7

DYNAMIC ROUTE SELECTION AT CALL SET-UP LEVEL IN ATM NETWORKS

R. Bolla, A. Dalal'ah, F. Davoli, M. Marchese

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

Abstract

A new policy for routing in ATM networks is introduced. Virtual circuits are routed at call set-up level, by combining Call Admission Control (CAC) and routing decisions in a hop-by-hop fashion at the nodes along a path. The network traffic is divided into classes, homogeneous with respect to performance requirements and statistical characteristics, which are dynamically assigned bandwidth partitions over the network links, by means of scheduling algorithms at the switching nodes. 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 (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. In case that no route is available at a node, the call request is traced back to the least-loaded node (according to the measure chosen) among the ones already visited along the path. This strategy is analysed and compared by simulation, in terms of performance and of the capability of reacting to network failures, with a previous one that immediately drops the call at the node where blocking occurs.

Keywords

ATM, routing.

1 INTRODUCTION

The heterogeneous traffic flows nature and performance characteristics in the Asynchronous Transfer Mode (ATM) technique give rise to several control problems, such as bandwidth allocation, admission control and routing, (Onvural, 1994, Ross, 1995, Saito 1994, Roberts 1992), which still deserve investigation, both in their own respect and in their interaction. The routing problem, in fact, is considered a part of the admission control scheme: at the connection set-up, a decision has to be taken about the acceptance or rejection of the new call and a source-destination route has to be chosen for that connection. On the other hand, the admission schemes are strictly related to the allocation of the resources (bandwidth or buffer space) in the network; nonetheless, trying to solve the listed control issues independently not only simplifies the global task, but also yields greater flexibility in combining different schemes.

The proper design of a routing algorithm involves many related issues, from network dimensioning to the topology of the network under analysis. Concerning the first one, the problem is often decoupled into a bandwidth estimation problem and a combined virtual path routing and capacity design problem (Medhi, 1995). In this cases the choice and configuration of Virtual Paths (VP) is a crucial point (see (Pióro, 1995, Arvidsson 1995, Sharma, 1995, Murakami, 1996), for references about the topic). As far as the topology is concerned, some hypotheses are often made in the literature: full connectivity, two physical hops at most (Murthy, 1995), or, about traffic conditions, homogenous traffic flow (Begain, 1995). In our approach these constrained are relaxed to allow an efficient use of the ATM network resources in a real environment.

Besides, an efficient routing strategy should have some fundamental properties: for example, it should be adaptive to changes in the network, by having a prompt restoration from a network failure and a fast reaction to a change in the traffic conditions (Murakami, 1996), very useful to improve the network performance. It would be rather valuable to have a routing strategy independent from a Virtual Circuit (VC) arrival model that is known 'a priori'. So, a family of 'competitive analysis' routing strategies based on on-line information and on no statistical assumption on the VC arrival pattern has been summarised in (Plotkin, 1995). From an implementation point of view, routing strategies, even if combined with CAC schemes, should be very fast; to this aim, the evaluation of the routing algorithms and CAC function delays should be performed with a particular attention (Hwang, 1995, Cornero, 1994).

In the present paper, we propose two new strategies. A first one, named Re-attempt Distributed Least Congested Path (R-DLCP) in the following, and a second one, a slight modification of R-DLCP, called Re-attempt First-fit Distributed Least Congested Path (RF-DLCP). Both of them operate in a hop-by-hop fashion to route a connection request, with the possibility of tracing back to a previously visited node in case of failure in finding an available route at a certain point along the path. Although this mechanism can be applied to a family of algorithms, the strategies that will be described are proposed to overcome some shortcomings, in particular with respect to robustness to failures, emerged from the DLCP (Distributed Least

Congested Path), a routing method previously introduced by the authors (Bolla, 1995), and to enhance, in general, its performance as well. The R-DLCP and the RF-DLCP are step-backward algorithms, in the sense that, as mentioned above, if the process of establishing a path faces a congested point, it returns back to some node and retries again.

The present work is proposed to be integrated in an overall control scheme, so that the admission control and bandwidth allocation methods used in (Bolla, 1995) are preserved. The traffic is divided into classes, characterised by statistical characteristics and Quality of Service (QoS) requirements and, upon a connection request, the admission control and the routing strategy are implemented independently for each class of traffic. The routing decisions are realised in a distributed computational structure.

The paper is organised as follows. In the next section, the features of the proposed algorithm are discussed; in Section 3, the routing strategy is described; in Section 4, some simulation results are considered and discussed, and Section 5 is dedicated to the conclusions.

2. FEATURES OF THE PROPOSED DYNAMIC ROUTING MECHANISM

2.1 General framework

The traffic is supposed to be divided into H classes, where each class differs from the others for the required QoS, in terms of cell loss and delayed cell rate, and for statistical properties, such as average (or sustainable) and peak cell rate. The source model used in the computations is an Interrupted Bernoulli Process (IBP), and the superposition of IBP sources is considered to derive QoS requirements at the cell level analytically (cell loss rate and delayed cells rate). The latter provide the basis for the Admission Control and bandwidth allocation rules. The actual source generation process used in the simulation runs is a superposition of Talkspurt-Silence models (Bolla, 1997). The distribution of the connection time is assumed to be exponential.

Each class is dedicated a fixed amount of bandwidth over each link, as in a Complete Partitioning scheme (Ross, 1995). However, we suppose that a dynamic fair assignment of the bandwidth partitions among the various classes of traffic is maintained by link bandwidth allocators (Bolla, 1995), which periodically adjust the bandwidth assignments, to be kept over a fixed period of time. At each node, a connection of a certain class is admitted if the nodal local admission control can find at least an outgoing link over which the call can be accommodated, by ensuring the desired QoS level to the new call as well as to the ones in progress.

2.2 General behaviour of the proposed routing strategies

At connection set-up, the source sends a special packet, which will be called Resource Reservation Packet (RRP) in the following. When a node receives a RRP, it verifies whether it can accommodate the required resources at one of its outgoing links towards the destination or not. If it turns out that there is at least one available link, the routing algorithm, yet to be described, is used to send the RRP to the successor node; otherwise, in the R-DLCP and the RF-DLCP strategy, the RRP returns back to the least loaded node (whose definition will be given in the following), along the already established path, and it retries again starting from that node, by choosing another outgoing link, but excluding the link previously chosen. The strategy to choose the new link to be followed characterises the difference between R-DLCP and RF-DLCP. If the RRP cannot find the required resources over one of the outgoing links of the above mentioned node, it will be changed into a Free Resource Packet (FRP), which must be sent back in its turn along the already established path, releasing the resources reserved for the call related to it. With a similar philosophy as in (Dighe, 1995), the RRP is sent forward with two objectives, the first one being the establishment of a primary route and the second one being the usage of an alternative route in the case of failures. The alternative route, as the primary one as well, is released only upon the completion of the call. For the sake of non saturating the network, the process of re-routing the RRP back to the least loaded node along the already established path is limited just to a single attempt.

The decisions of the algorithm are based on the minimisation of the sum of a "local cost" and an "aggregate cost", evaluated at the node at the moment of the decision only; thus, there is no need to know the global situation over the network to execute the algorithm at a specific node. More specifically, the information needed to take a decision is limited to the node under consideration and to an information exchange mechanism among adjacent nodes.

