepted and blocked. Based on this measurement, we have chosen to construct a cost function to be minimized with respect to the parameters $V_m^{(h)}$, $h = 1,\ldots,H$, which is an evaluation over the next period of the sum of the steady-state values of the cell loss rates of the various classes that would result from the application of that load. An equality constraint (the sum of the assigned capacities must equal the total capacity of the link) and a set of inequality constraints, which assure service quality for the connections already in progress, must be taken into account in the minimization procedure. The latter can be treated as a mathematical programming problem, which is solved by using a gradient projection method.[22,23]

We can note that all the computations involved, i.e. the calculation of the capacity partitions and of

$N_{ij,L}^{(h)}(m)$, and $N_{ij,D}^{(h)}(m)$, are performed only at the beginning of a decision interval; its duration should be quite long compared to the slot because the cost function reflects the dynamics of the connection requests which take place on a much longer time scale. Thus, if the number of time slots τ to perform the computations after the reallocation instants, represent a small fraction of K, we can avoid the use of special approximations to speed up the computations. On the other hand, the acceptance rule is very simple and it can react very quickly during the K-slot interval to the connection requests.

It is also worth noting that the hierarchical control scheme just described can be implemented, in principle, with any access control rule. In other words, the calculation of Equation (1) can be based on different source models and computational approaches than the one mentioned. Given the link capacity C, if we define a region in call-space ('schedulable region'[21] or 'all allowable VC' region[28]), within which the cell-level QoS is guaranteed, our bandwidth allocation scheme is just a complete partitioning policy over this region, where the partitions can be time-varying and adapt to changes in the offered load.

3. THE ROUTING ALGORITHMS

3.1. *The distributed least congested path (DLCP) routing*

Let us assume for simplicity that a route has already been found; in this case a scheme which will verify the availability of this route is the node-by-node control proposed in Reference 15. A special call request messenger packet is forwarded from node to node, checking whether a connection can be accepted or not. The packet runs through every node of the route, undergoing the access control rule at each hop. Each node must take into account the resources it has allocated. If the call is accepted, the messenger runs further along the route; otherwise, the packet stops, the call is rejected and a message is sent back through the same route. Therefore, the intermediate nodes can free the allocated resources.

Node-by-node admission control in the above fashion and routing, can be performed jointly. In this case, the best route is not chosen beforehand and then verified, as we have assumed above to illustrate the admission control, but, at each node we choose the 'best' outgoing channel among the available ones by means of a cost function associated with each link. The cost function takes into account the link's load situation and the load associated with the subsequent hops to the destination. It is considered only for those links that are not congested (in the sense defined by the admission control rule).

A bit stream in the messenger packet is used to remember the nodes traversed along the route. At each node, a table look-up is performed to find congested channels. The link with the lowest cost function among the non-congested channels is chosen, and the packet is sent along that route. If every channel is congested, or a hardware failure occurred at an intermediate node, the connection is rejected and a message is sent back through the same route, so resources can be relesed. Otherwise, if the messenger reaches its destination, the connection is accepted, a VC is established, and the path the cells of that connection will follow is fixed.

At any instant (in slots), a generic node i chooses the link to which a call request packet generated by a class h connection request should be forwarded, by minimizing (over all successor nodes j) the quantity

$$c_{ij}^{(h)} = c_{ij,L}^{(h)}(k) + \alpha_j c_j^{(h)}(s) \tag{3}$$

where $c_{ij,L}^{(h)}(k)$ is a 'local' cost, related to link ij, and $c_j^{(h)}(s)$ is a 'global' cost (to be defined below), referring to the traffic conditions of node j and its successors at some instant $s < k$ (in slots). $\alpha_j \in [0, 1]$ is a weighing coefficient used to balance the influence of the local and global cost. $c_{ij,L}^{(h)}(k)$ should weigh the local congestion of link ij. A possible choice (already adopted in the simulations in References 24–25 could be the following:

$$c_{ij,L}^{(h)}(k) = \begin{cases} \dfrac{1}{N_{ij}^{(h)}(m) - N_{ij,A}^{(h)}(k)} & \text{if } N_{ij}^{(h)}(m) > N_{ij,A}^{(h)}(k) \\ Z & \text{if } N_{ij}^{(h)}(m) = N_{ij,A}^{(h)}(k) \end{cases} \tag{4}$$

where $m + \tau < k \le m + \tau + K - 1$, and m is the reallocation instant when the access control rule parameters, active at the time of the request packet's arrival, were recomputed (remember that K indicates the length of the reallocation period, i.e. of the interval of validity of a set of bandwidth partitions for each link, and τ is the time required for computation). By using this type of function, the link cost increases with decreasing 'available space' on the link, expressed in the number of acceptable connections. When the link is saturated, the cost value is set to a number Z, which should be large enough to ensure that no saturated link will be chosen if non-congested links are available.

The cost referred to a generic node j ($c_j^{(h)}$(s) in Equation (3)) is composed of two terms as shown below:

$$c_j^{(h)}(s) = c_{j,L}^{(h)}(s) + \beta_j c_{j,A}^{(h)}(s) \tag{5}$$

where β_j is a second weighing coefficient. $c_{j,L}^{(h)}(s)$ represents the average situation of the node with respect to its congestion state, and $c_{j,A}^{(h)}(s)$ is an aggregate information on the average congestion of its successor nodes. More specifically, we have defined:

$$c_{j,L}^{(h)}(s) = 1/L_j \sum_{m \in Succ(j)} c_{jn,L}^{(h)}(s) \quad (6)$$

$$c_{j,A}^{(h)}(s) = 1/L_j \sum_{m \in Succ(j)} c_n^{(h)}(s) \quad (7)$$

Succ(j) being the set of nodes that are successors of node j. As can be seen, the values related to the successor nodes are referred to the instants s, where $s = T, 2T, \ldots$, with T defined as the length of the aggregate cost updating time. It is measured in slots. This means that each node i sends its costs $c_i^{(h)}(s)$, $h = 1, \ldots, H$, to its predecessors every T slots and then after receiving the costs from its successors, recomputes its new aggregate information on the congestion of the network. It is important to note that the updating time is given as a fraction of the reallocation time in the simulation results.

