Distributed Dynamic Routing of Virtual Circuits in ATM Networks under Different Admission Control and Bandwidth Allocation Policies

Raffaele Bolla, Franco Davoli, Mario Marchese

Department of Communications, Computer and Systems Science (DIST)
University of Genoa
Via Opera Pia 13, I-16145 Genova, Italy
e-mail: (lelus, franco, dama)@dist.unige.it

Abstract.

The routing problem for ATM Virtual Circuits at call set-up level is considered, by combining Call Admission Control (CAC) and routing decisions in a hop-by-hop fashion at the nodes along a route. Service separation is assumed, and the overall traffic flows are divided into classes, homogeneous with respect to performance requirements and statistical characteristics; the bandwidth of each link in the network is shared among the classes according to some specified policy that also determines the CAC rule. At each node traversed, a call request of a certain class is first assigned a subset of outgoing links towards the destination, upon which the resources necessary to maintain the required level of Quality of Service are available; then, a routing decision is taken, by choosing the least "cost" link, according to a specific criterion. The link costs are dynamically updated for all traffic classes, by means of local information and of aggregate information exchanged among neighbouring nodes, and are based on a measure of the link saturation, in terms of some "distance" from the boundary of the load region in call-space that the multiplexer serving the link can support. Five different strategies are described and compared, which essentially differ in terms of the way the link bandwidth is allocated among the traffic classes (and, accordingly, the CAC operation is performed).

1. Introduction

With the progress in standardization and the increasing deployment of ATM switching nodes, the issue of routing in ATM networks has found its place among the various controls that are necessary to maintain Quality of Service (QoS) requirements to the different traffic types, while ensuring at the same time a high utilization of network resources (like, e.g., Call Admission Control (CAC), policing, shaping, etc.). Actually, the routing problem in ATM cannot be considered as separated from the other controls, but it should be rather integrated with them. Moreover, the particular nature of ATM, where the dynamic structure of statistical multiplexing must be combined with resource allocation techniques like the above mentioned ones, allows to draw suggestions from the vast previous experience in routing from both circuit- and packet-switched networks (see, e.g., [1, 2, 3]).

Among the various control approaches (especially as regards CAC), some impose a certain structure on the allocation of the resources, where typically traffic is subdivided into classes, which are homogeneous in terms of statistical or performance characteristics, and bandwidth is allocated accordingly, thereby restricting the statistical multiplexing within each class; this does not sensibly degrade performance in case of services with largely different QoS or statistical nature (see [3, p. 141] and references therein). This approach (service separation) has been adopted in various forms in a number of works in the literature [4-15]. This often also eases the decomposition of a very complex overall control task, which is in general characterized by very different time scales and requirements, according to the level where the system dynamics is 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 [6, 7] through the concept of schedulable region, whereas a hierarchical decomposition has been used by the authors in previous works [11]. The routing problem in ATM has been considered in several contexts (see, e.g., [14-22]), both with and without service and/or route separation [3], over either Virtual Path (VP) or Virtual Circuit (VC) connections.

In this paper, we consider the routing problem for ATM VCs at call set-up level, by combining CAC and routing decisions in a hop-by-hop fashion at the nodes along a route. In doing this, we assume service separation, by dividing the overall traffic flows into classes, homogeneous with respect to performance requirements and statistical characteristics; the bandwidth of each link in the network is shared among the classes according to some specified policy that also determines the CAC rule. At each node traversed, a call request of a certain class is first assigned a subset of outgoing links towards the destination, upon which the resources necessary to maintain the required level of QoS are available; then, a routing decision is taken, by choosing the least "cost" link, according to a specific criterion. The link costs are dynamically updated for all traffic classes, by means of local information and of aggregate information exchanged among neighbouring nodes, and are based on a measure of the link "saturation", in terms of some "distance" from the boundary of the load region in call-space that the multiplexer serving the link can support. More specifically, a cost is attached to each link, for each traffic class, which is composed of two terms: i) a "local" one that accounts for the current situation of the link at the decision instant; ii) a "global" term, which condenses aggregate information about the situation of the network, by referring to the "average" traffic conditions of the node at the other end of the link and its neighbours. This aggregate information is acquired through an exchange among the nodes and, as a consequence, it represents a sort of delayed indicator of some downstream congestion.

Within this general scheme, which was introduced and referred to as DLCP (Distributed Least Congested Path) in previous works by the authors [14, 15], we distinguish and compare several strategies, which essentially differ in terms of the way the link bandwidth is allocated among the traffic classes (and, accordingly, the CAC operation is performed). In all cases, some pre-defined maximum number of acceptable calls for each traffic class forms the basis for the calculation of the link saturation measure, which is used in computing the link cost. Moreover, all methods can be interpreted in terms of the region in call-space within which QoS constraints at the cell level are satisfied (referred to as "feasibility region" in the following), evaluated according to a specific technique [13].

The first method, named Feasibility Region Scheme (FRS), simply accepts an incoming call, as long as the system state (the vector of accepted connections for each traffic class) is within the feasibility region (Dynamic Partitions [3]). The inverse of some measure of the "distance" between the current "point" in call-space and the boundary of the region is used in this case; routing is still performed on a per-class basis.

The second method, named Dynamic Reallocation Scheme (DRS), is based on a Complete Partitioning strategy [3] within the feasibility region. Two controls are applied, in a two-level hierarchical structure, one of them acting on the scheduler that serves the link buffer, and the other on the admission of connection requests. At the higher level of the hierarchy, a bandwidth allocation controller periodically reassigns (by minimizing a suitable cost function) bandwidth partitions to the traffic classes, whose sum amounts to the total link bandwidth (the partitions may be interpreted as Virtual Paths, one for each traffic class). The scheduler receives the values of the partitions and must ensure that each traffic class is assigned the necessary slots accordingly. At the lower level, independent access controllers for each class decide upon the acceptance of incoming connection requests, by computing the maximum number of calls that the class can support, given the assigned bandwidth. In this case, the local cost per class used for routing is the inverse of the difference between the maximum and the current number of accepted connections of that class on the link.

The two techniques above will be compared, by simulation, between them and with the routing policies arising from some other acceptance methods (defined and analyzed, as such, in [12, 13]). More specifically, the Erlang Scheme (ES), the Balanced Erlang Scheme (BES), and the Constrained Erlang Scheme (CES) [12, 13] form the basis of our third, fourth and fifth combined routing and acceptance control methods, respectively. They are also Complete Partitioning strategies, where the maximum acceptable number of calls for each class is computed either from the minimization of a weighted sum of blocking probabilities (ES), or from the minimization of the maximum (over the classes) blocking probability (BES), or by enforcing (as close as possible) a set of constraints on the per-class blocking probabilities (CES).

The paper is organized as follows. In the next Section, we define the above mentioned bandwidth allocation and CAC schemes. The combination of the latter with the distributed routing strategy is described in Section 3. Section 4 reports and discusses simulation results that compare the various DLCP routing methods stemming from the different resource allocation strategies; a comparison with hot-potato and centralized shortest path routing (based on the same metrics as DLCP) will also be provided. Section 5 contains the conclusions .

2. Service Separation and Complete Partitioning CAC and bandwidth allocation strategies

We suppose the traffic on the network to be divided into H classes of bursty (on-off) sources, each class being characterized by statistical parameters like peak and average transmission rate and average burst length, as well as by QoS requirements, e.g., on cell loss probability and cell delay. We indicate with $b^{(h)}$, $B^{(h)}$ [slots] and $P^{(h)}$ [bits/s], the burstiness (peak to average rate ratio), the average burst length, and the peak bit rate, respectively, of the h-th class. Moreover, let $\lambda^{(h)}$ and $1/\mu^{(h)}$ represent the average arrival rate and the average duration of connections of class h, respectively, and $\rho^{(h)} = \lambda^{(h)}/\mu^{(h)}$.

At each link ij within each network node, traffic class h is assigned a separate buffer of length $Q^{(h)}$ [cells], whose output is statistically multiplexed on the outgoing link by a scheduler, which substantially divides the global channel capacity C_{ij} [bits/s] into "virtual" partitions $V_{ij}^{(h)}$ among the classes, whose sum amounts to C_{ij}. Connection requests, which can come from the users directly connected to the node or from other nodes, are also processed on a per-class basis. Given a model for the traffic sources of a class, the cell-level performance requirements (e.g., in terms of average cell loss and delayed cell rate) allow to define a region in call-space (which, as we mentioned in the introduction, will be referred to as "Feasibility Region" (FR)), where QoS requirements are surely satisfied. Accepting connections all over the FR corresponds to the CAC method named "Service Separation with Dynamic Partitions" in [3, p. 147]. One such region is associated with each link ij in the network.