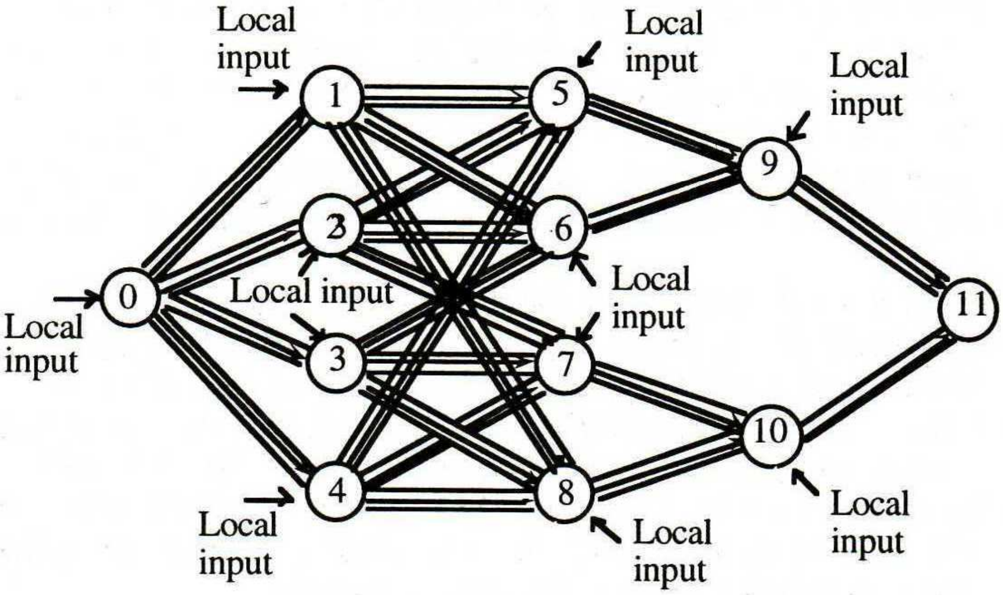

Figure 1. A network topology to illustrate node or link failure.

It was argued in (Shaikh, 1995) that basing the decision on a similar kind of information makes the routing procedure more practical from an implementation point of view. In the case of a huge network, the overhead incurred in acquiring remote information would be considerable; as will be apparent in the following, the aggregate information exchanged among the nodes in our case is limited to an "indication" of the downstream saturation, without regard to the specific destinations. In that sense, the algorithm can be considered to be scalable (even though, obviously, the "age" of the aggregate information, as an indicator of the network status, grows with the network size).

2.3 Management of network failures

In general, any routing algorithm must be robust to link or node failures. To illustrate this point, consider Figure 1, which depicts a network topology that we will use for this purpose; node 11 is the only destination. Suppose that the link between node 9 and node 11 is broken or node 9 has failed. In the DLCP algorithm of (Bolla, 1995) (where the same information exchange mechanism is used as in the R-DLCP, but a call is rejected if no available route is found at a node, without re-attempting to find an alternative path), the neighbours of node 9, after notification by the signalling protocol (in case of node failure) or by node 9 itself (in case of link failure), would increment their cost. The increment would then be propagated downstream, but it would take some time (depending on the frequency of the cost updating mechanism) to reach the furthermost nodes; in the meanwhile, the latter would be unaware of the congestion, and might still direct calls in the direction of node 9, which might incur higher rejection probability the closer they get to it. This problem might be overcome by using an algorithm as the one proposed in (Dighe, 1995); however, in that approach, an alternative path must be established from every node traversed to its successor on the way to the destination and to be released only upon the completion of the call. Thus, a part of the network capacity is dedicated for the re-routed calls after any single failure, which not only diminishes the overall capacity of the given network, but is also hard to compute. On the other hand, the R-DLCP and the RF-DLCP do not dedicate any part of the network to be used only in the case of failure, and they do not always guarantee the availability of the alternative route. So, they are less safe than the strategies in (Dighe, 1995, Murthy, 1995), but they are simpler to implement.

The R-DLCP reacts implicitly in the same way as the DLCP, but the calls attempting to pass through node 9 would be deviated, instead of being refused, by the back-tracking mechanism. Thus, the nodes in the network other than node 9 should not suffer unnecessary call rejection during the transient period, before the network becomes aware of the congestion. In fact, if the other part of the network can accommodate those calls, they will be accepted. In this sense, the R-DLCP better meets the robustness requirements.

2.4 Scope of the proposed algorithm

Concerning the scope of the algorithm, we think it could range from the small private network to a "medium size" public one (or a portion of a large one, if embedded in a hierarchical scheme); in fact, the only limit to the scalability of the algorithm is the potential staleness of the aggregate information coming from very distant parts of the network (which is implicitly contained in the information received by the neighbours). In the authors' opinion, the proposed strategy seems to be a reasonable compromise (in terms of accuracy and quantity of information exchanged) between a totally centralised method and some totally decentralised ones, like hot-potato or flooding.

3. ROUTING STRATEGY

As stated in sub-section 2.2, when a node receives a RRP, it makes its decision about the acceptance or rejection of the call and about the route to follow, in case of acceptance, by computing a simple cost function, related to each outgoing link, given by the sum of a local cost and an aggregate cost. Let i be the node considered; then the cost of link ij for class h at decision instant k (in slots) is defined as

$$W^{(h,ij)}(k) = W_{loc}^{(h,ij)}(k) + \alpha_j W_{agg}^{(h,j)}(s) \qquad (1)$$

where ij is the link with origin i and destination j and α_j is a weighting coefficient. Actually, the index j is not related to the destination, but rather to the neighbouring nodes (or "downstream" nodes with respect to the whole set of destinations reachable from a given node). The link with minimum weight is chosen from among a subset leading to a specific destination. α_j is a weighting coefficient, whose choice can be only qualitatively suggested by simulation results or experience. The aggregate cost $W_{agg}^{(h,j)}(s)$ refers to the traffic conditions of node j and its successors, which was evaluated at instant $s<k$ (in slots) and communicated to node i (s then represents the most recent updating instant of the costs in (1)). Using the current values of the local cost and the aggregate cost at the node (which highlights the behaviour of the algorithm as a "greedy" one), has a drawback in the fact that, owing to the propagation delay of aggregate costs, in a large network, several changes may have happened, which are not reflected in the information upon which a decision is based. On the other hand, a large network dimension would also hinder the implementation of a centralised algorithm in the same way.

Let $N_k^{(h,ij)}$ be the number of connections in progress for class h on link ij and $N_{max}^{(h,ij)}$ be the maximum number of connections for class h at link ij that can be accepted, maintaining QoS and according to the admission policy. Moreover, let m indicate the slot when a bandwidth re-allocation was performed, and K the duration (in slots) of the reallocation period (we recall that we are using a Complete Partitioning scheme, where, however, bandwidth partitions of the link or Virtual Path (VP) capacities may be periodically recomputed). We define the local cost of link ij and class h at instant k as

$$W_{loc}^{(h,ij)}(k)=\begin{cases}\dfrac{1}{N_{max}^{(h,ij)}(m)-N_k^{(h,ij)}} & \text{if } N_{max}^{(h,ij)}(m)>N_k^{(h,ij)}\\ Z & \text{if } N_{max}^{(h,ij)}(m)=N_k^{(h,ij)}\end{cases} \tag{2}$$

where $k \in [m, m+K-1]$, and Z is a very large value (Z should be large enough to ensure that no saturated link will be chosen if non-congested links are available). Thus, the local cost is inversely proportional to the "available space" (in terms of the number of acceptable connections on the link), and it is Z, when there is no more bandwidth available; i.e., when no more calls of class h can be accepted on that link.

As regards the aggregate cost, we have chosen the following

$$W_{agg}^{(h,j)}(s)=W_{agg,node}^{(h,j)}(s)+\beta_j W_{agg,succ}^{(h,j)}(s) \tag{3}$$

where β_j is a weighting coefficient. Concerning β_j, a similar comment as regarding α_j can be made. $W_{agg,node}^{(h,j)}(s)$ represents the "average" situation of the node with respect to its congestion state, and $W_{agg,succ}^{(h,j)}(s)$ is an aggregate information on the "average" congestion of the nodes at the other end of its outgoing links. More specifically, we have defined

$$W_{agg,node}^{(h,j)}(s)=\frac{1}{L_j}\sum_{n\in Succ(j)} W_{loc}^{(h,jn)}(s) \tag{4}$$

$$W_{agg,succ}^{(h,j)}(s)=\frac{1}{L_j}\sum_{n\in Succ(j)} W_{agg}^{(h,n)}(s) \tag{5}$$

where Succ(j) is the set of links outgoing from node j and L_j is its cardinality.