It is worth noting that the distribution of the proposed strategies is based on a mix of local real time (dynamic) and overall delayed aggregate information. Additionally, they do not require the presence of a real time supervisory controller, which would be questionable in a wide area network. Moreover, it can be observed that the characteristics of the strategy mentioned above are substantially independent of the specific access control and bandwidth allocation scheme chosen. This aspect implies that the routing structure can also be applied by using other access control and bandwidth allocation techniques discussed in the literature. It can take into account possible modifications in the cost function. We will refer to the overall routing scheme as distributed least congested path (DLCP).

Finally, it can be observed that performance requirements in our approach are chosen on a node-by-node basis. The problem of guaranteeing performance requirements on an end-to-end basis is not considered in this paper. It is currently under investigation. However, a simple solution can be suggested by adjusting node-by-node requirements on the basis of a fixed maximum path length.

3.2. *The R-DLCP routing scheme*

As already stated, the only difference between the two routing algorithms is the behaviour when a node hasn't enough resources to guarantee performance requirements to the new call. Therefore, for the sake of reducing the redundancy of the presentation, we assume everthing discussed in Section 3.1. still holds, except where the messenger packet arrives at a node whose outgoing links are congested or where a failure occurred. In this second approach the call is not rejected, but the messenger packet is sent back to a selected node and after releasing the resources already allocated from there, it retries from that point. Selecting such a 're-attempt' node is a problem. In this paper, the least loaded node (in terms of the cost function) is chosen. More specifically, the node with the minimal average local cost (that is, the sum of the local costs of all its outgoing links divided by the number of outgoing links) is considered the least loaded one and is chosen.

The strategy which we call re-attempt distributed least congested path (R-DLCP) works as follows. When a source wants to set-up a call, it issues a special packet to reserve the required resources to carry the call. At each node traversed, the local cost is written in the packet header, along with the node identifier. If the messenger packet arrives at its destination, the call is accepted and all the resources involved are held until the connection terminates. Otherwise, if the packet becomes blocked at some node, due either to failure or congestion, it returns to the least loaded node along the established path, releasing resources already reserved upstream and tries again to reach its destination from that point (still respecting the maximum number of hops allowed). If it reaches the destination, the call is established; otherwise, a 'free-resource' packet is sent back to release the resources already reserved and the call is rejected. The re-attempt is allowed only once. This scheme is introduced to enhance the performance of the DLCP in terms of the total number of blocked calls and to achieve a more robust behaviour in case of failure or congestion.

4. PERFORMANCE SIMULATION RESULTS

In this section we discuss the results of several simulation experiments that have been performed on a simple twelve-node test network with a single destination, shown in Figure 2, in order to obtain some indication of the performance of the proposed routing scheme and to compare it with other possible schemes. Three traffic classes ($H = 3$), a 'reallocation interval' $K = 8 \times 10^7$ cells and a transfer capacity $C = 150$ Mbits/s, with a related slot duration $T_s = 2{\cdot}83 \times 10^{-6}$ s (53 bytes/cell), for all channels, have been used. The updating time is given as a fraction of the reallocation time and K/T is dimensionless. The quantities $N_a^{(h)}$ [Erlangs], $h = 1, \ldots, H$, represent the global average traffic intensities offered to the network. The call arrival processes follow independent Poisson distributions. All other parameter values are shown in Table I.

Table I. Parameter values

Traffic class: h	$h = 1$	$h = 2$	$h = 3$
Peak bandwidth: $P^{(h)}$	1 Mbits/s	2 Mbits/s	10 Mbits/s
Burstiness: $b^{(h)}$	2	5	10
Average burst length: $B^{(h)}$	100 cells	500 cells	1000 cells
Average connection duration	20 s	15 s	25 s
Ploss upper bound: $\epsilon^{(h)}$	0·0001	0·0001	0·0001
Pdelay upper bound: $\delta^{(h)}$	0·001	0·001	0·001
Delay constraint: $D^{(h)}$	400 slots	200 slots	100 slots
Buffer length: $Q^{(h)}$	20 cells	15 cells	10 cells