The computation of the FR has been the object of several studies. In any case, it is worth noting that, in the context of the problem we are considering in this paper, the FR should be just a tool to describe the CAC schemes. The specific technique to ensure QoS satisfaction at the cell-level could be changed (always within the framework of Service Separation), without affecting the access control general procedure. However, to fix ideas and to provide a basis for the numerical results, we refer here to a specific method, already used in previous works [11-15]; a comparison of the method with a well-known approach based on equivalent bandwidth [23] has been performed in [13].

Let $\square^{(h)}$ and $\square^{(h)}$ be upper limits on the average cell loss rate $P_{loss}^{(h)}$ and on the average rate $P_{delay}^{(h)}$ of cells suffering a delay longer than a fixed value ($D^{(h)}$ in Section 4), respectively. The quantities $P_{loss}^{(h)}$ and $P_{delay}^{(h)}$ are computed as follows (see [11] and [24] for details): i) an Interrupted Bernoulli Process (IBP) [25] is used to model each single class-h bursty source; ii) conditional values of $P_{loss}^{(h)}$ and $P_{delay}^{(h)}$ are found, for each fixed number n of active (generating cells) connections out of $N^{(h)}$ accepted ones; iii) the conditional values are averaged with respect to the binomial distribution of the active connections. The set of (integer) H-tuples $\mathbf{N}_{ij,FR} = \mathrm{col}\left[N_{ij,FR}^{(h)},\ h = 1,\ldots,H\right]$ that satisfy the limits with equality forms the boundary of the FR.

We let

$$\mathbf{N}_{ij,A}(k) = \mathrm{col}\left[N_{ij,A}^{(h)}(k),\ h = 1,\ldots,H\right] \tag{1}$$

where $N_{ij,A}^{(h)}(k)$ is the number of connections in progress at the generic instant k for traffic class h and link ij. The vector in (1) represents the state of the system at instant k (the VC-profile in [3]).

By using the feasibility region, together with call-level QoS requirements, we can introduce the five CAC and bandwidth allocation methods that we mentioned in the introduction. Here we only touch the main aspects that will be necessary for inclusion in the routing scheme; more detailed descriptions and comparisons can be found in [12, 13].

It is worth noting at this point that the Complete Partitioning (CP) schemes that will be described in the following fix the bandwidths $V_{ij}^{(h)}$ to be allocated , with

$$\sum_{h=1}^{H} V_{ij}^{(h)} = C_{ij} \tag{2}$$

and the corresponding maximum number $N^{(h)}_{ij,max}$ of connections for each class (on the boundary of the FR), thus defining a "hyper-rectangular" sub-region inside the feasibility region, within which the VC-profile shall be maintained by the CAC rule.

Feasibility Region Scheme (FRS)

The CAC in this case is performed, as already mentioned, by accepting a call whenever the system's state remains within the feasibility region after acceptance. This is obviously not a partitioning scheme.

Dynamic Reallocation Scheme (DRS)

The DRS is a CP scheme, extensively explained in [11, 24]: two controls are exerted on the system, by using a two-level hierarchical control scheme, one acting on the scheduler and the other on the admission of the connections. At the higher level, a bandwidth allocation controller periodically reassigns capacity partitions to every class. The scheduler receives the values of the partitions and must assure that each 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 $V^{(h)}_{ij}(m)$, $h = 1,\dots,H$, are computed by the controller at discrete time instants m = 0, K, 2K, ... (K is the length of the intervention period in slots), through the minimization of a cost function, whose value depends on the overall cell loss rate that would be generated by the total offered load (which is estimated by considering also the number of calls that were blocked in the previous period). "Dynamic" constraints on the minimum amount of bandwidth needed to guarantee QoS to the ongoing connections are taken into account. We may note, in passing, that this method requires no assumption on the incoming call statistics.

At the lower level, H access controllers decide about the acceptance of a connection request, independently for each class. The acceptance decisions are taken on the basis of $P^{(h)}_{loss}$ and $P^{(h)}_{delay}$, and therefore depend on the capacity currently allocated to the class, on the current number of connections in progress, and on the statistical and performance characteristics of the specific traffic. After each access controller receives its capacity assignment from the bandwidth allocation controller, it computes the maximum number $N^{(h)}_{ij,DRS}(m)$ of connections of the h-th class that can be supported on link ij. So, in this scheme, the maximum number of acceptable connections of class h on link ij at a generic slot k ($N^{(h)}_{ij,max}(k)$) is

$$N^{(h)}_{ij,max}(k) = N^{(h)}_{ij,DRS}(m) \tag{3}$$

where $m + \square \square k \square m + \square + K \square 1$ (where $\square$ indicates the number of slots required for the computations).

The DRS scheme can be interpreted as an adaptive CP mechanism, where the reallocation operation gives rise to different CP "rectangular" regions, with a vertex moving on the boundary of the FR to adapt to load variations (see Fig. 1, where an example is reported). The vector $\mathbf{N}_{ij,DRS}(m) = \mathrm{col}\left[N^{(h)}_{ij,DRS}(m),\ h = 1,\dots,H\right]$ has been introduced to simplify the notation.

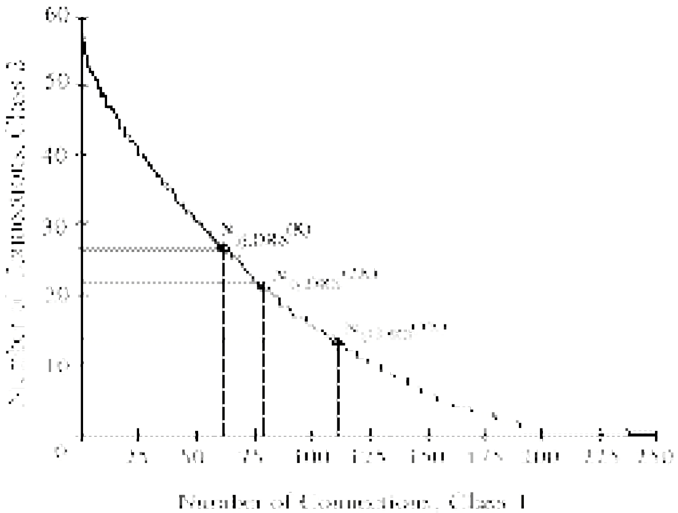

Figure 1. DRS acceptance scheme in three different reallocation instants.

Erlang Scheme (ES)

In the ES, the maximum numbers of acceptable calls $N_{ij,max}^{(h)}$, h=1,...,H (we do not need, for the time being, the argument k here, because of the absence of reallocation) are computed through the minimization, over the FR, of a weighted sum of the per-class blocking probabilities, namely

$$P_{B,ij}(\mathbf{N}) = \sum_{h=1}^{H} \alpha_{ij}^{(h)} P_{B,ij}^{(h)}\left(N^{(h)}\right) \tag{4}$$

where the parameters $\alpha_{ij}^{(h)}$ are weighting coefficients. We let the corresponding vector be $\mathbf{N}_{ij,ES}$. The individual blocking probabilities $P_{B,ij}^{(h)}(N^{(h)})$ are simply given by the Erlang B formula [3], due to the CP assumption (here and in the following, use is made of the statistical characteristics (birth-death type) of the call process and the knowledge of the traffic intensity is assumed; blocking probabilities are evaluated over an infinite time horizon, i.e., by using stationary distributions). If $\alpha_{ij}^{(h)} = \lambda_{ij}^{(h)} \Big/ \sum_{k=1}^{H} \lambda_{ij}^{(k)}$, (4) represents the average blocking probability of link ij.

Balanced Erlang Scheme (BES)

Here, the main goal is the achievement of the equalization of the blocking probabilities over the classes that traverse link ij. Therefore we look for the vector $\mathbf{N}_{ij,BES}$ that minimizes

$$P'_{B,ij}(\mathbf{N}) = \max_{h}\left\{\beta_{ij}^{(h)} P_{B,ij}^{(h)}\left(N^{(h)}\right)\right\} \tag{5}$$

over the FR. $\beta_{ij}^{(h)}$ are weighting coefficients.

Constrained Erlang Scheme (CES)

This scheme is defined by adding a constraint on the maximum call blocking probability for each class:

$$P_{B,ij}^{(h)}\left(N_{ij}^{(h)}\right) \le \gamma_{ij}^{(h)}, \quad h=1,2,..., H \tag{6}$$

If the points $\mathbf{N}_{ij}$ that satisfy the constraints are inside the FR, then $\mathbf{N}_{ij,CES}$ is found by minimizing (4) over a subset of the boundary points of the FR that satisfy (6); otherwise, a quadratic deviation of the blocking probabilities from the upper bounds in (6) is minimized over the boundary (the details of this method can be found in [12]).