When a node receives a RRP, whose requested destination is d, the node scans a list (already maintained by the access controller) of the links and their corresponding cost values (in non-decreasing order), and a set ℓ^d for each destination d, and stops at the first link $i\hat{j}$ with $\hat{j} \in \ell^d$. Averaging over the L_j links is used to obtain an aggregate value of the costs attributed to the nodes (irrespective of the number of outgoing links and of the destinations). If we imagine a network (a small private network or a "medium size" public one, as stated above), the set of neighbouring nodes leading to a certain destination can be maintained at each node, by propagating topological information among adjacent nodes. However, this does not mean the aggregate costs have to be necessarily separated for each destination, but only that the choice of the link may be made from among a subset of the outgoing ones. The aggregate cost is an overall 'non-real-time' measure of forward congestion, without any distinction as regards destinations. This could seem not so precise, but the operation of distinction among destinations is very heavy from a computational viewpoint and would negatively affect the scalability of the algorithm. The use of a different cost for each destination would mean to have the same amount of information as a Real-Time Network Routing (RTNR), where a distinction among the various destinations is performed (Ash, 1991). Most part of the future research will be dedicated to a complete comparison between the three presented strategies and the RTNR scheme; preliminary results have shown that the distinction among destinations is always very heavy, but it is not so convenient in any network configuration. The related quantitative results cannot be reported yet, because the matter is still under analysis.

If $W^{(h,\hat{ij})}(k) = Z$, the connection request is referred to the Least Loaded Node (LLN) r_k along the path already traversed and retries, starting at node r_k. The LLN is defined as the node, along the traversed path, with minimum $W^{(h,x)}_{agg,node}$, if h is the considered traffic class and 'x' a generic node belonging to the mentioned path (time is not mentioned here for the sake of simplicity). Then we consider two slightly different variants to choose the new link to follow, outgoing from the LLN r_k:

- the R-DLCP (Re-attempt DLCP), where the connection request retries under the same conditions (but excluding the link that was previously chosen at r_k and led to node i) of the first choice, i.e. the outgoing link with minimum cost (1) is chosen;
- the RF-DLCP (Re-attempt First-fit DLCP), where, upon reaching the re-attempt node, the first next hop is chosen from a list of the links in arbitrary order, as the first one that can accommodate the connection. This variant is aimed at avoiding the use of the cost in the first choice, in order to reduce the possibility that a prevalence of the aggregate information (not yet updated) would lead again toward the same region of the network.

If the connection is accepted at r_k, say over link $r_k s$, then the resources are reserved on that link and the connection attempt proceeds further; otherwise the connection request is refused and a FRP is sent back to release the already allocated resources. Whenever a connection is accepted on link ij, $W^{(h,ij)}$ is updated, by adding 1 to $N_k^{(h,ij)}$, and it is placed in the list in the correct position; then, the RRP proceeds to a successor node, until either it is refused or the destination is reached. In the latter case, the resources must be freed after the connection is closed by either the origin or the destination.

The updating of the aggregate parts of the cost can be effected in different ways. In the simulations reported in the next section, a periodic synchronous information exchange (with period T [slots]) was assumed, whereby each node updates its aggregate cost after receiving the updated cost by its downstream neighbours. In the formulation used in the present paper, the aggregate cost is itself "instantaneous" (i.e., a sample value of a random variable), as is the local cost, evaluated at the decision time. In this case, the effect of the updating frequency 1/T is sensible at short values of T, but becomes rather unimportant beyond a certain point. Obviously, a frequent updating implies a high amount of control information exchange. Two variations are currently under investigation: i) the introduction of a "smoothing effect" on the aggregate information values, by performing some time averaging in their calculation; ii) the possibility of performing updates and exchanging information at asynchronous instants (as done, for instance, in distributed shortest path algorithms).

Currently, the exchange is synchronous, in the sense that all the nodes at an equal number of steps from the destination act simultaneously, and would require synchronisation signals among the nodes (not an unreasonable task if a synchronous underlying physical network, e.g. SDH, is present); however, an asynchronous updating mechanism is possible, just in the same way as in distributed shortest path algorithms (Bertsekas, 1987) . The propagation delay is anyway fixed by the speed of the signalling network links; it is always one hop, in terms of direct exchange among neighbouring nodes. Information on nodes more than one hop away is implicitly contained in the neighbour's aggregated cost and, as such, reflects the delayed situation of a part of the network (with delay approximately given by the average number of hops that separate nodes in a given "region" from the one considered).

It is important to note that, in the simulation results, the updating time is given as a fraction of the reallocation time. For example, T=K/10 (or K/T=10). This timing method, which can be better understood by observing Figure 6, as said in the next Section, does not affect the generality of the algorithm (in fact every value of T can be set) and it has been chosen to simplify the software implementation.

4. SIMULATION RESULTS

The performance of the proposed routing scheme has been tested by simulation on a 22-node network. Three traffic classes, a "reallocation interval" $K = 8 \cdot 10^7$ cells and a channel transfer capability $C = 150$ Mbits/s, with a related slot duration $T_S = 2.83 \cdot 10^{-6}$ s (53 bytes/cell), have been used.

The quantities $N_a^{(h)}$ [Erlangs], h=1, 2, 3, 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 1. The parameters $\varepsilon^{(h)}$, $\delta^{(h)}$ and $D^{(h)}$ represent the cell-level QoS requirements, namely, the upper bound on cell loss probability (P_{loss}), the upper bound on the probability (P_{delay}) of exceeding a delay $D^{(h)}$, and the delay constraint, respectively.

We refer to the traffic flow generated by the above data as an overall offered load 1, with a fixed value $N_a^{(1)}=168$; $N_a^{(2)}=140$; $N_a^{(3)}=21$ (traffic reference); 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,..., 20, are considered the same in each node, that is $\alpha_i=\alpha$ and $\beta_i=\beta$, $\forall i$.

The topology of the network that has been used in the simulations is shown in Figure 2, and is composed of 22 nodes, only one of which (node 21) is a destination.

Table 1. 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: $\varepsilon^{(h)}$	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

This Section is dedicated to a comparison of the two strategies R-DLCP and RF-DLCP introduced above, with the DLCP strategy, already presented by the authors in previous works (Bolla, 1995), and, in case of a network failure, with a centralised Shortest Path Routing (SPR) strategy and with a Hot-Potato strategy, which has to be seen just as a superior limit. The SPR algorithm is implemented by using the metric proposed in (2), for each link. The advantages and the drawbacks of each strategy are analysed.