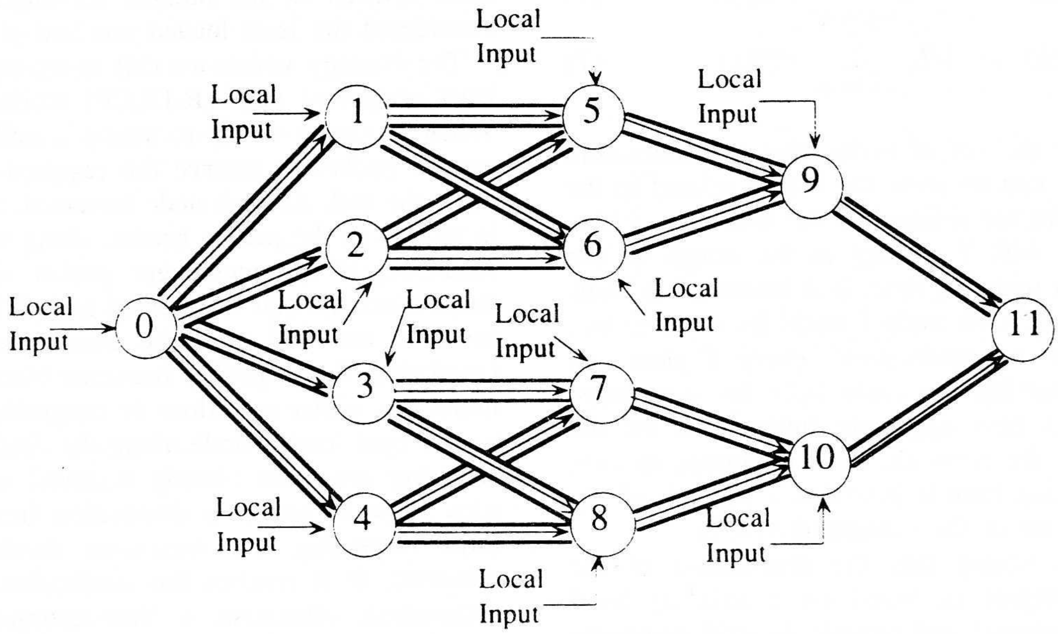


Figure 2. Topology of the first test network

We refer to a 'reference' traffic flow generated by the above data as an offered load 1 when $N_a^{(1)} = 120$; $N_a^{(2)} = 100$; $N_a^{(3)} = 15$, unless stated otherwise. An offered load x corresponds to the same data, except for the traffic intensities $N_a^{(h)}$, $h = 1, 2, 3$, which are multiplied by x. The coefficients α_i and β_i, $i = 0,\ldots,11$, are considered to be the same for each node, i.e. $\alpha_i = \alpha$ and $\beta_i = \beta$, $\forall i$. The behaviour of the access control and bandwidth allocation procedure at a node was tested extensively in References 22 and 23, and some initial simulation results on the performance of the DLCP routing scheme are reported in References 24 and 25.

We will investigate the senstivity of the system to the parameters α and β in order to show the importance of a careful updating procedure and weighing of the global cost and aggregate information. The best results obtained are then compared with two other routing strategies, which may be regarded as two extreme situations with respect to the use of information for routing purposes, namely, a centralized shortest path (SPR) and a totally decentralized 'hot-potato' strategy. The topology of the network used in the simulation is shown in Figure 2; only one node (node 11) is a destination.

In these tests, only the results obtained by using the DLCP routing will be shown. No meaningful difference has been obtained with the R-DLCP because of the particular topology of the network. This topology with two completely separate 'branches', as well as an unbalanced traffic load has been chosen to stress the effect of the weights and emphasize the importance of a careful choice.

The second part of this section is dedicated to the

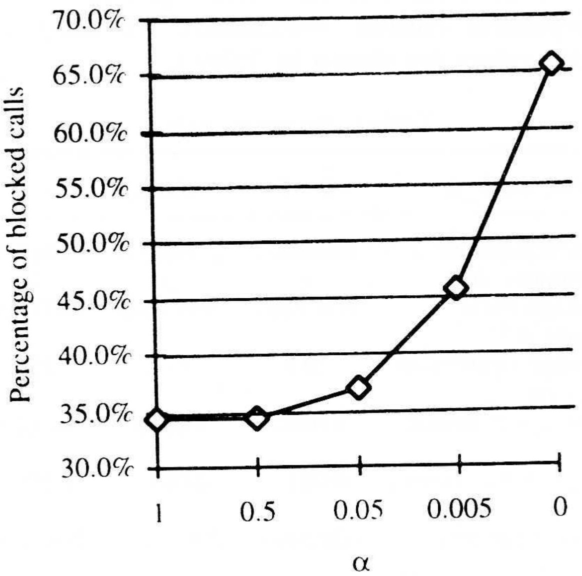


Figure 3. Total percentage of blocked calls versus the weighing coefficient α ($\beta = 1$)

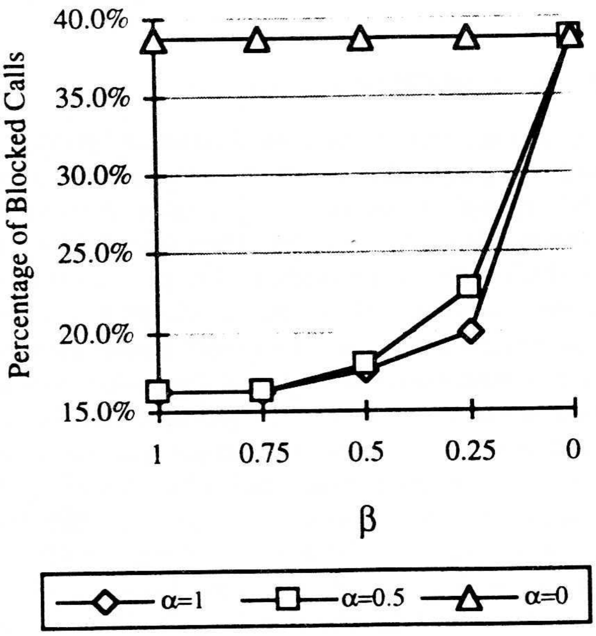


Figure 4. Total percentage of blocked calls versus the weighing coefficient β

comparison between the two strategies introduced in the previous section: DLCP and R-DLCP. The advantages and disadvantages of each strategy are analyzed. The second part of the test is performed by using a different network topology and a different load situation than those in the previous case. In both cases, the criterion for stopping the simulation is that the width of the 95% confidence interval should be less than 3% of the value of the sample average of the quantitiy of interest.