This completes the presentation of the CAC and bandwidth allocation methods. We now briefly describe a distributed routing procedure (the Distributed Least Congested Path (DLCP) in [14, 15]), which the various strategies can be combined with.

3. The routing schemes

The DLCP routing works as follows. At connection set-up, a call request packet is forwarded from node to node in a hop-by-hop fashion. At each VC-switching node along the route, the set of available outgoing links (if any) for the destination is determined by checking the available resources, according to the CAC rule in use. If no link can carry it, the connection is dropped, and the resources previously allocated along the route are freed. If the set of available links is non-empty, a choice is made among them, by using the mechanism described below.

At the generic slot k of arrival of a connection request, a generic node i chooses the link to which to forward a class h connection request, by minimizing (over the subset of outgoing links ij that lead toward the destination required) the quantity

$$c_{ij}^{(h)}(k,s) = c_{ij,L}^{(h)}(k) + \alpha_j c_j^{(h)}(s) \tag{7}$$

where $c_{ij,L}^{(h)}(k)$ is a local cost related to link ij and $c_j^{(h)}(s)$ is a "global" cost, referring to the traffic conditions of node j and its successors at some slot s<k. $\alpha_j \in [0, 1]$ is a weighting 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, and we choose for it the following form

$$c_{ij,L}^{(h)}(k) = \begin{cases} \dfrac{1}{N_{ij,max}^{(h)}(k) - N_{ij,A}^{(h)}(k)} & \text{if } N_{ij,max}^{(h)}(k) > N_{ij,A}^{(h)}(k) \\ \infty & \text{if } N_{ij,max}^{(h)}(k) = N_{ij,A}^{(h)}(k) \end{cases} \tag{8}$$

where $N_{ij,max}^{(h)}(k)$ and $N_{ij,A}^{(h)}(k)$ have been defined in the previous Section (in practice, the infinity in (8) can be substituted with a number Z, large enough to ensure that no saturated link will be chosen if non-congested links are available). In more detail, we can associate our routing procedure with each of the previously described CAC and bandwidth allocation methods, by choosing

$$\begin{cases} N_{ij,max}^{(h)}(k) = N_{ij,DRS}^{(h)}(k) & \text{DRS} - \text{DLCP} \\ N_{ij,max}^{(h)}(k) = N_{ij,FRS}^{(h)}(k) & \text{FRS} - \text{DLCP} \\ N_{ij,max}^{(h)}(k) = N_{ij,ES}^{(h)}(k) & \text{ES} - \text{DLCP} \\ N_{ij,max}^{(h)}(k) = N_{ij,BES}^{(h)}(k) & \text{BES} - \text{DLCP} \\ N_{ij,max}^{(h)}(k) = N_{ij,CES}^{(h)}(k) & \text{CES} - \text{DLCP} \end{cases} \tag{9}$$

In all these cases, the difference in (8) represents the "available space". As regards the FRS scheme, the corresponding $N_{ij,max}^{(h)}(k)$ (namely, $N_{ij,FR}^{(h)}(k)$, to be used in FRS-DLCP) is obtained, starting from the state of the system and maintaining fixed the number of calls of the other traffic classes, by increasing the number of calls of class h up to the boundary of the 'feasibility region' (see Fig. 2). Essentially, in all cases, the function in (8) can be interpreted as a penalty on the link congestion.

To complete the description of the algorithm, we define the cost referred to a generic node j in (7) to be composed by two terms

$$c_j^{(h)}(s) = c_{j,L}^{(h)}(s) + \alpha_j c_{j,A}^{(h)}(s) \tag{10}$$

where α_j is a weighting coefficient. $c_{j,L}^{(h)}(s)$ represents the average situation of the node with respect to the congestion state of its links, and $c_{j,A}^{(h)}(s)$ is an aggregate information on the average congestion of its adjacent nodes. More specifically, we define

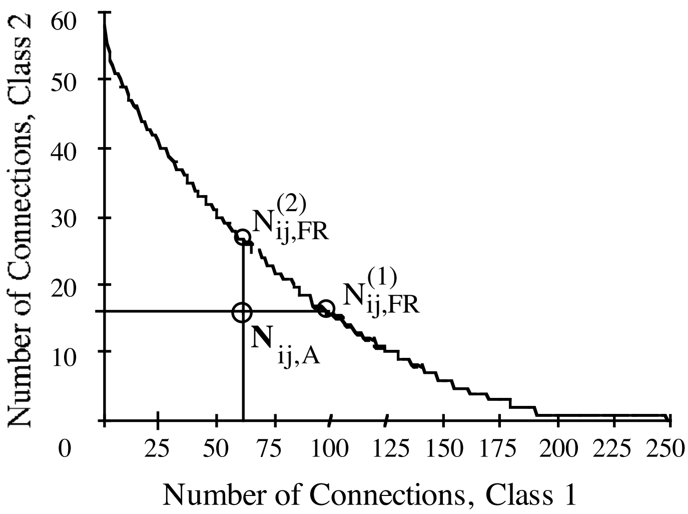

Figure 2. FRS acceptance scheme.

$$c_{j,L}^{(h)}(s) = \frac{1}{L_j} \sum_{k \in Succ(j)} c_{jk,L}^{(h)}(s) \tag{11}$$

$$c_{j,A}^{(h)}(s) = \frac{1}{L_j} \sum_{k \in Succ(j)} c_k^{(h)}(s) \tag{12}$$

Succ(j) being the set of nodes that are successors of node j (i.e., such that jk is a link outgoing from node j), and L_j its cardinality. More in detail, $c_{j,L}^{(h)}(s)$ is related to the "free space" left for the links outgoing from node j; $c_{j,A}^{(h)}(s)$ is the average of the costs related to each node connected to node j. As can be seen, the values related to the successor nodes are referred to the instants s, where s=T, 2T, ..., with T equal to a fixed number of slots. This means that each node i sends its costs $c_i^{(h)}(s)$, h=1,..., 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. This updating mechanism is adopted in the simulations; however, an asynchronous information exchange is also possible, as noted in [14].

As we mentioned, the list of links to be examined for a given destination is a subset of the ones outgoing from the node; the links that will not lead to the specified destination are dropped from the list. However, this does not mean the aggregate costs have to be necessarily separated for each destination; the aggregate cost is an overall "non-real-time" measure of forward congestion, without any distinction as regards destinations. This may be considered a rather rough aggregation, but the operation of distinction among destinations is very heavy from a computational viewpoint and would negatively affect the scalability of the algorithm (as, e.g., in the well-known vector distance routing scheme, distributed version of the Bellman-Ford algorithm [2]). In this respect, even though the exchange mechanism is similar, our algorithm differs from the vector distance family. It must be noted, however, that the traffic intensities at the inputs to the links, which are used by BS, ES and CES in their computation, are now dependent on the routing decisions. In practice, they must be estimated over a time window, and the corresponding bandwidth partitions must be periodically recomputed. Thus, also these schemes will exhibit an adaptive behaviour as the DRS, with the rectangular regions "moving" over the FR. In all cases, this effect, being affected by and affecting the routing decisions, might give rise to an oscillatory behaviour; even though this should happen to a lesser extent than in shortest path algorithms with delay metrics, owing to the "smoother" aggregate cost, this fact deserves further investigation.

It could be observed that cell-level performance requirements in our approach are chosen on a node-by-node basis. The problem of guaranteeing end-to-end cell-level performance requirements is not considered in this work and it is currently being investigated; however, simple solutions can be suggested as, for example, setting node-by-node constraints from the end-to-end ones, by accounting for a fixed maximum path length (as in the approach called Static Service Separation/Multiplexing Across Routes - Restricted Version in [3]). It is also worth noting that basing the cell level calculations on a given model of the external traffic sources is valid for the originating access nodes; at each node along the path, the characteristics of the cell streams at the output of the multiplexers change, in general. We have adopted the same model throughout, by considering that, if the cell loss and delay probabilities are minute, the input to a downstream buffer may be approximated by the superposition of the inputs to the buffers it drains from (see again [3, p. 214]).

