A DYNAMIC SELF-HEALING ROUTING STRATEGY FOR ATM NETWORKS

*R. Bolla, *A. Dalal'ah, *F. Davoli, *M. Marchese and **M.S. Obaidat

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

**Department of Computer Science and Center for Technology Development & Transfer
Monmouth University
W. Long Branch, NJ 07764, U.S.A.
obaidat@monmouth.edu

ABSTRACT

A hop-by-hop decentralized routing strategy for the ATM technology is considered, in which the traffic generated by different services is divided into classes, depending upon performance requirements. At each node traversed, all the outlets are considered and the least loaded one is chosen to carry the call, if it is capable to respect the Quality of Service requirements. The maximum number of calls for each class that would be carried over a link is limited. To accept a new connection, a cost function composed of the current values of a "local" cost and an "aggregate" cost, which is passed along each path periodically, is to be minimized at each node. Simulation results are presented and compared with those of other strategies.

I. INTRODUCTION

The topic of routing in ATM networks has received a great deal of attention for the last few years. Recently, it has been object of publications, both in books and journals or conference papers. In more detail, a chapter entirely dedicated to this subject can be found in [7], where the most used ATM routing algorithms are summarized and presented. A more mathematical approach can be seen in [8]. The routing problem is often a part of a global control mechanism in the ATM environment, i.e. a control system including the definition of traffic parameters and Quality of Service (QoS) requirements, whose satisfaction is assured by a CAC and bandwidth allocation scheme. In this context, a possible proposal is in [4], where a routing algorithm (called DLCP), which the scheme presented here is based on, is also investigated.

In the present paper, a routing scheme (named R-DLCP) is proposed, also capable of managing the failure of some components (nodes or links) in the network structure, is proposed. Some constraints, often present in other works, are removed. For example, the maximum number of hops from the source to the destination is often limited (as in [6], among many others, where two is the maximum number of hops allowed); or the homogeneity of the traffic flows is supposed (as in [2]). The concept of failure managing is not new, and some papers ([5], among the others) are dedicated to this topic. In this paper, an attempt is performed to modify an algorithm (presented in [4], where its efficiency was also tested, but not well suited to manage network failures) in order to obtain a self-healing algorithm.

Some simulation results are presented, aimed at showing the efficiency of the algorithm and the possible applications. As explained in the last Section, dedicated to the results, the analysis is limited to a single destination. This may seem a limitation; however, the specific network that is considered may be interpreted as a particular 'view' of a larger network, as seen by the destination node (in a sense similar to [1], even though the Virtual Path (VP) concept, used therein, is not explicitly mentioned here). The presence of multiple destinations would neither

0-7803-3873-1/97 $10.00 © 1997 IEEE

affect the structure of the algorithm, nor increase the information exchange among the nodes. The multicast problem (see [9], for a reference), is not considered in this work.

The paper is structured as follows. The features of the proposed strategy are underlined in Section 2. Section 3 is dedicated to the presentation of the routing scheme. Section 4 contains the results and Section 5 the conclusions.

II. CHARACTERISTICS OF THE PROPOSED MECHANISM

The traffic is divided into classes, characterized by statistical parameters (peak and average bandwidth) and performance requirements (bounds upon the cell loss and the delayed cell rate). These quantities are specified in Section 4; their complete explanation is not the object of this paper and can be found in [4]. The overall bandwidth of each channel outgoing from a network node is allocated among the various traffic classes. The allocation strategy works as follows: the partitions are assigned at periodic instants, by using a specific algorithm (see again [4]). The network behaviour is monitored (the number of connections in progress and of blocked calls is checked) and the allocation algorithm chooses the new partitions depending on the behaviour in the previous interval. The length of the reallocation interval (K slots) is chosen to obtain significant measures of the mentioned quantities. It is important to note that, during each reallocation interval, the bandwidth allocated to the traffic classes is fixed. In that sense, the mechanism can be considered a Complete Partitioning scheme [8], within a reallocation period, and a maximum number of acceptable calls (formally defined in the next Section) can be found for each class. A network link is considered 'free' if the number of calls in progress on that link is lower than the maximum number of acceptable calls; if the connections in progress equal the maximum value, the link is saturated and no connection can be routed on that path.