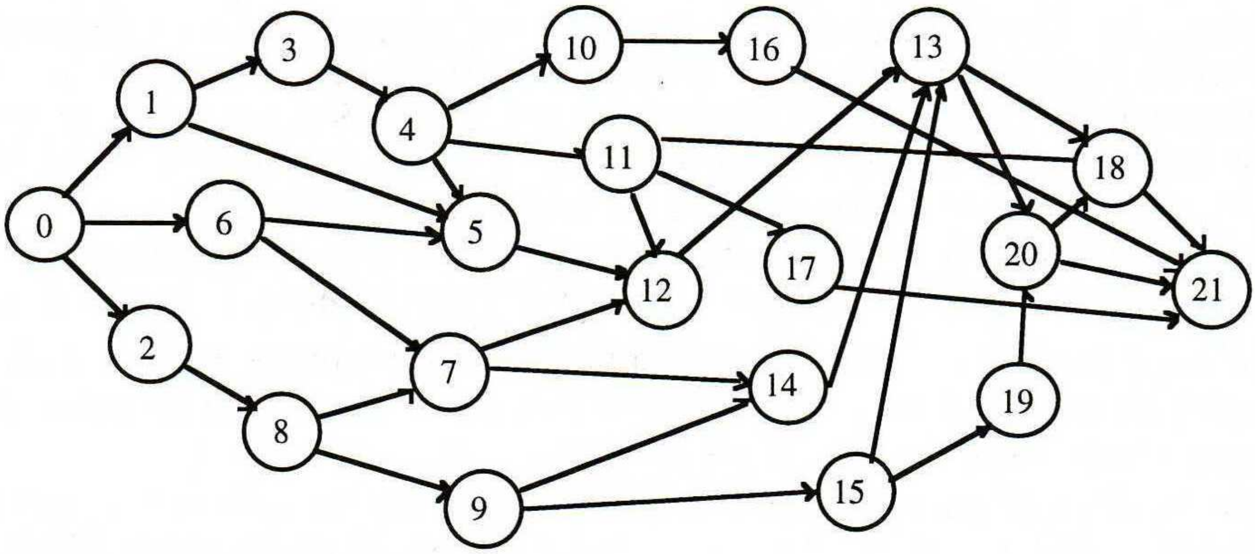

Figure 2. Topology of the test network.

The choice to have a single destination may seem a strong restriction; however, the topology chosen can be seen as a particular network "view" of node 21 and, as such, it allows a simple analysis of network parameters and a simple comparison among the presented routing strategies, which is the main object of this paper. Values (α=1, β=1) of the weighting coefficients and an updating time such that K/T=10 have been chosen for the simulations. Figure 11 shows the two quantities K and T, although with another aim, and it can be used to clarify the timing of the reallocation and of the updating. 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 quantity of interest.

Two load configurations are taken into account: the first one, called "distributed load", where each node generates traffic except for the selected destination (node 21, in our case); the second one, called "unbalanced load", where only nodes 0, 2 and 3, in Figure 2, generate traffic. These two configurations allow to test the proposed algorithm in two completely different situations.

The values chosen for α and β are the best ones for this topology. The simulative analysis shown in Figure 3 and in Figure 4 verifies that not too small values of α and β provide the best results; in fact, since in this network the average number of hops to get to the destination is relatively large (see Figure 7), the importance both of the local cost and of the aggregate cost is clear. This is not true if the extension of the network used is more limited. Figure 3 shows the percentage of blocked connections versus the weighting coefficient α, with a fixed value of β (supposed equal to 1); while Figure 4 shows the same quantity versus the coefficient β, with α=1. The best values of the two coefficients, as stated above, can be taken from the two graphs. A "distributed load" configuration is used to obtain these values.

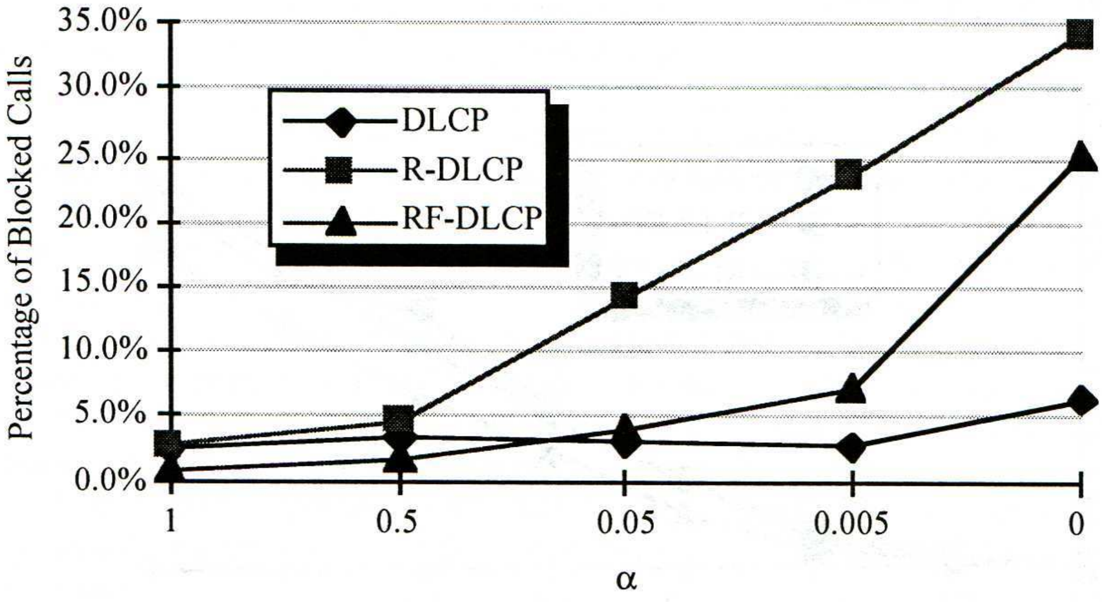

Figure 3. Total percentage of blocked calls versus the coefficient α (β=1).

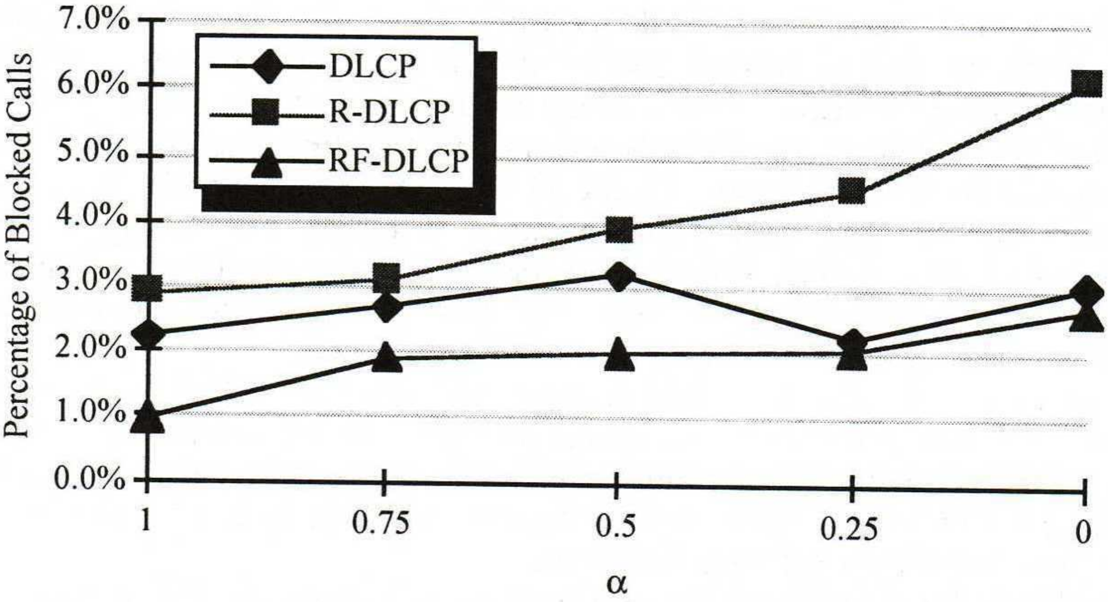

Figure 4. Total percentage of blocked calls versus the coefficient β (α=1).

A comparison among the three routing strategies is shown in Figure 5, where the percentage of blocked calls is depicted, in the "distributed load" configuration, versus the traffic load. It can be noted that the performance of the R-DLCP is not satisfying. This is due to the fact that the connection request, after the first blocking, goes towards the less saturated part of the network which is likely to be the same as in the first attempt; but the selection of the new link to follow from the LLN, in the second attempt, is performed, in fact, with the same metric (even if the link chosen the first time is discarded). So, concerning R-DLCP, we have a slight advantage only for a high load (1.8-2).