4. Numerical results

In this Section, we report the results of several simulations that have been performed on a simple twelve-node test network, in order to obtain some indications on the performance of the proposed routing schemes, and to compare them with other possible solutions. Two traffic classes (H=2), a 'reallocation interval' $K = 32 \cdot 10^7$ cells, an updating time $T = K/10$ and a transfer capacity $C = 150$ Mbits/s (with a related slot duration $t_s = 2.83 \cdot 10^{-6}$ s), for all channels, have been used. The quantities $\square^{(h)}$ [Erlangs], h=1, ..., H, represent the global average traffic intensities offered to the network; the call arrival processes follow independent Poisson distributions, and connections have exponentially distributed duration. All other parameter values are shown in Table 1.

We refer to a 'reference' traffic flow generated by the above data as an offered load 1 when $\square^{(1)}=320$, $\square^{(2)}=48$, unless stated otherwise. An offered load "x" corresponds to the same data, except for the traffic intensities $\square^{(h)}$, h=1, 2, which are multiplied by x. The coefficients $\square_i$ and $\square_i$, i=0,...,11, are considered to be the same for each node, i.e., $\square_i=\square$ and $\square_i=\square$, $\square$i.

TRAFFIC CLASS: h	h=1	h=2
PEAK BANDWIDTH: $P^{(h)}$	2 Mbits/s	10 Mbits/s
BURSTINESS: $b^{(h)}$	5	10
AVERAGE BURST LENGTH: $B^{(h)}$	500 cells	1000 cells
AVERAGE CONNECTION DURATION	15 s	25 s
Ploss UPPER BOUND: $\square^{(h)}$	0.0001	0.0001
Pdelay UPPER BOUND: $\square^{(h)}$	0.001	0.001
DELAY CONSTRAINT: $D^{(h)}$	200 slots	1000 slots
BUFFER LENGTH: $Q^{(h)}$	15 cells	10 cells

Table 1. Parameter values.

The topology of the network that has been used in the simulations is shown in Fig. 3; it contains twelve nodes, with only one destination (node 11).

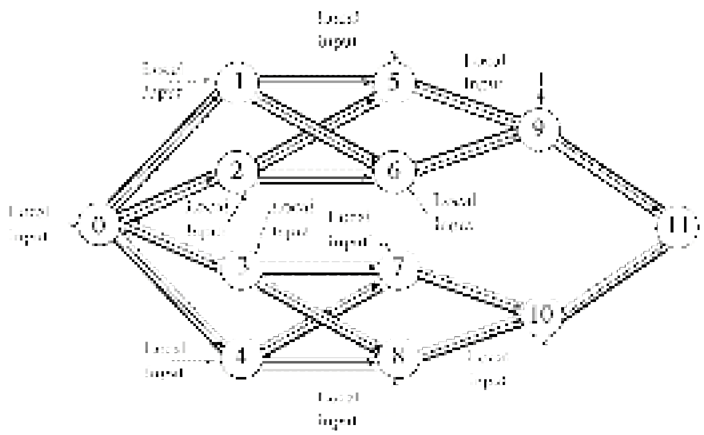

Figure 3. Topology of the test network.

The behaviour of the access control and bandwidth allocation procedure at a node was tested extensively in [13], and other simulation results on the performance of the DLCP routing scheme and its variations are reported in [14, 15]. Here, the aim is to investigate the behaviour of DLCP under the different CAC and bandwidth allocation policies that we have introduced. The results presented in the following are organized in two different scenarios: i) all nodes in the network, except the destination, generate external traffic; ii) only two nodes (namely, nodes 0 and 9) generate external traffic, in the proportions: 37.5% to node 0, 62.5% to node 9. In both cases, two alternative generation patterns are considered, consisting of constant and variable generation rates, respectively. With constant load, all simulations extend over a time period such that the 95% confidence interval is less than 3% of the value to be estimated; in the case of variable load, the simulation covers approximately 12 hours of network time (48 reallocation intervals).

Case i)

We begin by investigating the sensitivity of the system to parameters □ and □. The effect of both parameters on the percentage of blocked calls is shown in Fig. 4. The figure refers to the load 0.6; the behaviour is similar for other load values. In this case, the best choice appears to be □=1, □=0; actually, in a relatively small network with a uniform generation, the average number of hops to reach the destination is quite low and, as a consequence, the importance of the downstream aggregate information is greatly reduced. The values □=1, □=0 will be used in all further figures relating to case i).

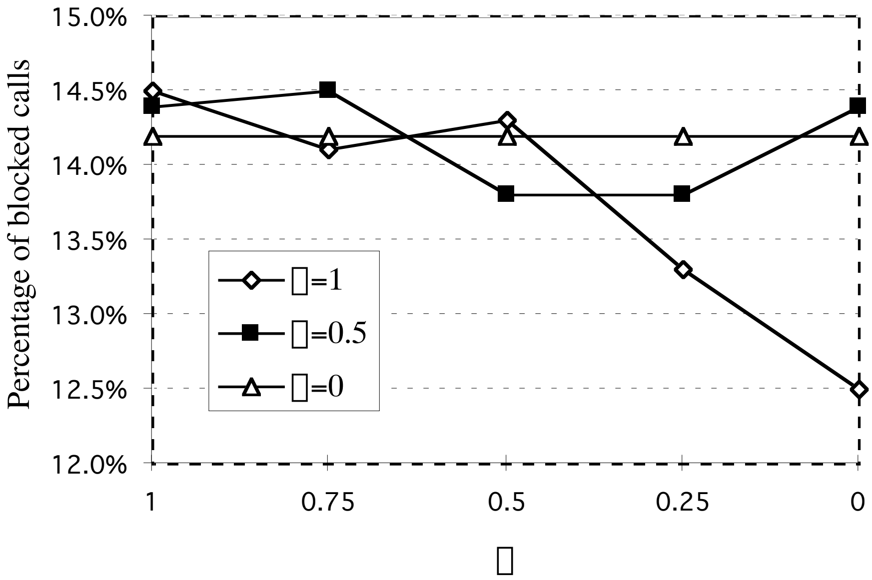

Figure 4. Percentage of blocked calls vs. coefficient □, for different values of coefficient □ (case i); offered load 0.6; DRS-DLCP).

The various schemes (except the CES) are compared in Fig. 5, with respect to the overall call blocking probability (irrespectively of the class). As a further term of comparison, another routing strategy has been introduced, which represents an "extreme" situation, namely, a completely centralized Shortest Path Routing (SPR) strategy (based on the ES-DLCP metric). As might be expected, the ES-DLCP yields the lowest overall blocking at high load, among the distributed strategies (very close to the centralized SPR). Actually, the objective of the ES-DLCP mechanism is just the minimization of the overall call blocking probability. The results about the CES-DLCP scheme are not reported here because the CES-DLCP constraints are referred to each traffic class and not to the overall percentage; in this context, the CES-DLCP results would not be meaningful. The highest values of the overall blocking percentage are obtained by using the BES-DLCP, aimed at balancing the blocking probability among the various classes; this effect is evidenced in Fig. 6, where the percentage of blocked connections for each class is depicted, versus the traffic load. It can be also observed that the low overall blocking is 'paid' by the ES-DLCP with a marked imbalance. The same behaviour is present also for the SPR. In can be noted in Fig. 6 that a satisfactory balancing is provided also by the DRS-DLCP mechanism. Generally, the results of this section show that the control algorithms at CAC level [13], extended by a suitable routing algorithm, maintain their characteristics and allow the network manager to enforce some performance requirements throughout the overall network.

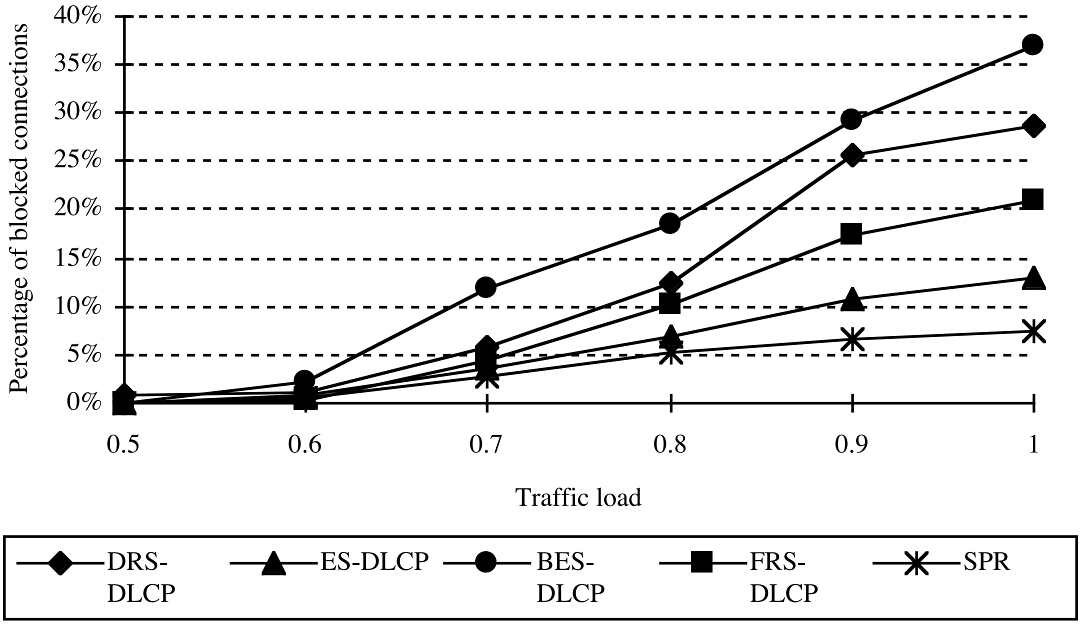

Figure 5. Overall percentage of blocked calls vs. offered load (case i)).

