The Impact of Point-to-Multipoint Traffic Concentration on Multirate Networks Design

Aref Meddeb, *Member, IEEE*, André Girard, and Catherine Rosenberg, *Senior Member, IEEE*

Abstract—We consider the problem of multirate network design with point-to-multipoint communications. We give a mathematical formulation for this problem. Using approximations, we show that traffic concentration on a small number of links significantly reduces the cost of the network. We then propose a heuristic based on the traffic concentration principle to solve the network design problem approximately. Because this heuristic no longer requires advanced knowledge of demand matrices, we explain how it can be used as the basis for real-time design procedures. By means of numerical results, we show that this heuristic yields nearly optimal solutions.

Index Terms—ATM, IP, multicast, network design, routing, traffic concentration, trees.

I. INTRODUCTION

IN THE LAST decade, a wide variety of new applications has emerged, such as multimedia broadcasting, video conferencing, distant learning, video on demand, etc. All these applications require multicasting and the need for multipoint communications is increasing rapidly both in the Internet and in ATM networks.

The Multicast backbone (Mbone [1]) which was first introduced as an experimental backbone network to provide multicasting on top of the Internet is now widely used and is seemingly here to stay. Both the ATM Forum [2] and the International Telecommunication Union-Telecommunication (ITU-T) standardization sector [8] have introduced specifications for multipoint communications support.

Given the growing demand for multipoint communication services, the revenue generated by such connections is expected to become very important. Because of the large number and the diversity of resources that they require, Internet service providers (ISPs) will need to adapt the structure of their network and their connection procedures to the very different nature of multipoint connections. Unless they do so, the performance could turn out to be very poor and the revenue correspondingly

small. Just as for point-to-point networks [5], [4], [9], the networks and the real-time connection procedures should be designed keeping in mind the overall network performance and not simply from arbitrary rules applied to each connection independently of each other.

This paper describes techniques that might be useful to an ISP to build a minimum cost network that supports multicast. We assume that the service provider can lease capacity from some network provider in any required amount, e.g., in the form of ATM virtual paths (VPs). We do not consider the provisioning problem for the network provider since this is generally based more on marketing considerations than traffic. Also, the provisioning of real transmission systems, such as fiber optics, is done in bundles of large capacity, typically some number of optical channels and requires other models and solution techniques. As a consequence, we assume that switches are connected by bidirectional logical links such as VPs. In this paper, we consider only point-to-multipoint connections and we assume that a full multipoint connection can be realized by a set of point-to-multipoint connections as specified in [2]. Our objective here is to optimize the link capacities and the routing of connections in a given network.

For multipoint communications, four features stand out as important differences with point-to-point communications: 1) for a given multipoint connection, the resources of this connection are shared by more than two users; 2) members may join or leave a connection dynamically and independently from each other at any time during the communication's lifetime; 3) not all the participants need to be present at connection set up; and 4) most multipoint connections require a guaranteed quality of service (QoS) expressed as the packet (or cell) delay, loss, or jitter that may differ from one user of the connection to another (e.g., video conferencing and multimedia broadcasting where users use terminals with different speeds). It should then be no surprise that the efficient routing of multipoint connections will be substantially more difficult than for point-to-point calls.

In the case of point-to-point, there is an obviously good choice for the path connecting a given origin–destination pair. In a well-designed network, it is known that the direct link [3] carries most of the traffic. When this is not available, alternate paths can be used but they should be kept short since long paths are inherently inefficient [7], [9].

One specific difficulty is that, in the case of multipoint connections, there is no obviously good choice of tree that could carry most of the traffic. In [6], it was conjectured that small trees (in terms of number of links) such as spanning trees might

be the obvious choice, but as we will see in this paper, this is not true in general. In other words, we currently do not have an intuitive understanding of what constitutes a good routing in a multipoint network, at least with respect to the global network performance.

The paper is organized in two main parts, one based on the results of [15] and the other presenting new results. We describe the general framework of the model and formulate the network design problem in Section II. Then, in Section III, we explain the traffic concentration principle and we introduce the tree selection heuristic and in Section III-D, we give numerical results to test the validity of this heuristic and to illustrate the significant impact of traffic concentration on the cost of the network. New results are presented in Section IV, where we discuss how the solution of the dimensioning problem has to be modified to be able to accommodate unforeseen changes in the traffic demands. This also gives a measurement-based algorithm that no longer needs the explicit knowledge of demand matrices. Numerical results are presented to evaluate the merit of this approach. Finally, conclusions are presented in Section V.