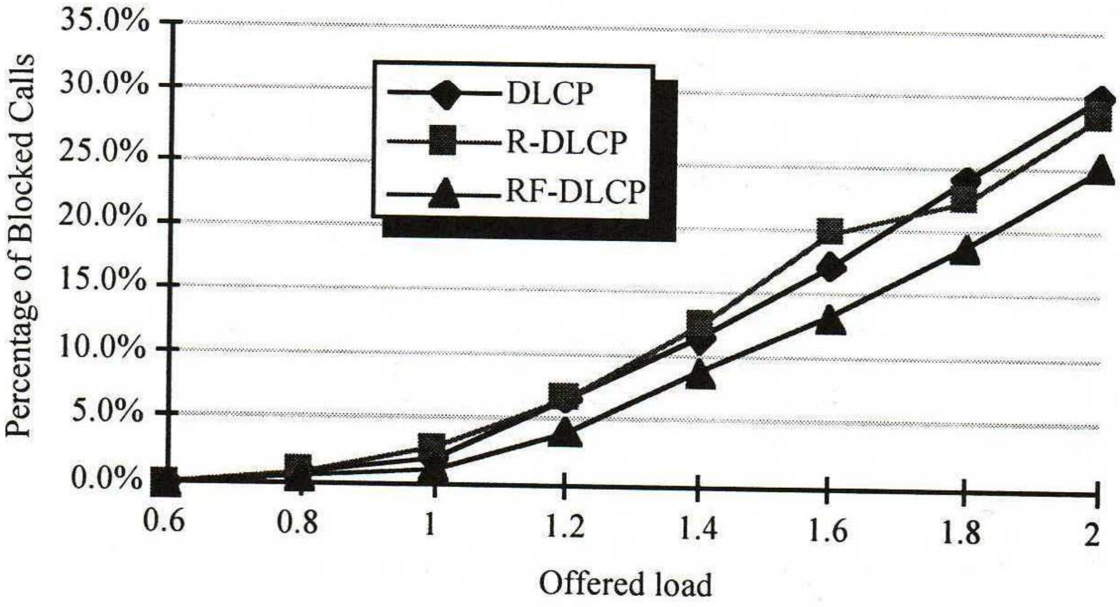

Figure 5. Total percentage of blocked calls versus the offered load, "distributed load".

On the contrary, the gain is clear for RF-DLCP, whose re-attempt mechanism has been designed just to skip the problem mentioned for R-DLCP. The gain of RF-DLCP, whose percentual values are given in Figure 6 versus the traffic load, taking the classic DLCP algorithm as a reference, is "paid" by an increase in the average number of hops necessary to get to the destination (shown in Figure 7 versus the traffic load).

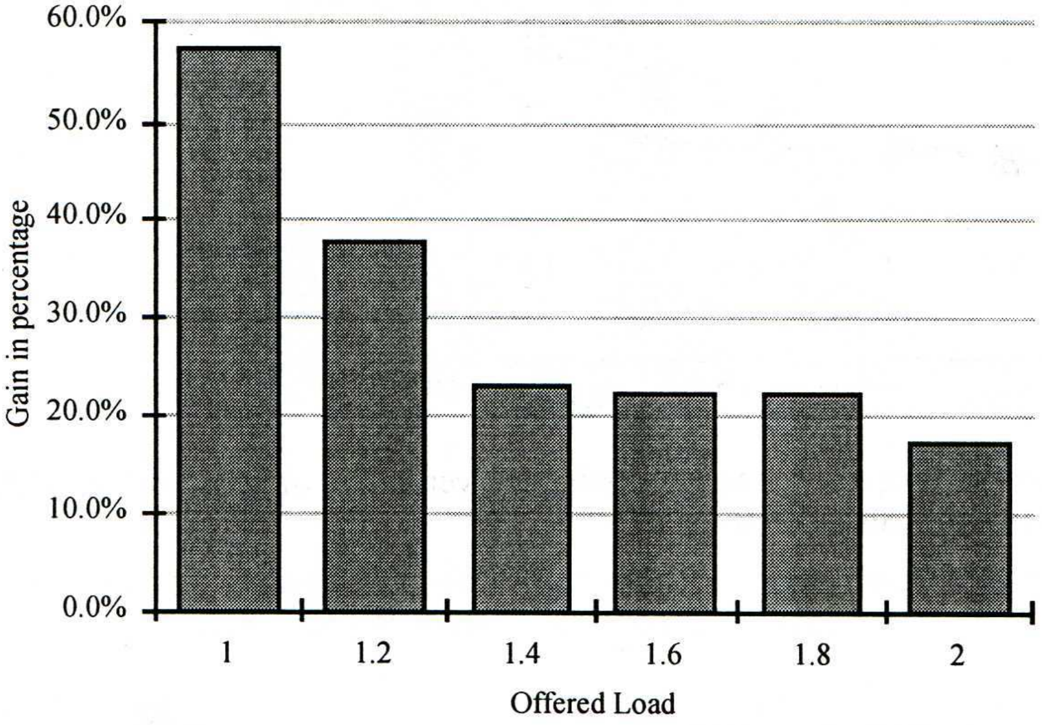

Figure 6. Gain percentage with respect to DLCP of the total blocked calls versus the offered load (RF-DLCP), "distributed load".

Anyway, it can be noted that the increase is very limited (less than one average hop) for most part of the load values shown, and that it becomes relevant only at very high load values. The evaluation of the re-attempt strategy gain and of the cost due to the increased number of hops may help the network manager to choose the more convenient strategy.

Similar observations can be made for Figure 8, where the percentage of blocked calls is depicted versus the traffic load in the "unbalanced load" situation. The difference with the distributed situation is that, in this case, even the R-DLCP gives a gain, whose percentage is shown in Figure 9, versus the traffic load, along with the RF-DLCP gain. It is important to see that, for very high traffic load, corresponding to a completely saturated network, the different strategies tend to have the same behaviour.

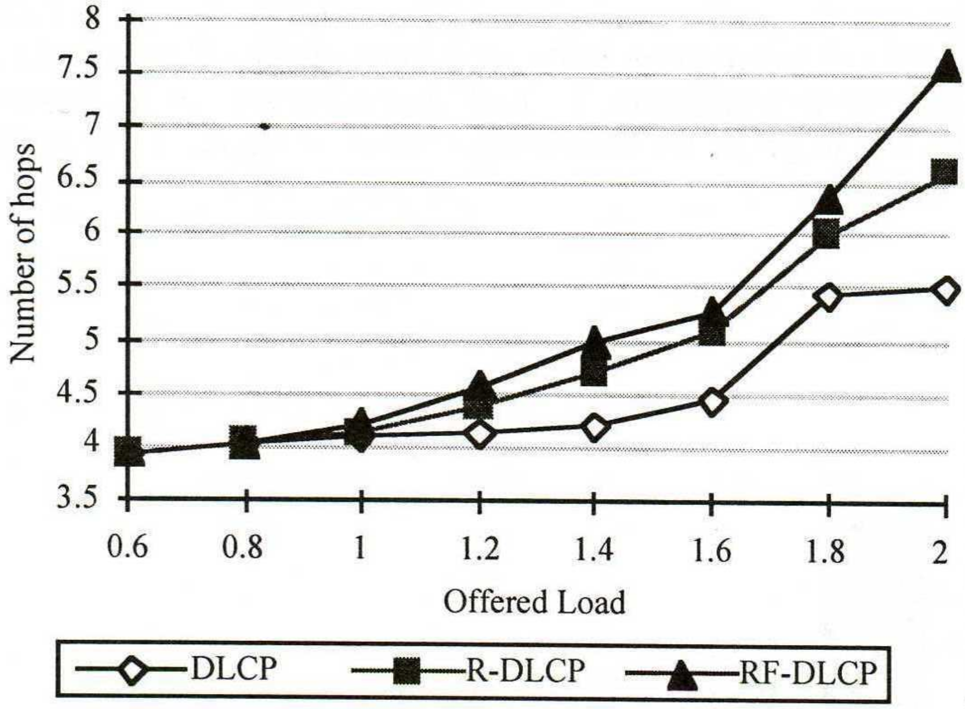

Figure 7. Number of hops to get to the destination versus the offered load (DLCP, R-DLCP, RF-DLCP), "distributed load".