The routing strategy can be presented by using the concepts defined above. At connection set-up, a Resource Reservation Packet (RRP) is sent forward in order to choose the 'best' route from source to destination. The RRP verifies the status in each traversed node, i.e. it verifies if there are enough resources to accept the new connection. If there is at least a 'free' link towards the destination, a link is chosen and the packet is sent forward (a step is performed) and the necessary resources are reserved; the routing strategy used to choose the route is underlined in the next Section. Otherwise, if there is no available link, the strategy depends on the chosen algorithm: if DLCP is chosen, the connection is rejected and a Free Resource Packet (FRP) is sent back to free the resources already allocated. So, in this case, there is no possibility to find an alternative path. If R-DLCP (Re-attempt DLCP) is considered, an alternative route is looked for. The FRP goes back until the 'Least Loaded Node' (LLN) (whose definition will also be given in the next Section) along the path traversed (and, in this last case, memorized) by the RRP and starts again the routing algorithm from there. Theoretically, the number of re-attempts may be very large, but, for the sake of not saturating the network and limiting the response time at the connection set-up, the process of re-routing is limited to a single attempt. It is important to see that, with a similar philosophy as in [5], the RRP has two objectives, the first one being the purpose of establishing a route, and the second one being the use in the case of re-attempt.

The routing decisions are based on the minimization of a cost function which is composed by 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. The proposed scheme is completely distributed. 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. It was argued in [5] 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). The amount of information, in fact, does not increase if the network enlarges.

Concerning the robustness against network failures, an example should be of help. Fig. 2 depicts an arbitrary network, used for simulation and analysis purposes in Section 5. Suppose that the link between node 9 and node 11 has broken or node 9 has failed. In the DLCP algorithm the neighbours of node 9, after notification by the signaling protocol (in the case of node failure) or by node 9 itself (in the 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 [5]; however, in that approach, 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 contrary, 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 that sense, the R-DLCP better meets the robustness requirements, and it improved the efficiency of the previous scheme without a relevant load for the network, as will be seen in the part dedicated to the results.

III. THE ROUTING SCHEME

The quantities mentioned in the previous Section are formally defined here, and the routing strategy is completely defined in the following.

As already said, the algorithm applied at each node is based upon the computation of 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

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

where ij is the link with origin i and destination j and α_j is a weighting coefficient. 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 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 centralized algorithm in the same way.

Let $N_k^{(h,ij)}$ be the number of connections in progress for class h on link ij at time k and $N_{max}^{(h,ij)}$ be the maximum number of connections for class h at link ij that can be accepted, in the sense of the previous Section. Moreover, let m indicate the slot when a bandwidth re-allocation was performed, and K the duration (in slots) of the reallocation period. The local cost of link ij and class h at instant k is defined as

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

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). It has to be noted that 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 (theoretically infinite, but it could not be used in the aggregate cost defined below), when no more calls of class h can be accepted on that link.

The following formulation has been chosen for the aggregate cost:

$$W^{(h,j)}_{agg}(s) = W^{(h,j)}_{agg,node}(s) + \beta_j W^{(h,j)}_{agg,succ}(s) \quad (3)$$

where β_j is a weighting coefficient. $W^{(h,j)}_{agg,node}(s)$ represents the average situation of the node with respect to its congestion state, and $W^{(h,j)}_{agg,succ}(s)$ is an aggregate information on the average congestion of its successor nodes. More specifically, we have defined

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

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

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 $\hat{j} \in \ell^d$. If $W^{(h,i\hat{j})}(k) = Z$, the connection request is referred to the Least Loaded Node r_k along the path already done, and retries under the same conditions (but excluding the link that was previously chosen at r_k and led to node i), 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 traffic class considered and 'x' a generic node belonging to the mentioned path, (we have dropped the time index here for the sake of simplicity).

If the connection is accepted at r_k, say over link ks, 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 resources already allocated. Whenever a connection is accepted on link ij, $W^{(h,ij)}$ is updated, by adding 1 to $N^{(h,ij)}_k$, 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 was assumed whereby each node updates its aggregate cost after receiving the updated cost by its downstream neighbours. Anyway, even an asynchronous updating mechanism is feasible, as in the distributed shortest path algorithms [3].