Figure 3 shows the total percentage of blocked connections versus the weighing coefficient α. To stress the effect of weighing the global cost, the bottleneck has been created at nodes 1 and 2, by reducing the channel capacity of each of their outgoing links to 25 Mbits/s, and by dividing the global

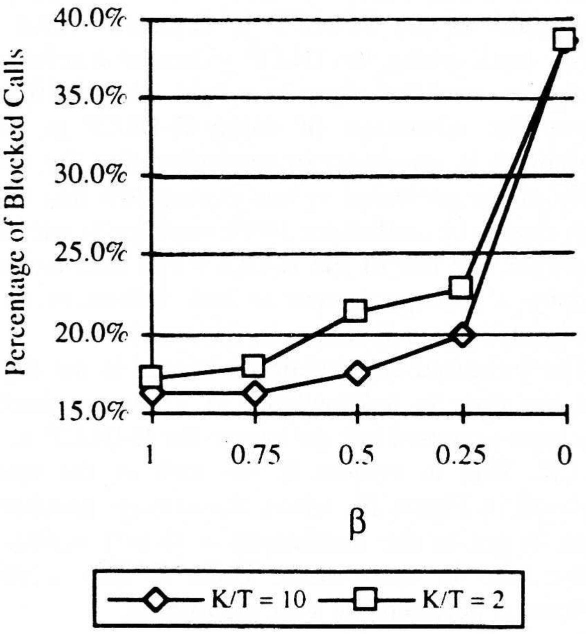


Figure 5. Total percentage of blocked calls versus the weighing coefficient β

offered load as follows: 37·5% to node 0; 31·25% each to nodes 1 and 2; and no load for the other nodes. The effect of the coefficient should be made clear by observing the plot in Figure 3; when α has a value close to 0, the network oversaturates. We can avoid critical nodes (by choosing the other branch) by increasing the value of α. This results in a drastic decrease in the percentage of blocked calls.

A similar comment can be made on Figure 4, which depicts the total percentage of blocked calls versus the coefficient β, for three different values of α. In this case, the bottleneck is moved to node 9, by increasing the local input of this node. The new offered load network configuration is: 37·5% to node 0; and 62·5% to node 9, since the goal is to show the effect of the knowledge of the aggregate information. Note that simulation results presented in the following figures have been performed by using this new load configuration. It is important to note the increase in blocked calls caused by decreasing the weight β, except for $\alpha = 0$, where, of course, variations in β have no effect.

In Figure 5, the percentage of blocked connections versus the weighing coefficient β, is shown in two different cases of the aggregate information's updating time T, namely ten ($K/T = 10$) and two ($K/T = 2$) times every bandwidth reallocation interval for $\alpha = 1$. The improvement obtained by more frequently updating is clear, except at $\beta = 0$, where the aggregate information is ignored.

Figure 6 depicts the total percentage of blocked calls versus the offered load. By setting $\alpha = 1$ and $\beta = 1$, which are the best values in this case, the (DLCP), the centralized (SPR) and the local hot-potato algorithms are compared. As shown in Figure 6, the percentage of blocked calls for DLCP is quite close to that of SPR.

The effect can be better appreciated from

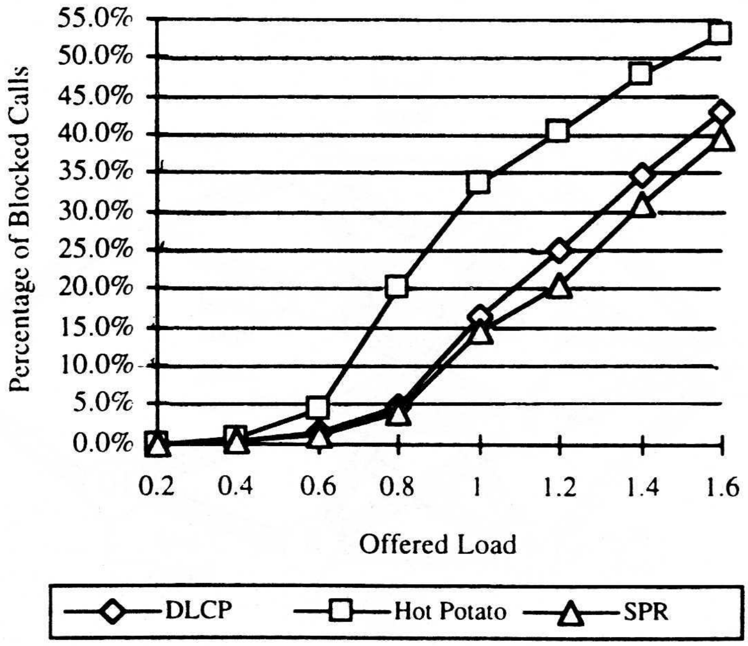


Figure 6. Total percentage of blocked calls versus the offered load

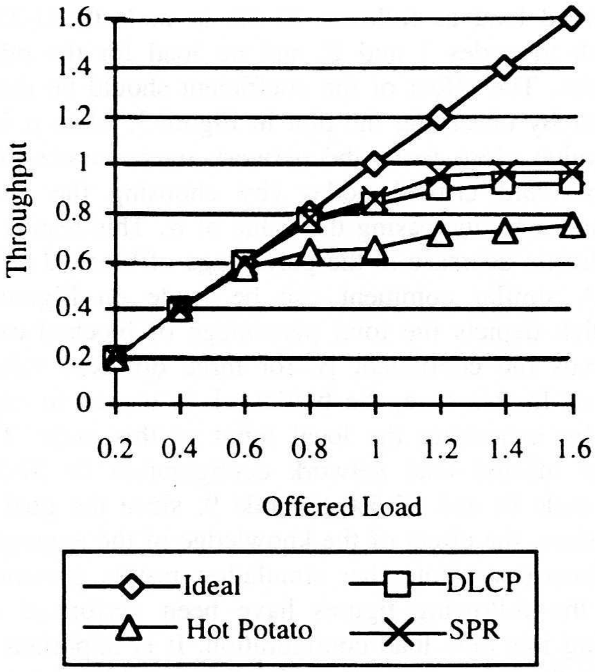


Figure 7. Total network throughput versus the offered load

Figure 7, where the network throughput (expressed in terms of the reference load) is plotted as a function of the offered load, showing an increase close to the ideal throughput characteristic up to the value of 0·8 and a behaviour very similar to the SPR.