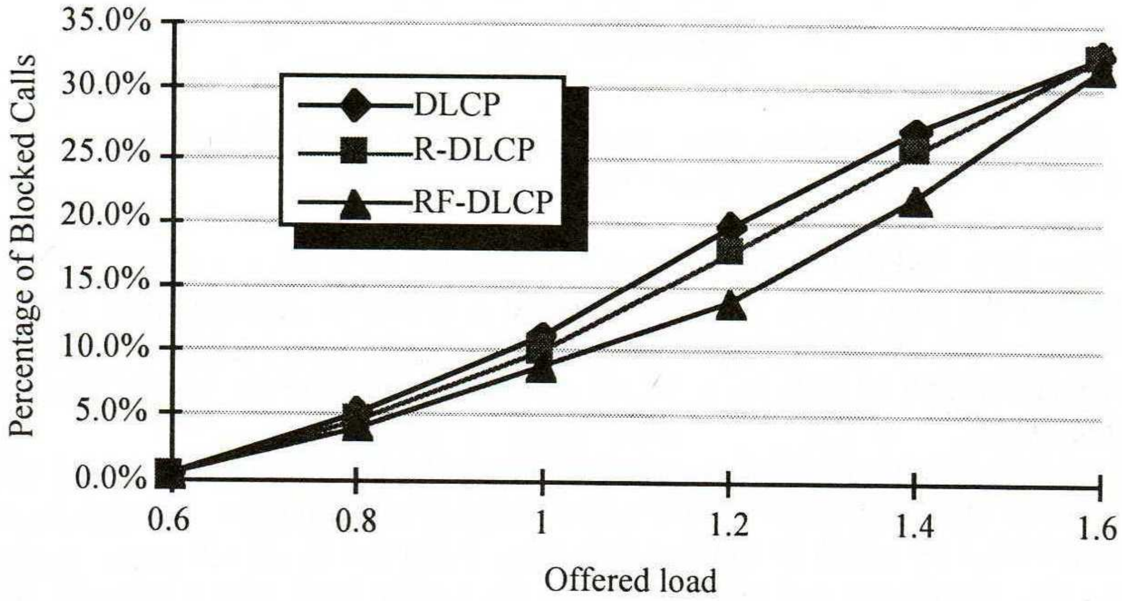

Figure 8. Total percentage of blocked calls versus the offered load, "unbalanced load".

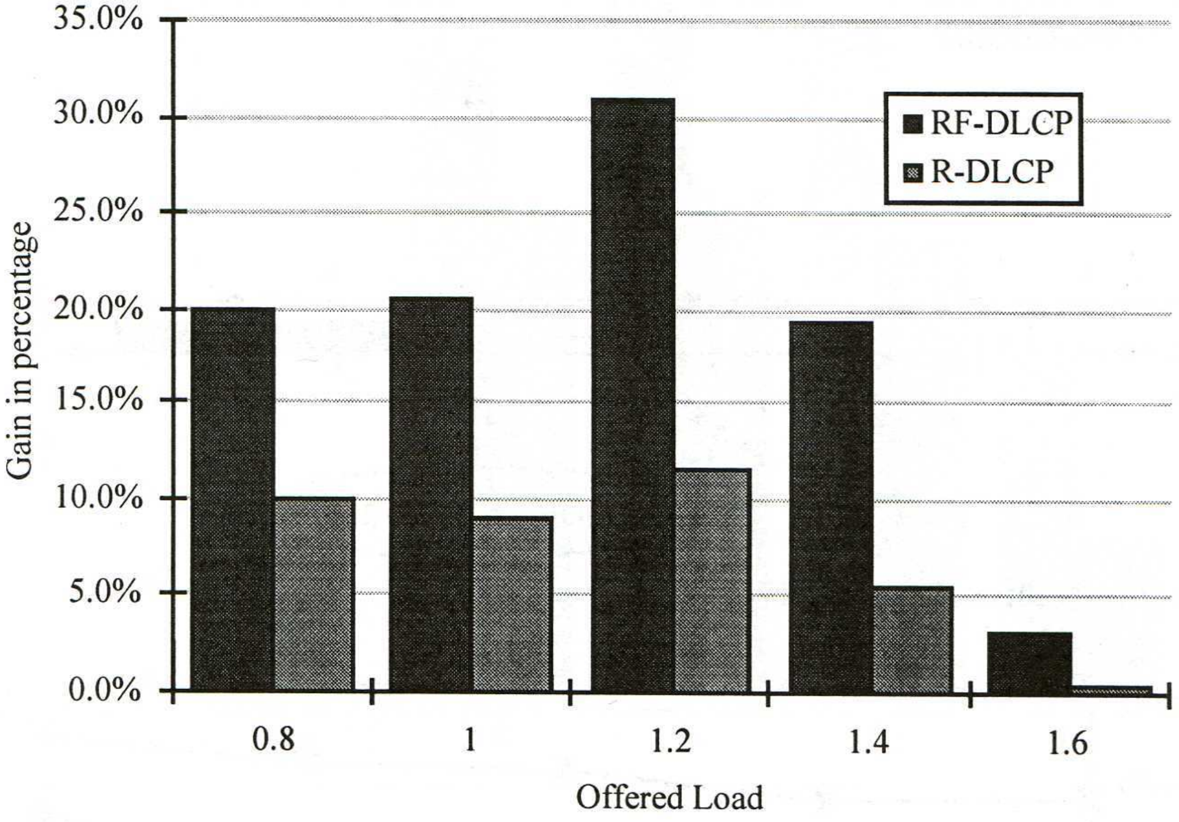

Figure 9. Gain percentage with respect to DLCP of the total blocked calls versus the offered load (RF-DLCP, R-DLCP), "unbalanced load".

The "unbalanced load" configuration is used to test the robustness of the algorithm in case of network failure (sub-section 2.3). The use of the two re-attempt strategies R-DLCP and RF-DLCP can be very useful when there is a node or a link failure because the DLCP routing 'realises' the new topological configuration at the first updating of the aggregate cost, while R-DLCP and RF-DLCP provide a prompt response to the new situation. In Figure 10, a change in the network topology is shown: just after the third "reallocation instant", link 17-21 fails. The black rectangle represents the period of time where link 17-21 is down. There is a period of time (before the following "updating time"), where the network does not know the failure has happened; clearly, if a more frequent updating time had been used, the difference among the evaluated strategies would have been less outstanding, but it should be observed that an updating equal to K/T=10 can be considered already quite short. The situation described above is used to obtain the next three graphs (Figure 11, Figure 12 and Figure 13).

The percentage of blocked calls versus the traffic load for the two presented routing algorithms is shown in Figure 11, along with the values obtained by using DLCP, SPR and Hot Potato. It can be simply noted that the gain of the two re-attempt strategies is more evident than in Figure 5; the re-attempt strategies are really efficient in case of network failure. The behaviour of SPR has to be regarded as a lower bound, because this strategy is centralised and can immediately detect the network failure, taking the proper actions. In Figure 12 the percentual difference of R-DLCP and RF-DLCP is depicted, with respect to the DLCP, versus the traffic load. The average number of hops to get to the destination is shown in Figure 13 for the failure situation.