Fig. 7 is dedicated to the CES-DLCP, which allows to fix precise requirements in terms of percentage of blocked connections for each class. The notation CES [x%-y%] means that the upper bounds for the percentage of blocked connections are x (concerning class 1) and y (concerning class 2), respectively; this means that $\square^{(1)} = 0.01 \cdot x$ and $\square^{(2)} = 0.01 \cdot y$, where $\square^{(h)}$ is the constraint for class h blocking probability. As can be seen from Fig. 7, the CES-DLCP strategies tend to maintain the constraints within the specified range. Again, this behaviour closely reflects that of the corresponding CAC schemes when examined on their own, relatively to a single ATM multiplexer [13].

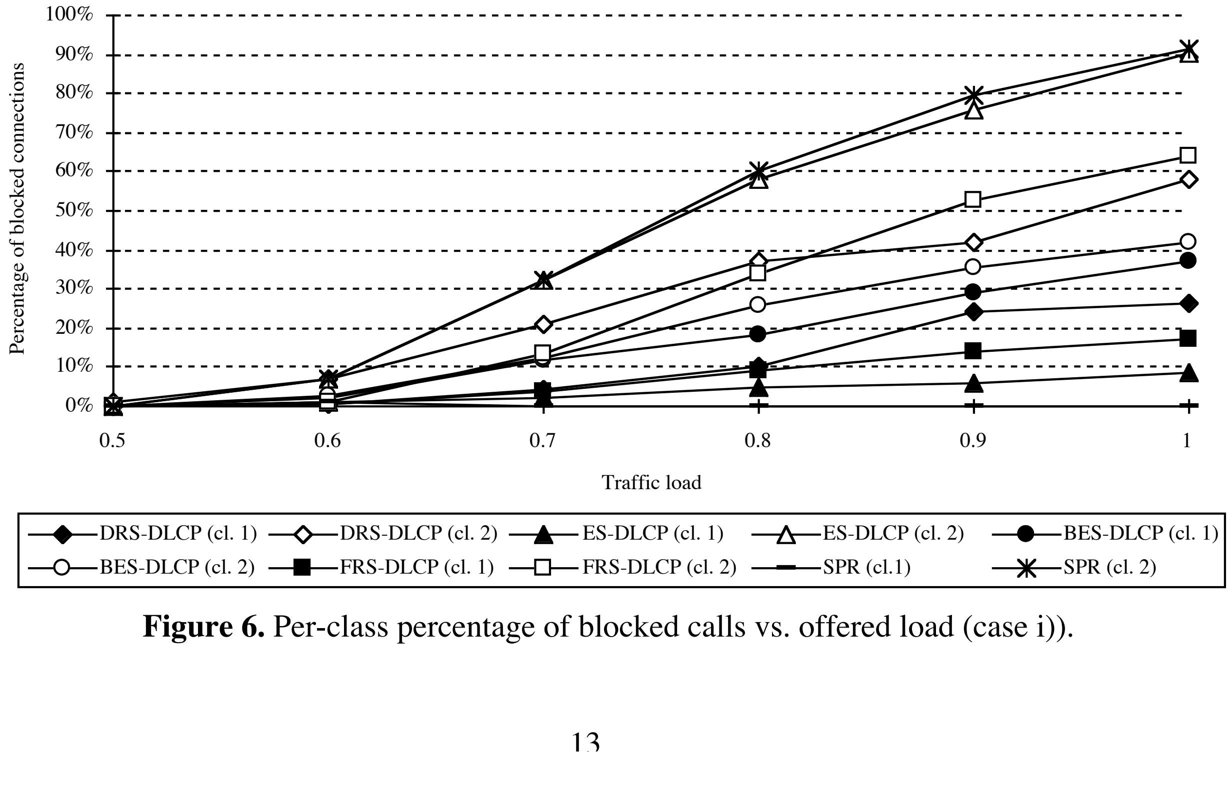

Figure 6. Per-class percentage of blocked calls vs. offered load (case i)).

In more detail, concerning [20%-20%] values, the constraints are satisfied for a traffic load lower than 0.7; beyond this value the scheme has a balancing effect. Concerning the other configurations, the constraints, even the more restrictive ones (as 1%) are satisfied in almost all the situations considered; if there are not enough resources, for very high traffic load values, the values obtained are close to the desired ones.

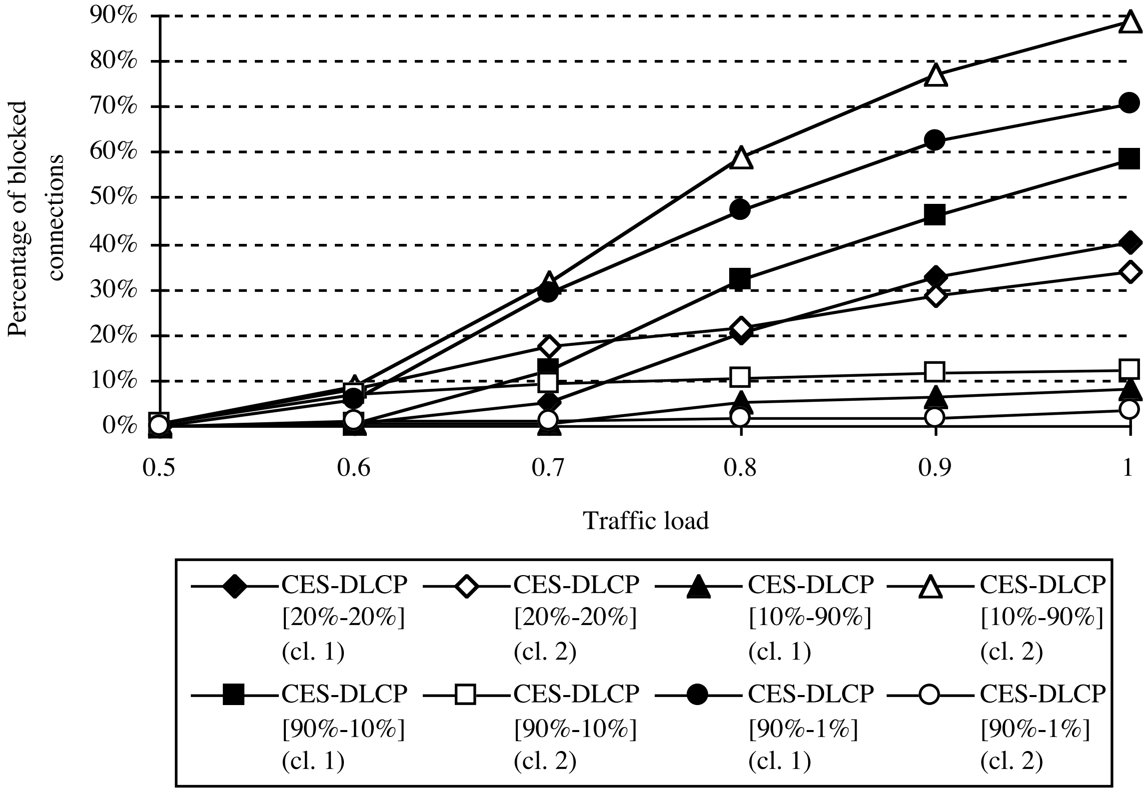

Figure 7. Per-class percentage of blocked calls vs. offered load (CES-DLCP, case i)).

As regards the time-varying load situation, we have tested all schemes by using the load pattern shown in Fig 8, where class 2 load is maintained fixed, whereas class 1 load is changed every 8 reallocation intervals of the DRS (approximately corresponding to 2 hours). In this case there is one traffic load reference for each class: $\square^{(1)}$=320, for class 1, $\square^{(2)}$=48, for class 2.