IV. RESULTS

The parameter values of the three traffic classes used in the simulations are shown in Fig.1. 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 global average traffic intensities offered to the network are represented by the quantities $N^{(h)}_a$ [Erlangs], (h=1,2,3). The call arrival processes follow independent Poisson distributions. 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.

The topology of the network used in the simulations is shown in Fig. 2, and is composed of twelve nodes, only one of which (node 11) is a destination, as said in the Introduction. As already mentioned, 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 11 and, as such, it allows a simple analysis of network parameters and a simple comparison among the presented routing strategies, which is the main objective of this paper. By using this simple network, it is not necessary to implement a mechanism to avoid looping. Certainly, with larger and multi-destination networks the problem could be serious. A scheme taking into account even this problem will be the object of future research.

We refer to the traffic flow generated by the above data as an offered load 1, with fixed values $N_a^{(1)}=120$; $N_a^{(2)}=100$; $N_a^{(3)}=15$ (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,...,11, are considered to be the same at each node, that is $\alpha_i=\alpha$ and $\beta_i=\beta$, $\forall i$. The values ($\alpha=1$, $\beta=1$) of the weighting coefficients have been chosen for the simulations.

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

Figure 1. Parameter values.

The aggregate cost is updated twice in a reallocation period. 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. The α and β values are not the best ones for this topology. A simulative analysis, whose results have not been shown here, has verified that 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 short (see Fig. 5), the importance of the aggregate cost is rather decreased, and the network topology greatly enhances the importance of the "local" part of the cost. However the use of different values of α and β is reasonable if the network in Fig. 2 is considered to be a subnet of a larger one, where the choice of "optimal" values of the weighting coefficients for each node would be too difficult to manage, even if theoretically possible.

This Section is dedicated to a comparison between the strategies DLCP (in [4]) and R-DLCP (introduced above). The advantages and the drawbacks of each strategy are analyzed.

A comparison among the two routing strategies is shown versus the traffic load in Fig. 3. In Fig. 4, 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 (0.9 - 1.2) where the advantage of using R-DLCP can be noted. Even if, at a first glance, there would seem to be no particular difference between the two schemes, it has to be remembered that the variability of the presented values is really low, due to the high degree of confidence (3% confidence interval); so, we can conclude that the difference between the schemes really exists at certain loads. Furthermore the increased computational burden is not so higher and the time to get to know if a new connection has been accepted not so longer for R-DLCP with respect to DLCP. This sentence is justified by the values in Fig. 5, where the average number of hops to get to the destination is shown versus the traffic load for the two strategies.

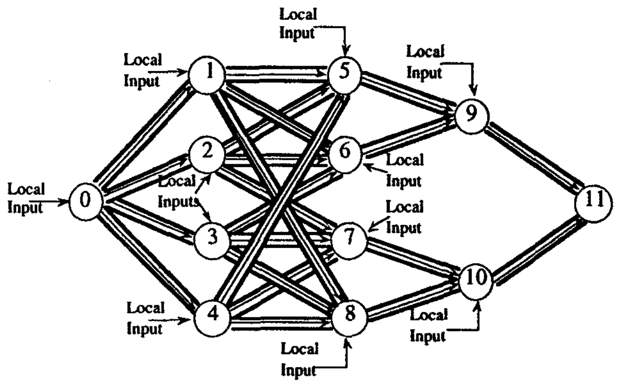

Figure 2. Topology of the test network.

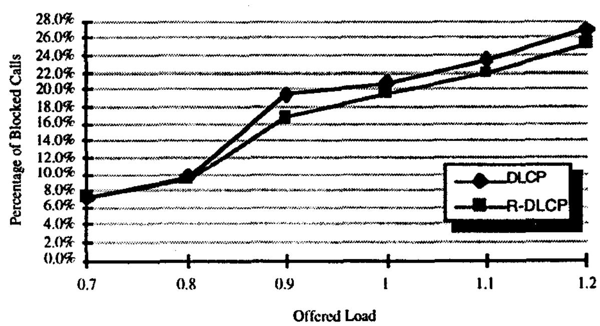

Figure 3. Total percentage of blocked calls versus the offered load (DLCP, R-DLCP).