II. NETWORK DESIGN PROBLEM

In this section, we formalize the discussion above, define notation, and state the mathematical model for the network design. Many multipoint communications of the future will likely have stringent QoS requirements at the packet or cell level. This can be achieved only by controlling both the access to the network and the allocation of resources to connections. This is much easier for some form of connection-oriented network and this is the assumption that we make here. The model we propose subsumes multirate circuit switched services, applies to ATM networks [6] through the equivalent bandwidth concept [8], and may also be relevant for IP-based networks using flow labeling [16] or resource reservation [17].

We also assume that we have some knowledge of the demands that will be made over some suitably long periods. More specifically, new connection requests arrive at random and the members of the group in a connection are known at the time of arrival. The connection is set up only if *all* destination switches can be reached. Also, we know in advance the *rate* at which each connection *type* arrives, where a type is defined by the source and destinations.

For each call type, we have a candidate set of trees on which to route the connections as described in [6]. We associate a probability to each tree in the candidate set. When a call request of that type is received, one of the trees is chosen randomly according to these probabilities and if the connection is not possible, the call is lost. This is not as efficient as other routing techniques such as alternate routing but we consider load sharing here because of its theoretical interest and tractability and because it can be used as the basis for more efficient real-time adaptive methods. The network design problem can be formulated as follows:

> *For a given set of source-destinations sets and their corresponding traffic demands, find the set of trees $\mathcal{S}$, link capacities $\mathbf{N}$, and the routing α that maximize the net value of the network.*

A. Model Assumptions

This paper addresses the design of the backbone network where we are given a set of N nodes where each node represents a multicast switch. All the switches have a multicast capability so that during the lifetime of a connection, *only one copy* of a data unit has to be sent to the end switches, regardless of the number of users present as long as there is at least one user connected to a given end switch and we can ignore the dynamic join and leave of group members.

There are C classes of calls where each class c has traffic descriptors used to compute the equivalent bandwidth $W(c)$ of each connection of that class. We denote the vector of equivalent bandwidths by $\mathbf{W}$. We use the same bandwidth for all the links involved in the connection setup for class c but different bandwidths, as in the case of effective bandwidth for ATM connections, can be easily handled by the model described in Section II-B.

There are K *types* of calls, each one corresponding to a source *node* and a *fixed* set of destination nodes. We also know the generalized *traffic demand* $A^k(c)$ that represents the ratio of the arrival rate to the service rate for calls of type k and class c and we assume that the arrival process for new calls is Poisson. In practice, it would be difficult to know these traffic demands but we make this assumption to make the problem tractable. One of the main results of Section IV-B is to provide a technique for updating the network links without having to know these matrices.

Finally, we are given $w^k(c)$, the revenue produced by a connection of type k and class c. Typically, $w^k(c)$ could depend on the number of destinations involved in call type k, on the geographical distance spanned by these destinations and also on the bandwidth $W(c)$. The special case $w^k(c) = 1 \ \forall k \ \forall c$ is equivalent to maximizing the network throughput. Here, the revenue generated by a network is the sum of the revenues produced by all accepted connections.

For each call type k of class c, we define $\mathcal{S}^k(c)$ as the candidate set of trees available to make the connections. In theory, this should be the set of all Steiner trees spanning the origin and destinations and this set can be very large for a given network. We also define $M^k(c) = |\mathcal{S}^k(c)|$ as the number of trees in $\mathcal{S}^k(c)$ and $\mathcal{S} = \cup_{k,c} \mathcal{S}^k(c)$ as the set of all candidate trees. We denote the mth tree belonging to $\mathcal{S}^k(c)$ by (k, c, m).

We also define the *load-sharing coefficients* $\alpha_m^k(c)$ as the fraction of calls of type k and class c offered to tree (k, c, m) [6]. For a given type, a call is accepted only if it can be connected from the source to *all* destinations. We make this assumption in order to simplify the model and it is not difficult to relax it by assuming that we can accept some calls with connections only to a given subset of the destination nodes.