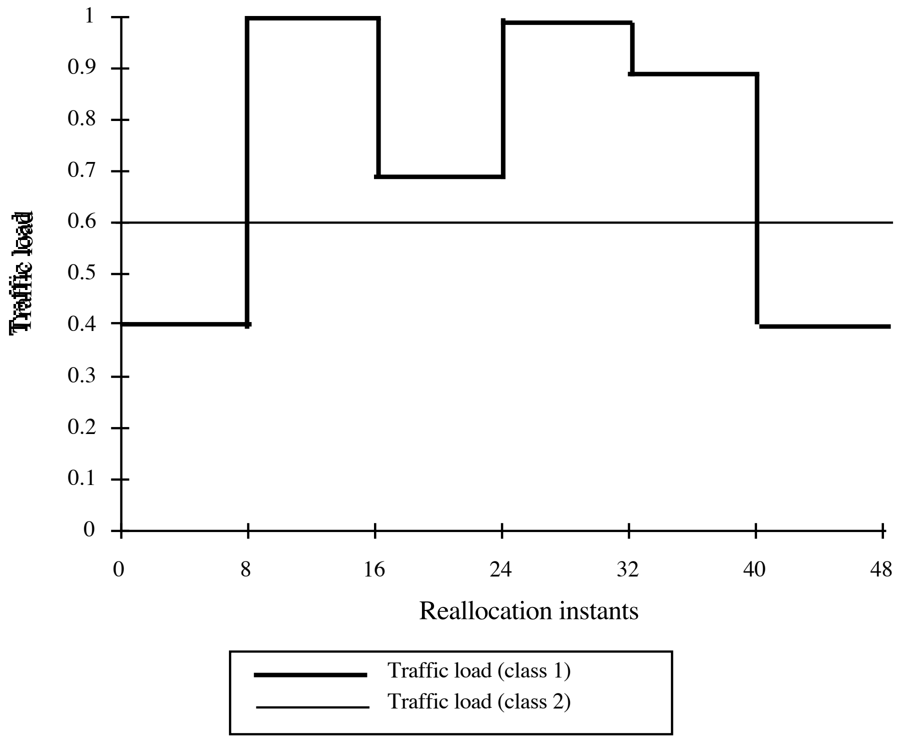

Figure 8. Time-varying traffic pattern.

The results obtained are reported in Fig. 9, where the percentage of blocked connections (overall and per-class) is shown for the various solutions proposed. ES-DLCP yields the minimum values for the overall blocking probability; BES-DLCP, along with CES-DLCP [20%-20%] and DRS-DLCP, allow to balance the percentage between the two classes; in general, CES-DLCP allows to establish call level constraints. Besides the already mentioned SPR (with ES-DLCP metric), we have added the results of a Hot-Potato strategy, as a further term of comparison.

Case ii)

Again, the first graph in this set (Fig. 10) depicts the overall percentage of blocked connections versus the weighting coefficients □ and □. For non-zero values of □ and □, the network is aware of the bottleneck in node 9, and avoids critic nodes by choosing the other branch; by decreasing the value of □, the network gradually looses the awareness of critic nodes, until a fully "blind" situation occurs (□=0), resulting in a drastic increase in the percentage of blocked calls. In all other simulations regarding this case, the values □=□=1 have been chosen. The results shown in Fig. 10 refer to DRS-DLCP; all other cases exhibit a similar behaviour.

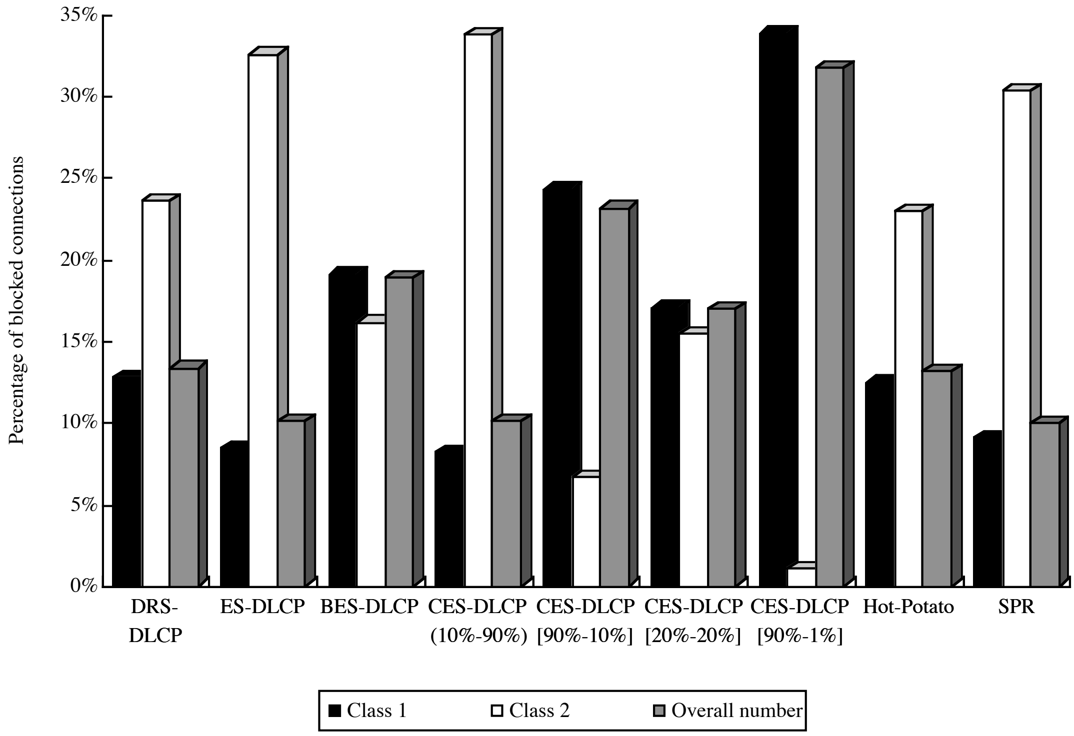

Figure 9. Time varying load situation, all strategies (case i)).

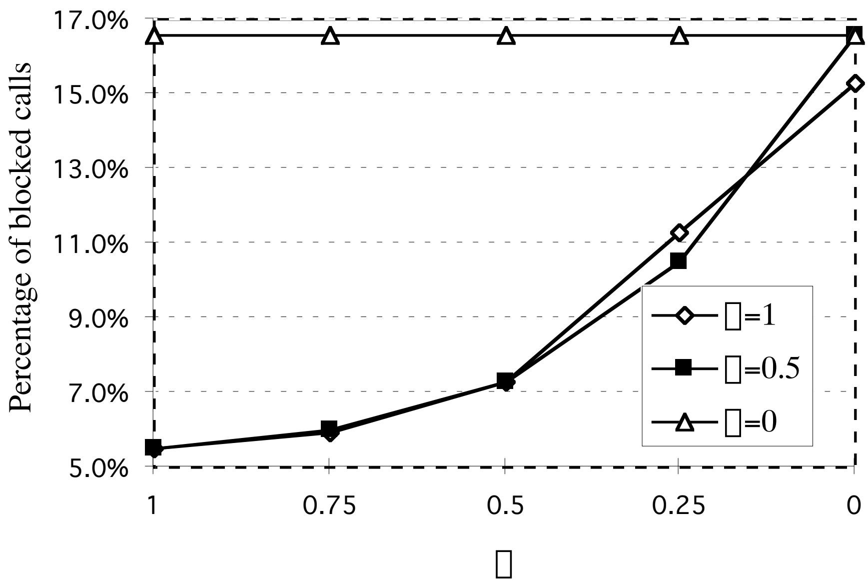

Figure 10. Percentage of blocked calls vs. coefficient , for different values of coefficient (case ii); offered load 0.6; DRS-DLCP).

The next figures (11 to 14) show the same quantities, except for the traffic configuration, as in Figs. 5, 6, 7 and 9, respectively. Fig. 14 is obtained by using the time-varying traffic load, whose configuration is shown in Fig. 8. Substantially, the same qualitative observations already reported for the corresponding figures can be repeated, even if the numerical values are completely different.

Furthermore, we can note that in such an unbalanced situation the complete absence of constraints in the FRS-DLCP mechanism allows to obtain, at low load, the best results concerning the overall blocking percentage (Fig. 11), at the price of a marked imbalance between the classes (Fig. 12). The SPR is not reported in Fig. 12, because its behaviour is very similar to that in Fig. 6. The CES-DLCP schemes (Fig. 13) allow to fix constraint on the call blocking probabilities which, however, cannot always be satisfied in this unbalanced situation.