Next we will compare the performance of the DLCP and the R-DLCP routing strategies. The topology in Figure 8 has been used, where the presence of the four addititional links (1, 8), (4, 5) (2, 7) and (3, 6) allows a greater flexibility in the alternative choices in case of re-attempt. The traffic is composed of calls of the same traffic classes as before but, now, each node (0 to 10) can be a source, except node 11 which is the only destination.

The simulation has been carried out with $K/T = 2$. The values of the weighing coefficients are ($\alpha = 1$, $\beta = 1$). These values are not the best values for this topology. A simulation analysis, the results of which have not been shown here, has verified that small values of α and β provide the best results. In fact, because the average number of hops required by this network to reach the destination is relatively short, the importance of the aggregate cost is minimized and the network topology greatly enhances the importance of the 'local' part of the cost.

However, the use of higher values of α and β is reasonable if the network in Figure 8 is considered to be a subnet of a larger one, where the choice of 'optimal' values of the weighing coefficients for each node would be too difficult to manage, even if it is theoretically possible (the general framework in Section 3.1. considered different coefficients α_i and β_i, for each node).

In Figure 9, a comparison between DLCP and R-DLCP is shown versus the traffic load. In Figure 10, the 'gain' of the R-DLCP is depicted versus the traffic load, taking the DLCP values as a reference. It can be seen that there is a load interval (0·8–1) where the advantage of using R-DLCP is more evident. It is important to remember that the variability of the presented values is very low due to the high degree of confidence (95% confidence interval). However, the use of the R-DLCP can also be justified by a prompt answer to link failure or to an unexpected steep increase in the traffic.

The increased computational burden is not higher and the time to acknowledge if a new connection has been accepted not so longer for R-DLCP as for DLCP. This is evident if we look at the results depicted in Figure 11, where the average number of hops to get to the destination is shown versus the traffic load in both cases. There is only a slight difference between the two algorithms.

5. CONCLUSIONS

To conclude, a global control architecture for access control, bandwidth allocation and routing, has been