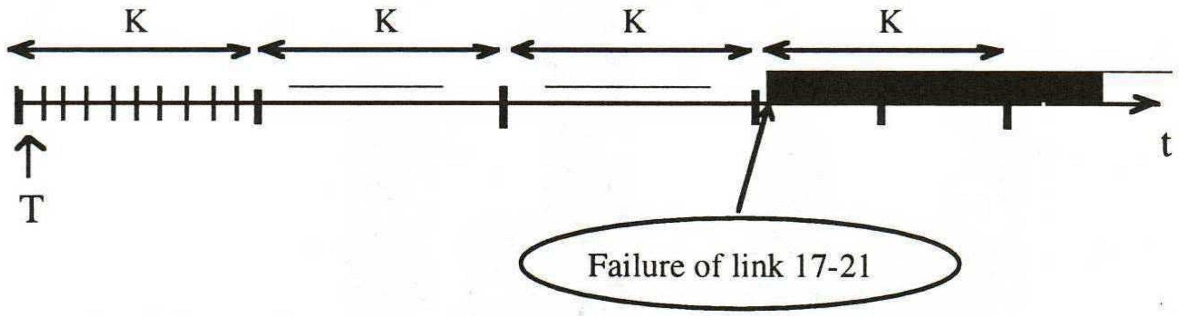

Figure 10. Failure of the link 17-21.

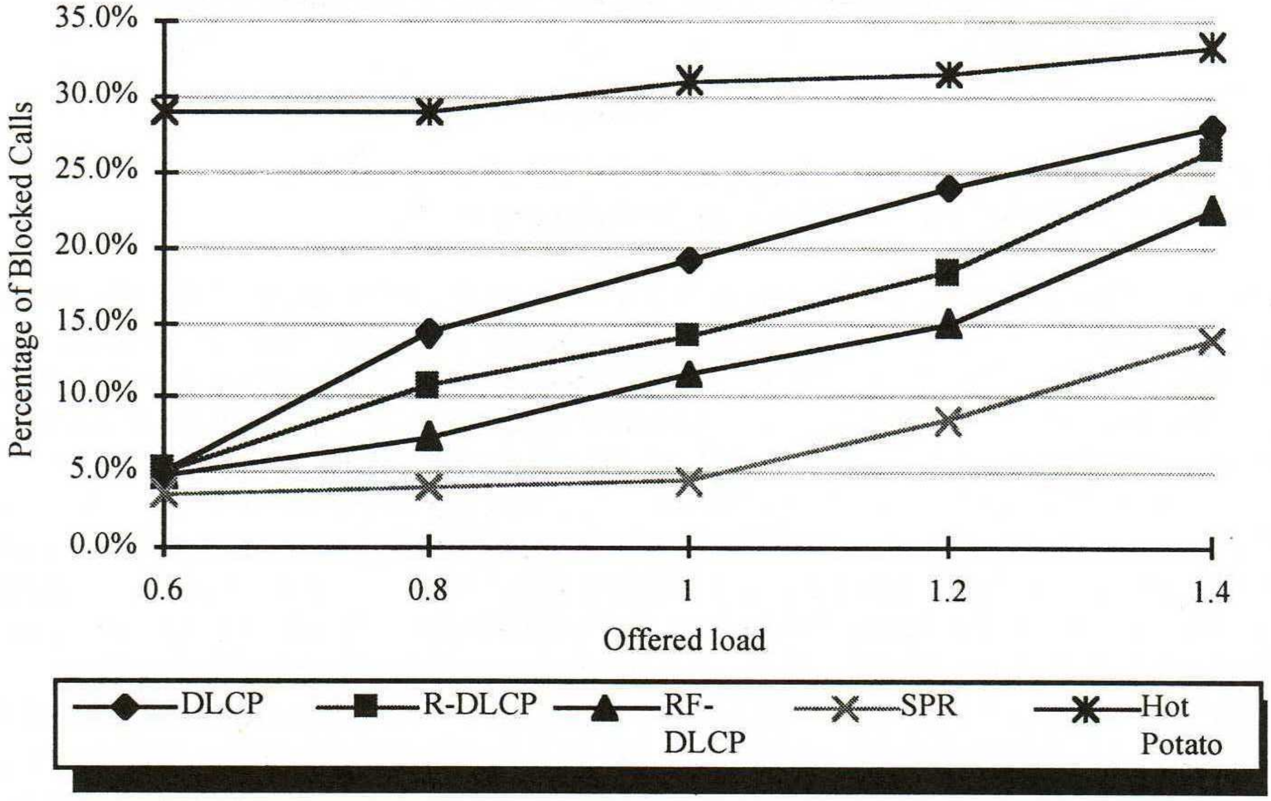

Figure 11. Total percentage of blocked calls versus the offered load (DLCP, R-DLCP, RF-DLCP, SPR, Hot Potato), "failure".

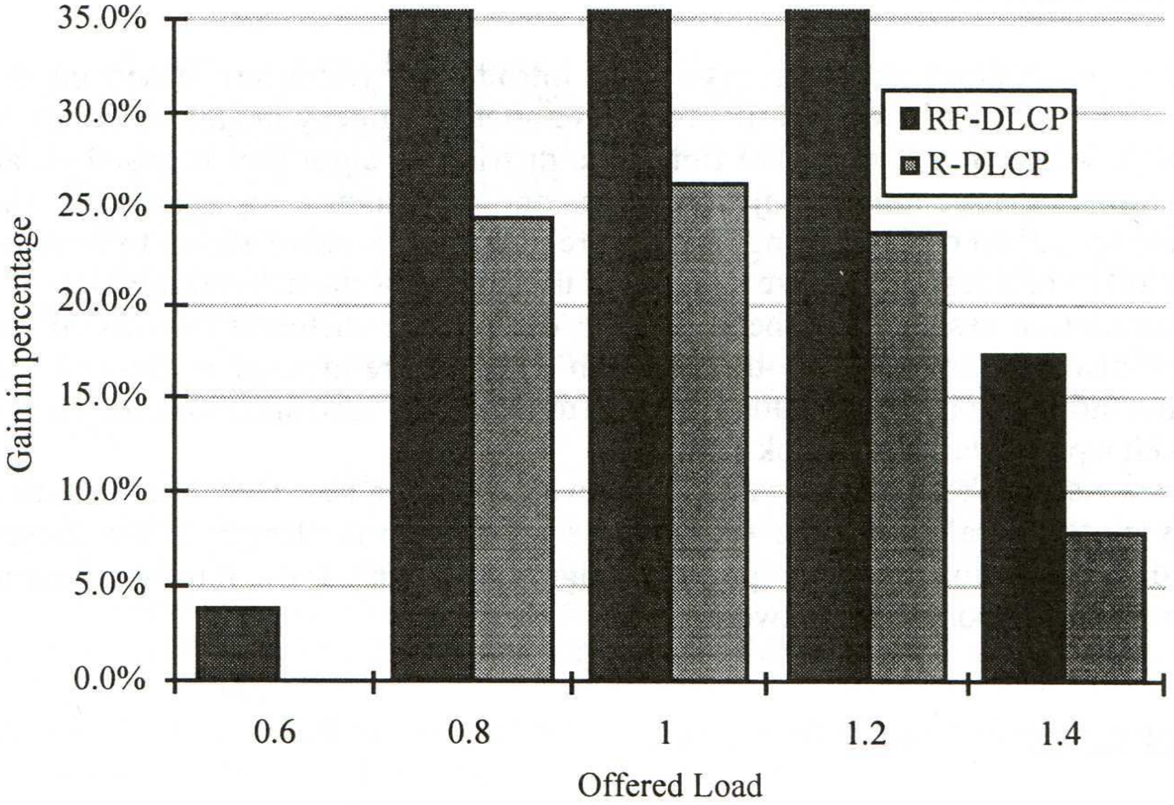

Figure 12. Gain percentage with respect to DLCP of the total blocked calls versus the offered load (RF-DLCP, R-DLCP), "failure".