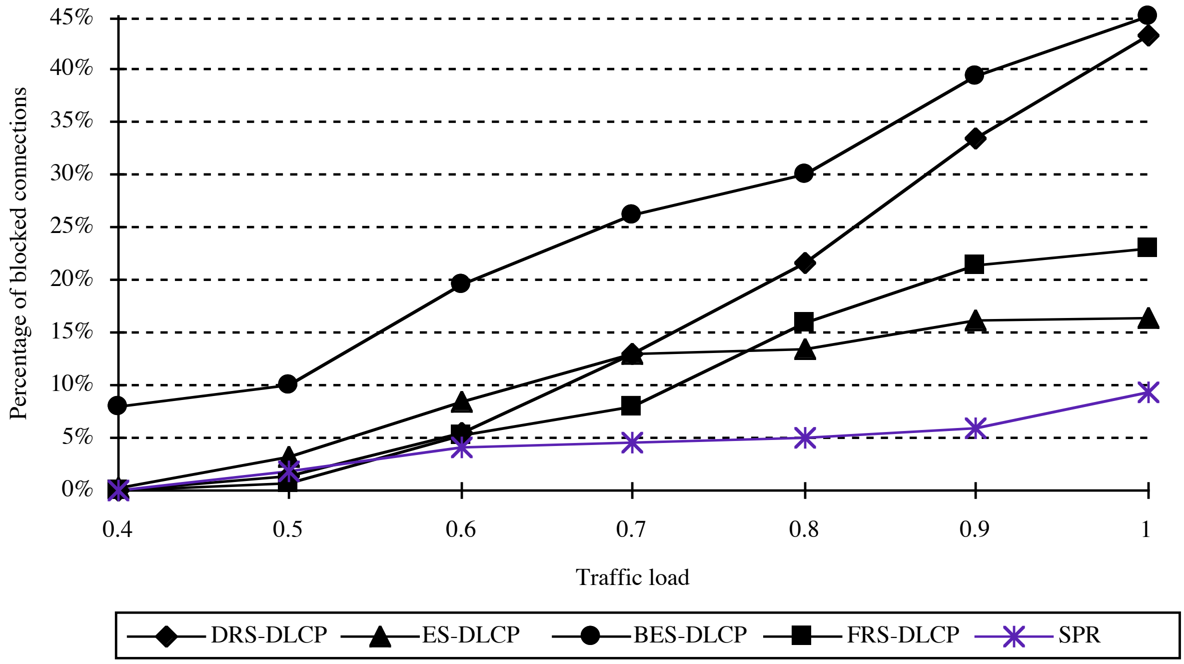

Figure 11. Overall percentage of blocked calls vs. offered load (case ii)).

5. Conclusions

A global control architecture for access control and routing has been considered, in an ATM network environment. The traffic is organized into service classes, characterized by specific performance requirements.

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 (DLCP) based on local (real time) as well as on aggregated (delayed) information. Five routing strategies have been explicitly defined in the paper, each one originating from a corresponding CAC scheme.

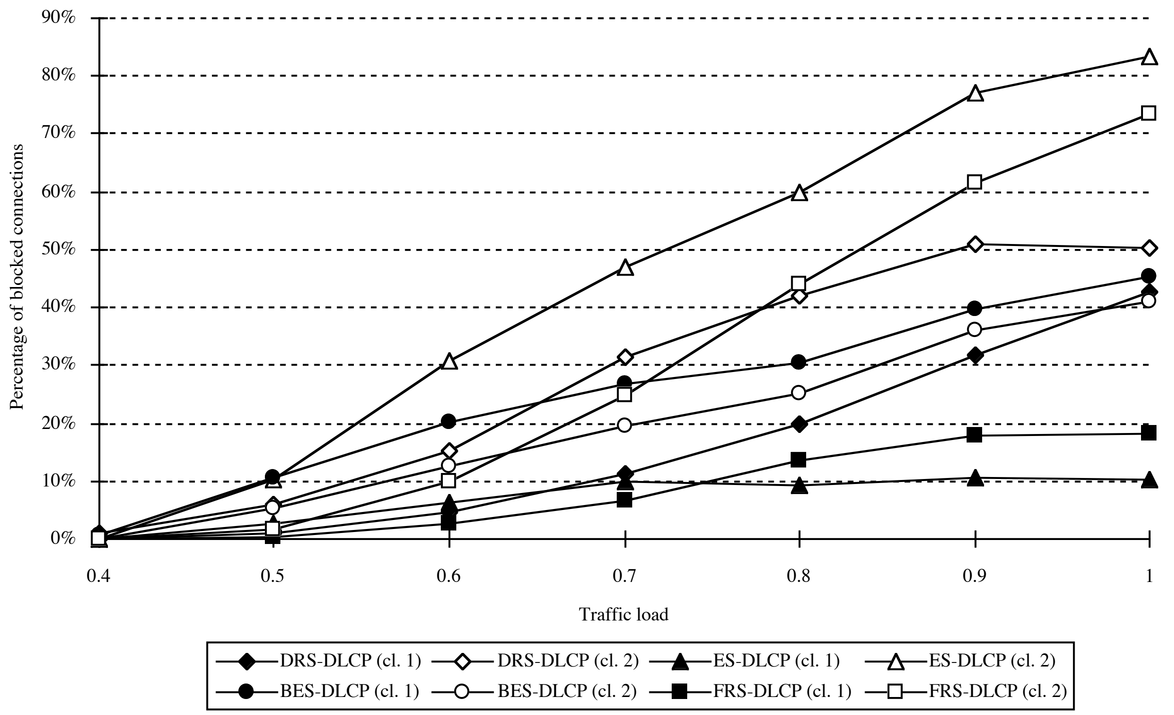

Figure 12. Per-class percentage of blocked calls vs. offered load (case ii)).

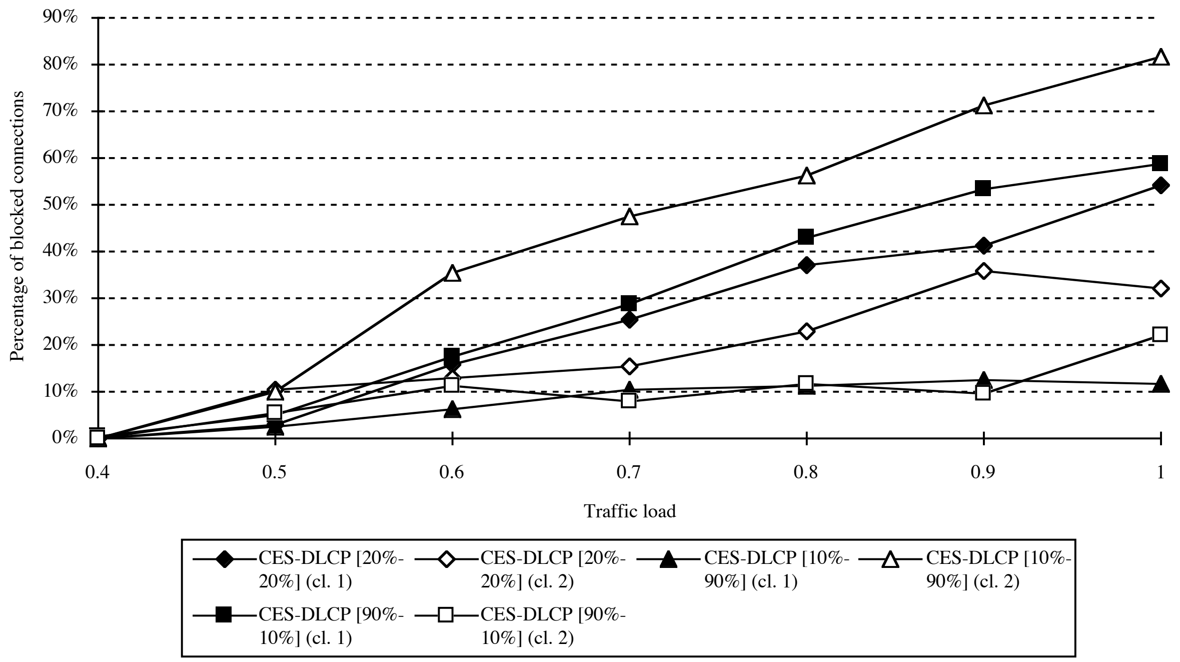

Figure 13. Per-class percentage of blocked calls vs. offered load (CES-DLCP, case ii)).