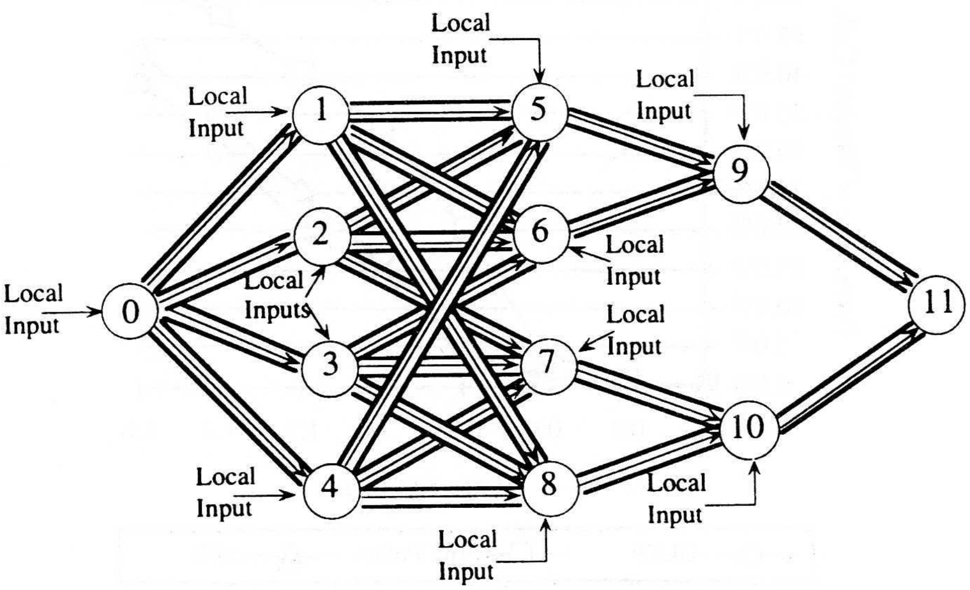


Figure 8. Topology of the second test network

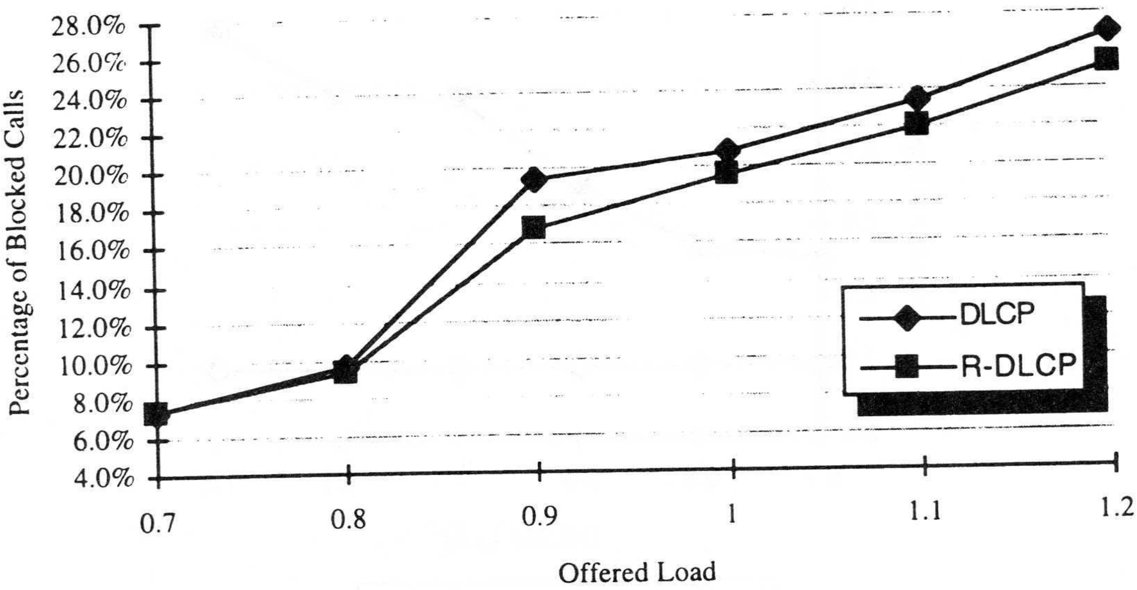


Figure 9. Total percentage of blocked calls versus the offered load (DLCP, R-DLCP)

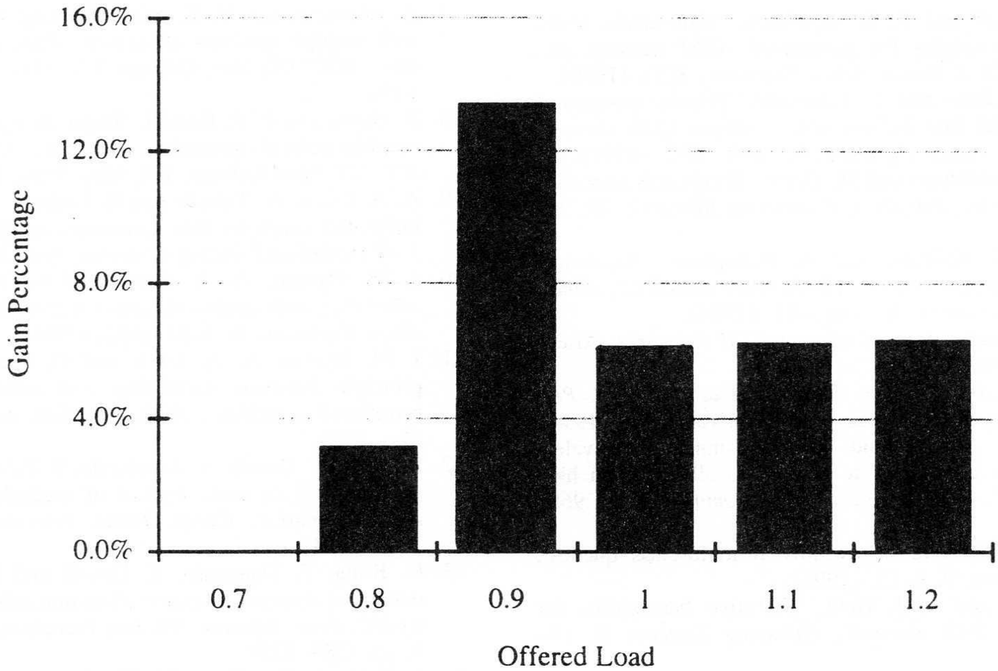


Figure 10. R-DLCP gain percentage in the total blocked calls with respect to DLCP versus the offered load

devised for an ATM network environment. The traffic is organized into classes, characterized by specific performance requirements. Bandwidth partitions are dynamically allocated among them by controllers assigned specifically to each link. Access control and routing are performed separately for each class on a hop-by-hop basis. The former is exerted by local controllers for each link, whereas the latter stems from a distributed procedure based on local (real time) as well as aggregated (delayed) information. Two routing strategies have been explicitly defined in this paper. Performance simulation results have been reported and compared with other routing strategies like local hot potato and centralized (SPR). The results highlight a low call rejection rate as an overall effect of the control structure over a large range of network load values. The routing strategies defined in this paper can be further investigated in order to assess the effect of different re-attempt mechanisms, consider their application to two-hop alternative routing in fully connected networks, and analyze convergence properties. These topics are the subject of current research.

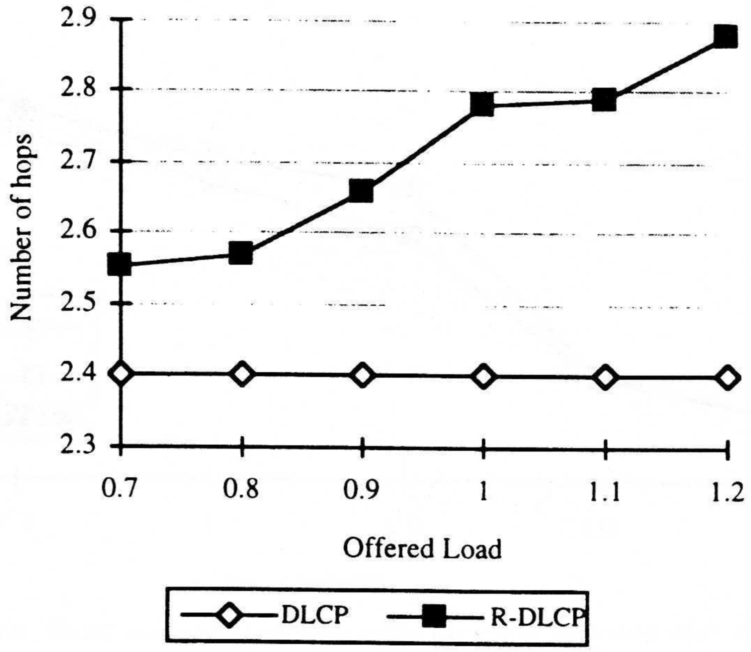
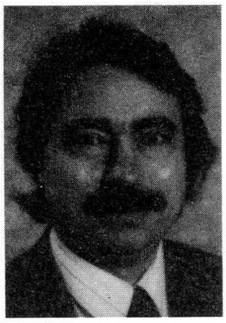