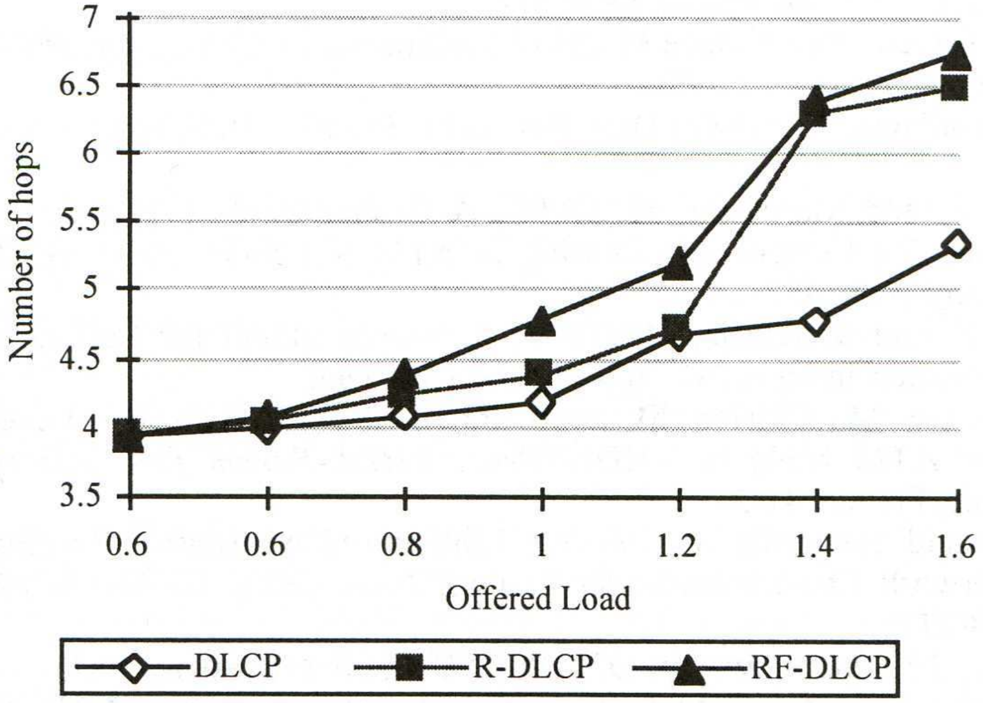

Figure 13. Number of hops to get to the destination versus the offered load (DLCP, R-DLCP, RF-DLCP), "failure".

5. CONCLUSIONS

Two dynamic routing schemes have been introduced, which are based on the combination of a distributed algorithm and a re-attempt strategy in case of failure to establish a connection for the first time. The distributed algorithm is based on an information exchange among adjacent nodes and on a metric represented by the "available space" on each outgoing link. The re-attempt procedure allows to increase the possibility of a faster response to changes in the load or the network topology.
Some simulation results have been used to discuss the different aspects of the schemes, also in comparison to the situation where no re-attempt is allowed and with other strategies in the literature. Two different traffic load situations are shown along with a particular case of link failure.
The future research may be dedicated to the evaluation of a new cost function based not only on the "available space" of the links but also on the length of the chosen path and to the evaluation of the proposed algorithms in case of a multidestination situation in a fully connected network.

REFERENCES

Arvidsson, A. (1995) High Level B-ISDN/ATM Traffic Management in Real Time, in *Performance Modeling and Evaluation of ATM Networks* (D.D. Kouvatsos, Ed.), Vol. **1**, pp 177-207.

Ash, G.R. (1991) Real-Time Network Routing in a Dynamic Class of Service Network. *Proc ITC 13*, Copenhagen, Denmark.

Begain, K. (1995) Discrete Performance Model of Multiplexer in ATM. *Proc. IEEE ICECS'95*, Amman, Jordan, 233-239.

Bertsekas, D. and Gallager, R. (1987) Data Networks. Prentice-Hall, Englewood Cliffs, NJ.

Bolla, R., Davoli, F. and Marchese, M. (1995) A Global Control System for Integrated Admission Control and Routing in ATM Networks. *Proc. IEEE Globecom'95*, Singapore, 437-443.

Bolla, R., Davoli, F. and Marchese, M. (1997) A Simple Model for Cell Loss Probability Evaluation in an ATM Multiplexer. This book.

Cornero, M., Marchese, M., Chirico, M. and Curatelli, F. (1994) System-Level Modeling of an ATM Node in VHDL. *Proc. VHDL-Forum for CAD in Europe*, Grenoble, France, 41-51.

Dighe, R., Ren, Q. and Sengupta, B. (1995) A Link Based Alternative Routing Scheme for Network Restoration under Failure. *Proc. IEEE Globecom'95*, Singapore, 2118-2123.

Hwang, R., Kurose, J.F. and Towsley, D. (1995) On-Call Processing Delay in High Speed Networks. *IEEE/ACM Transactions on Networking,* Vol. **3**, No. **6**, 628-636.

Medhi, D. (1995) Multi-Hour, Multi-Traffic Class Network Design for Virtual Path-Based Dynamically Reconfigurable Wide Area Networks. *IEEE/ACM Transactions on Networking*, Vol. **3**, No. **6**, 809-818.

Murakami, K. and Kim, H.S. (1996) Virtual Path Routing for Survivable ATM Networks. *IEEE/ACM Transactions on Networking,* Vol. **4**, No. **1**, 22-39.

Murthy, S. and Garcia-Luna Aceves, J.J. (1995) Dynamics of a Loop-Free Path-Finding Algorithm. *Proc. IEEE Globecom'95*, Singapore, 1347-1351.

Onvural, R.O. (1994) Asynchronous Transfer Mode Networks: Performance Issues. Artech House, Inc. Norwood, MA.

Pióro, M. and Gajowniczek, P. (1995) Stochastic Allocation of Virtual Paths to ATM Networks, in *Performance Modeling and Evaluation of ATM Networks* (D.D. Kouvatsos, Ed.), Vol. **1**, 135-146.

Plotkin, S. (1995) Competitive Routing of Virtual Circuits in ATM Networks. *IEEE Journal on Selected Areas in Communications,* Vol. **13**, No. **6**, 1128-1136.

Roberts, J.W. (Ed.) (1992) COST 224 - Performance Evaluation and Design of Multiservice Networks. Commission of the European Communities, Brussels.

Ross, K.W. (1995) Multiservice Loss Models For Broadband Telecommunication Networks. Springer-Verlag, London.

Saito, H. (1994) Teletraffic Technologies in ATM Networks. Artech House, Inc., Norwood, MA.

Shaikh, A., Lu, S. and Shin, K. (1995), Localized Multicast Routing, *Proc. IEEE Globecom'95*, pp. 1352-1356.

Sharma, S. and Viniotis, I. (1995) Conservation Laws for Dynamic Routing in Multi-path Networks, in *Performance Modeling and Evaluation of ATM Networks* (D.D. Kouvatsos, Ed.), Vol. **1,** 208-223.

BIOGRAPHIES

Raffaele Bolla was born in Savona, Italy, in 1963. He received the "laurea" degree in Electronic Engineering from the University of Genoa in 1989 and the Ph.D. degree in Telecommunications at the Department of Communications, Computer and Systems Science (DIST) of the University of Genoa, in 1994. From 1994 to 1996 he was a post-doc research fellow at DIST. Since November 1996 he is 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 B.Sc. 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; since 1994 he has been studying for the Ph.D. degree at the Department of Communications, Computer and Systems Science (DIST), University of Genoa, Italy. He is an IEEE member.

Franco Davoli was born in Genoa, Italy, in 1949. He received the "laurea" degree in Electronic Engineering from the University of Genoa in 1975. Since 1985 he has been Associate Professor and since 1990 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, Genoa, Italy, in 1992. He has concluded his Ph.D. in Telecommunications at the Department of Communications, Computer and Systems Science (DIST), University of Genoa. His research interests include traffic modelling, admission control and routing in ATM-based networks. He is a member of IEEE.