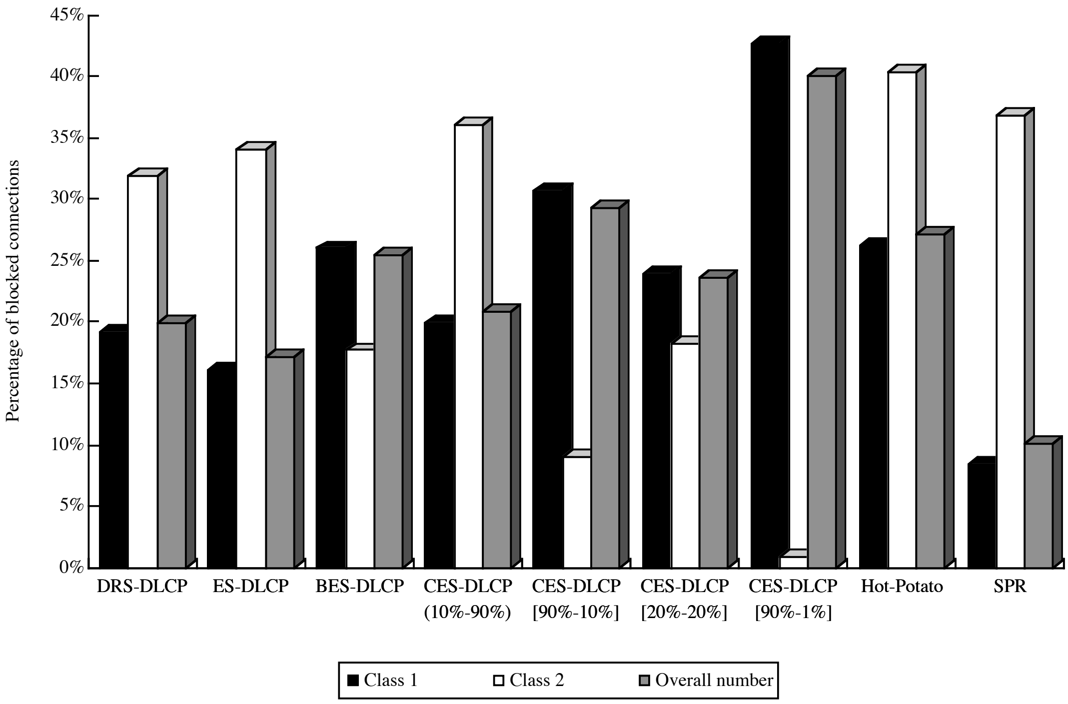

Figure 14. Time varying load situation, all strategies (case ii)).

Simulation results have been reported and compared with other routing strategies that can be considered as "extreme" cases, namely, a totally decentralized local Hot Potato routing and a centralized Shortest Path one. The results highlight a low call rejection rate as an overall effect of the control structure, over a wide range of network load values. Moreover, the overall network behaviour tends to inherit the one exhibited by the specific CAC scheme at a single ATM multiplexer.

Work still remains to be done to investigate the convergence properties and the possible looping effects of the DLCP mechanism in general.

References

[1] A. Girard, *Routing and Dimensioning in Circuit-Switched Networks*, Addison-Wesley, Reading, MA, 1990.

[2] D. Bertsekas, R. Gallager, *Data Networks*, Prentice-Hall, Englewood Cliffs, NJ, 1987.

[3] K.W. Ross, *Multiservice Loss Models for Broadband Telecommunication Networks*, Springer Verlag, London, 1995.

[4] Y. Takagi, S. Hino, T. Takahashi, "Priority assignment control of ATM line buffers with multiple QOS classes", *IEEE J. Select. Areas Commun.*, vol. 9, pp. 1078-1092, Sept. 1991.

[5] A.A. Lazar, A. Temple, R. Gidron, "An architecture for integrated networks that guarantees quality of service", *Internat. J. of Digital and Analog Commun. Systems*, vol. 3, pp. 229-238, July 1990.

[6] J.M. Hyman, A.A. Lazar, G. Pacifici, "Real time scheduling with quality of service constraints", *IEEE J. Select. Areas Commun.*, vol. 9, pp. 1052-1063, Sept. 1991.

[7] J.M. Hyman, A.A. Lazar, G. Pacifici, "A separation principle between scheduling and

admission control for broadband switching", *IEEE J. Select. Areas Commun.*, vol. 11, May 1993.
[8] G. Gallassi, G. Rigolio, L. Verri, "Resource management and dimensioning in ATM networks", *IEEE Network Mag.*, pp. 8-17, May 1990.
[9] K. Sriram, "Methodologies for bandwidth allocation, transmission scheduling, and congestion avoidance in broadband ATM networks", *Comput. Networks and ISDN Syst.*, vol. 26, pp. 43-59, 1993.
[10] A. Gupta, D. Ferrari, "Resource partitioning for real-time communication", *IEEE/ACM Trans. Networking*, vol. 3, pp. 501-508, Oct. 1995.
[11] R. Bolla, F. Davoli, A. Lombardo, S. Palazzo, D. Panno, "Hierarchical dynamic control of multiple traffic classes in ATM networks", *Europ. Trans. Telecommun.*, vol. 5, no. 6, pp. 747-755, Nov. 1994.
[12] R. Bolla, F. Davoli, M. Marchese, "Simple schemes for traffic integration at call set-up level in ATM networks", *Computer Commun.*, vol. 19, pp. 645-652, 1996.
[13] R. Bolla, F. Davoli, M. Marchese, "Bandwidth allocation and admission control in ATM networks with service separation", *IEEE Commun. Mag.*, vol. 35, no. 5, pp. 130-137, May 1997.
[14] R. Bolla, A. Dalal'ah, F. Davoli, M. Marchese, "Dynamic route selection at call set-up level in ATM networks", in D.D. Kouvatsos, Ed., *ATM Networks Performance Modeling and Analysis*, Chapman & Hall, 1997, pp. 121-140.
[15] R. Bolla, A. Dalal'ah, F. Davoli, M. Marchese, M.S. Obaidat, "Adaptive distributed admission control and routing in ATM networks", *Internat. J. of Commun. Syst.*, vol. 10, pp. 215-225, 1997.
[16] S. Gupta, K.W. Ross, M. El Zarki, "Routing in virtual path based ATM networks", *Proc. IEEE Globecom'92*, Orlando, FL, Dec. 1992, pp. 571-575.
[17] S. Gupta, P.P. Gandhi, "Dynamic routing in multi-class non-hierarchical networks", *Proc. IEEE Internat. Conf. Commun. (ICC '94)*, New Orleans, LA, May 1994, pp. 1390-1394.
[18] J.M. Hyman, A.A. Lazar, G. Pacifici, "A methodology for Virtual Path and Virtual Network bandwidth assignment in broadband networks with QOS guarantees", *Proc. IEEE Internat. Conf. Commun. (ICC '94)*, New Orleans, LA, May 1994, pp. 165-169.
[19] A. Maunder, P.S. Min, "Routing for multirate traffic with multiple qualities of service", *Proc. Internat. Conf. on Computer Commun. and Networks (ICCCN'94)*, San Francisco, CA, Sept. 1994, pp. 104-108.
[20] W.-T. Chen, U.-J. Liu, "Routing problem with performance requirement translation for multimedia communications in an ATM wide-area network" *Proc. IEEE Internat. Conf. Commun. (ICC '94)*, New Orleans, LA, May 1994, pp. 1490-1493.
[21] S. Plotkin, "Competitive routing of virtual circuits in ATM networks, *IEEE J. Select. Areas Commun.*, vol. 13, pp. 1128-1136, Aug. 1995.
[22] K. Murakami, H.S. Kim, "Virtual Path routing for survivable ATM networks", *IEEE/ACM Trans. Networking*, vol. 4, pp. 22-39, Feb. 1996.
[23] R. Guérin, H. Ahmadi, M. Naghshineh, "Equivalent capacity and its application to bandwidth allocation in high speed networks", *IEEE J. Select. Areas Commun.*, vol. 9, no. 7, pp. 968-981, Sept. 1991.
[24] R. Bolla, F. Danovaro, F. Davoli, M. Marchese, "An integrated dynamic resource allocation scheme for ATM networks", *Proc. IEEE Infocom '93*, San Francisco, CA, March 1993, pp. 1288-1297.
[25] J.P. Cosmas, G.H. Petit, R. Lehnert, C. Blondia, K. Kontovassilis, O. Casals, T. Theimer, "A review of voice, data and video traffic models for ATM", *Europ. Trans. Telecommun.*, vol. 5, no. 2, pp. 11-26, March-April 1994.