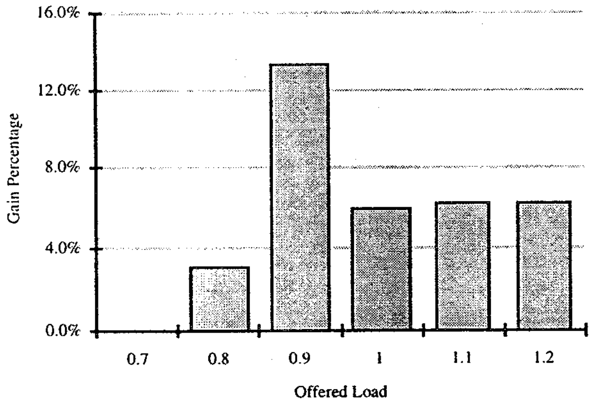

Figure 4. Gain percentage in the total blocked calls with respect to DLCP versus the offered load (R-DLCP).

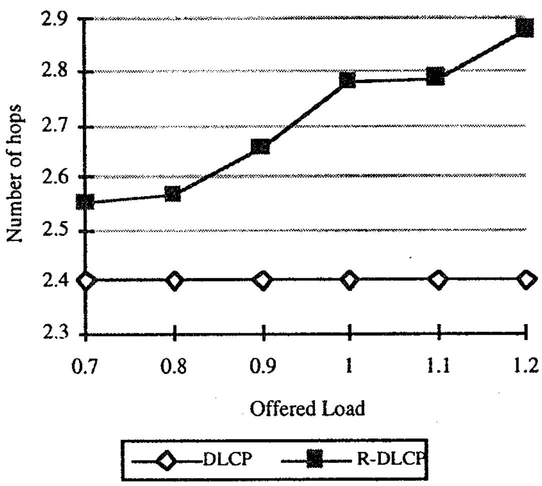

Figure 5. Number of hops to get the destination versus the offered load (DLCP, R-DLCP).

V. CONCLUSIONS

A dynamic routing strategy (called R-DLCP) has been introduced in this paper. The scheme is a modification of a previous one (DLCP), where no decision is taken in case of network failure. A specific self-healing procedure, in case of node or link failure, is elaborated here and compared with the previous algorithm.

The routing scheme is based on a 'local' (real time) information and on an 'aggregate' information, whose objective is to give an "indication" of the downstream network status. The overall scheme does not require a supervisory controller; in that sense it can be considered a completely distributed algorithm.

In the 'Results' Section, some comparisons with the DLCP algorithm have been performed. The R-DLCP has provrd convenient for average traffic load. The drawback (the increased number of hops to get to the destination) is not so remarkable.

REFERENCES

[1] C. Alaettinoglu, I. Matta and A. Shankar, "A Scalable Virtual Circuit Routing Scheme for ATM Networks", *Proc. IEEE ICCCN'95*, Las Vegas, Nevada, Sept. 1995, pp. 630-637.

[2] K. Begain, "Discrete Performance Model of Multiplexer in ATM", *Proc. IEEE ICECS'95*, Amman, Jordan, Dec. 1995, pp. 233-239.

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

[4] R. Bolla, F. Davoli and M. Marchese, "A Global Control System for Integrated Admission Control and Routing in ATM Networks", *Proc. IEEE Globecom'95*, Singapore, Nov. 1995, pp. 437-443.

[5] R.Dighe, Q. Ren and B. Sengupta, "A Link Based Alternative Routing Scheme for Network Restoration under Failure", *Proc. IEEE Globecom'95*, Singapore, Nov. 1995, pp. 2118-2123.

[6] S. Murthy and J.J Garcia-Luna Aceves, "Dynamics of a Loop-Free Path-Finding Algorithm", *Proc. IEEE Globecom'95*, Singapore, Nov. 1995, pp. 1347-1351.

[7] R.O. Onvural, *Asynchronous Transfer Mode Networks: Performance Issues*, Artech House, Inc. Norwood, MA, 1994.

[8] K.W. Ross, *Multiservice Loss Models For Broadband Telecommunication Networks*, Springer-Verlag, London, 1995.

[9] A. Shaikh, S. Lu and K.Shin, "Localized Multicast Routing", *Proc. IEEE Globecom'95*, pp. 1352-1356.