Figure 11. Number of hops to get the destination versus the offered load (DLCP, R-DLCP)

REFERENCES

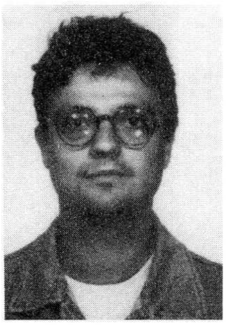
1. G. M. Woodruff and R. Kositpaiboon, 'Multimedia traffic management principles for guaranteed ATM network performance', *IEEE J. Select. Areas Commun.*, **8**(3), (1990).
2. Y. Takagi, S. Hino and T. Takahashi, 'Priority assignment control of ATM line buffers with multiple QOS classes', *IEEE J. Select. Areas Commun.*, **9**, 1078–1092, (1991).
3. J. A. Suruagy Monteiro and M. Gerla, 'Bandwidth allocation in ATM networks', *Annals of Operations Research*, **49**, 25–50, (1994).
4. L. Gun, V. G. Kulkarni and A. Narayanan, 'Bandwidth allocation and access control in high speed networks', *Annals of Operation Research*, **49**, 161–183, (1994).
5. H. Saito, *Teletraffic Technologies in ATM Networks*, Artech House, Inc., Norwood, MA, 1994.
6. R. O. Onvural, *Asynchronous Transfer Mode Networks: Performance Issues*, Artech House, Inc., Norwood, MA, 1994.
7. R. Guerin, H. Ahmadi and M. Naghshineh, 'Equivalent capacity and its application to bandwidth allocation in high speed networks', *IEEE J. Select. Areas Commun.*, **9**(7), 968–981, (1991).
8. K. P. Kelly, 'Effective bandwidths at multi-class queues', *Queueing Systems*, **9**, p. 15, (1991).
9. R. J. Gibbens and P. J. Hunt, 'Effective bandwidths for the multi-type UAS channel', *Queueing Systems*, **9**, 17–27, (1991).
10. T. Kamitake and T. Suda, 'Evaluation of an admission control scheme for an ATM network considering fluctuations in cell loss rate', *Proc. IEEE Globecom'89*, Dallas, TX, November 1989, pp. 1774–1780.
11. A. Lombardo, S. Palazzo and D. Panno, 'Admission control over mixed traffic in ATM networks', *Int. J. Digital and Analog Commun. Syst.*, **3**, (August 1990).
12. Z. Dziong, J. Choquette, K. Q. Liao and L. Mason, 'Admission control and routing in ATM networks', *Comput. Networks and ISDN Syst.*, **20**, 189–196, (1990).
13. C. Rasmussen, J. H. Sorensen, K. S. Kvols and S. B. Jacobsen, 'Source independent call acceptance procedures in ATM networks', *IEEE J. Select. Areas Commun.*, **9**(4), 351–358, (1991).
14. T. Murase, H. Suzuki, S. Sato and T. Takeuchi, 'A call admission control scheme for ATM networks using a simple quality estimate', *IEEE J. Select. Areas Commun.*, **9**, 1461–1470, (1991).
15. D. Hong and T. Suda, 'Congestion control and prevention in ATM networks', *IEEE Network Mag.*, **5**(4), 10–16, (1991).
16. S. Gupta, K. W. Ross and M. El Zarki, 'Routing in virtual path-based ATM networks', *Proc IEEE Globecom'92*, Orlando, FL, December 1992, pp. 571–575.
17. A. Maunder and P. S. Min, 'Routing for multirate traffic with multiple qualities of service', *Proc. Int. Conf. on Commun., ICC '94*, New Orleans, LA, May 1994, **1**, pp. 1390–1394.
18. S. Gupta and P. P. Gandhi, 'Dynamic routing in multi-class non-hierarchical networks', *IEEE Int. Conf. on Commun., ICC '94*, New Orleans, LA, May 1994, 1, pp. 1390–1394.
19. A. A. Lazar, A. Temple and R. Gidron, 'An architecture for integrated networks that guarantees quality of service', *Int. J. of Digital and Analog Commun. Syst.*, **3**, 299–238, (1990).
20. J. M. Hyman, A. A. Lazar and G. Pacific, 'Real time scheduling with quality of service constraints', *IEEE J. Select. Areas Commun.*, **9**, 1052–1063, (1991).
21. J. M. Hyman, A. A. Lazar and G. Pacific, 'A separation principle between scheduling and admission control for broadband switching', *IEEE J. Select. Areas Commun.*, **11**, May 1993.
22. R. Bolla, F. Davoli, A. Lombardo, S. Palazzo and D. Panno, 'Hierarchical dynamic control of multiple traffic classes in ATM networks', *Europ. Trans. Telecommun.*, **5**(6), 747–755, (1994).
23. R. Bolla, F. Danovaro, F. Davoli and M. Marchese 'An integrated dynamic resource allocation scheme for ATM networks', *Proc. Infocom '93*, San Francisco, CA, March 1993, **3**, pp. 1289–1297.
24. R. Bolla, F. Davoli and M. Marchese, 'A distributed routing and access control scheme for ATM networks', *Proc. IEEE Int. Conf. on Commun., ICC '94*, New Orleans, LA, May 1994, **1**, pp. 44–50.
25. R. Bolla, F. Davoli and M. Marchese, 'Performance of hop-by-hop distributed routing and resource allocation in an ATM network', *Proc. ICCCN'94*, San Francisco, CA, September 1994, pp. 230–234.
26. M. Aicardi, R. Bolla, F. Davoli and R. Minciardi, 'A parametric optimization approach to admission control and bandwidth assignment in hybrid TDM networks', *Int. J. of Digital and Analog Commun. Syst.*, **6**, 15–27, (1993).
27. R. Bolla and F. Davoli, 'Dynamic hierarchical control of resource allocation in an integrated services broadband network', *Computer Networks and ISDN Syst.*, **25**(10), 1079–1087, (1993).
28. K. W. Ross, *Multiservice Loss Models for Broadband Telecommunication Networks*, Springer-Verlag, London, 1995.

Authors' biographies:

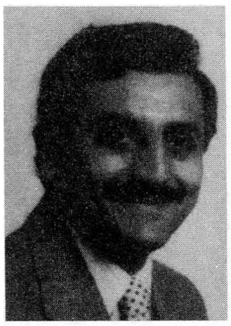


Raffaele Bolla was born in Savona, Italy, in 1963. He received the 'laurea' degree in electronic engineering from the University of Genoa, Italy, in 1989 and the PhD degree in telecommunications at the Department of Communications, Computer and Systems Science (DIST), University of Genoa, in 1994. From 1994 to 1996 he was a post-doc research fellow at DIST. Since November 1996 he has been Assistant Professor in the same department. His current research interests are in management and control of STM and ATM networks, multimedia communications and multiple access in integrated mobile radio networks. He is a member of IEEE.

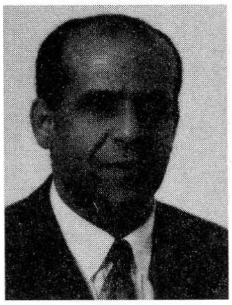
Ahmad Dalal'ah was born in Beit Yafa, Jordan, in 1957. He got his BSc in computer science from Yarmouk University, Jordan, in 1985. From 1986 to 1988 he worked as a high school teacher in Irbid, Jordan. From 1989 to 1993 he was a teaching assistant at the Computer Department of Mu'tah University, Jordan, and since 1994 he has been studying for his PhD degree at the Department of Communications, Computer and Systems Science (DIST), University of Genoa, Italy. His research interests include network planning, traffic modelling, admission control and routing in ATM networks. He is a member of IEEE.



Franco Davoli was born in Genoa, Italy, in 1949. He received the 'laurea' degree in electronic engineering from the University of Genoa, Italy, in 1975. In 1985 he became an Associate Professor and in 1990 a Full Professor of Telecommunication Networks at the University of Genoa, where he is with the Department of Communications, Computer and Systems Science (DIST). From 1989 to 1991 and from 1994 to 1996, he was also with the University of Parma, Italy, where he taught a class in Telecommunication Networks, by means of interactive distance learning over ISDN. His current research interests are in bandwidth allocation, admission control and routing in STM and ATM networks, multimedia communications and services, and integrated services mobile radio networks. He is a member of IEEE.



Mario Marchese was born in Genoa, Italy, in 1967. He received the 'laurea' degree in electronic engineering from the University of Genoa, Italy, in 1992 and his PhD in telecommunications at the Department of Communications, Computer and System Science (DIST), University of Genoa, in 1997. His research interests include traffic modelling, admission control and routing in ATM-based networks. He is a member of IEEE.



Mohammad S. Obaidat is a Professor and Chair of the Department of Computer Science at Monmouth University, New Jersey. He received his MSEE and PhD in electrical/computer engineering from The Ohio State University, Columbus, Ohio. He has received extensive research funding and has published over 120 refereed technical articles. Dr Obaidat is an associate editor/editorial board member of six refereed scholarly journals including two *IEEE Transactions*. He has guest edited three special issues of the *Simulation* journal and is currently guest editing a special issue of *IEEE Transactions on Systems, Man and Cybernetics*, SMC, on Neural Network Applications. Obaidat has served as the Steering Committee Chair and the Programme Chair of the 1995 IEEE Int. Conference on Electronics, Circuits and Systems, ICECS'95, and as the Programme Chair of the 1996 IEEE International Phoenix Conference on Computers and Communications, IPCCC'96. He is the Vice General Chair of the 1997 IEEE Int. Performance, Computing and Communications Conference and the General and Programme Chair of the 1997 SCS Summer Computer Simulation Conference, SCSC'97. He is also the Vice Programme Chair of the 1997 IEEE International Conference on Computer Communications and Networks. Dr Obaidat will be the General Co-Chair of the 1998 IEEE International Performance, Computing and Communications Conference. He will also be the General Chair of the 1998 Summer Computer Simulation Conference. He has served on the programme, organization and steering committees of many IEEE, ACM and SCS international conferences. Professor Obaidat is a member of the Board of Directors of the Society for Computer Simulation International. Obaidat has received a recognition certificate from IEEE. He is listed in several Who's Who including, *Who's Who in the World* (13th and 14th edn), *Who's Who in Science and Engineering* (3rd edn), *Who's Who in America* (51st edn), *Dictionary of International Biography* (24th edn), *Men of Achievements* (17th edn), and *5,000 Personalities of the World* (5th edn). Obaidat is a distinguished speaker of IEEE, a distinguished lecturer of ACM, and a member of the Computing Sciences Accreditation Board/Commission (CSAB/CSAC). His research interests are; performance evaluation of computer systems, algorithms and networks, high performance and parallel computing/computers, applied neural networks and pattern recognition, computer and telecommunication networking and speech processing. He is a senior member of IEEE and SCS and a member of ACM.