B. Call Blocking Evaluation

Before we begin the description of the design algorithm, we should say a few words about the evaluation of the call blocking probability, the network performance measure used in the model. Let $L_m^k(c)$ be the probability that a call offered to tree (k, c, m) cannot be connected on that tree. We have $L_m^k(c) = 1 - Q_m^k(c)$ where $Q_m^k(c) = \prod_{s \in (k,c,m)} [1 - B_s(c)]$

is the call connection probability on tree (k, c, m) and $B_s(c)$ is the probability that link s is blocked. This is based on the independence hypothesis [6] for the blocking probabilities of the links of a tree. There is no proof for this hypothesis, not even in the point-to-point case, but we must use it to make the problem tractable.

For a given vector of load-sharing coefficients α, the call loss probability for calls of type k of class c is given by

$$L^k(c) = \sum_m \alpha_m^k(c) L_m^k(c). \tag{1}$$

In order to evaluate $L^k(c)$, we need to compute the link blocking probabilities $B_s(c)$ which are obtained by solving a system of nonlinear questions [6], [9]. Here, we assume that $B_s(c) = g_c(\mathbf{a}_s, \mathbf{W}, N_s)$ where g_c is given by the Labourdette approximation [11] and $\mathbf{a}_s$ is the vector of total traffic offered to link s [6]. Given the independence hypothesis, we can compute the $L_m^k(c)s$ by the same method even if the effective bandwidths of a given class c are different on different links since $B_s(c)$ depends only on the traffic vector and the capacity of link s.

C. Optimization Model

The routing and dimensioning optimization can be written as

$$\min_{\alpha, \mathbf{N}} z = \sum_s C_s N_s - \sum_{k,c,m} w^k(c) A^k(c) \times \alpha_m^k(c) Q_m^k(c) \tag{2}$$

$$\sum_m \alpha_m^k(c) = 1 \tag{3}$$

$$\alpha_m^k(c) \geq 0, \quad N_s \geq 0 \tag{4}$$

$$L^k(c) \leq \bar{L}^k(c) \tag{5}$$

where the objective function z given by (2) represents the net value of the network: this is the difference between the cost of the network and the revenue generated by the connections. N_s is the capacity installed on link s and the constant C_s gives the cost of one unit of capacity. Recall that we assumed that the capacities could be rented in any amount required so that the N variables are continuous. In the remainder of the paper, we will assume that $C_s = C_0 \, \forall s$ (we assume that the links are of the same type, fiber for example). $\bar{L}^k(c)$ is the maximum call loss probability for call type k and class c and defines the grade of service (GoS) for the network.[1]

D. Exact Solution

An appropriate method to solve problem (2)–(5) exactly on the full set of paths $\mathcal{S}$ would be to use the gradient projection with column generation as described in [13] and [15]. Because of the complexity of the tree selection problem, however, this method yields a very complex algorithm which cannot be used for any network of realistic size [13].

[1]For convenience, we use the term QoS for the measures of quality that apply to cells or packets such as delay, loss, jitter. GoS applies to connections and here is simply the probability that a connection will be rejected.

The obvious solution is then to limit from the outset the set $\mathcal{S}$ to some suitably chosen subset of trees.

The results of [6], [14] then raise the important question of how to select the candidate set of trees $\mathcal{S}^k(c)$ that are made available to connect the multipoint calls. This we call the *tree selection* problem. The actual *choice* of trees will be a result of the optimization through the values of the $\alpha_m^k(c)$ variables and this in turn will depend on the link *dimensioning*, also a result of the optimization. This is why we now restate problem (2)–(5) as

$$\min_{\alpha, \mathbf{N}, \mathcal{S}} z = \sum_s C_s N_s - \sum_{k,c,m} w^k(c) A^k(c) \times \alpha_m^k(c) Q_m^k(c)$$

$$\sum_m \alpha_m^k(c) = 1$$

$$\alpha_m^k(c) \geq 0, \quad N_s \geq 0$$

$$L^k(c) \leq \bar{L}^k(c)$$

where we have added the set $\mathcal{S}$ as another decision variable. Considering the importance of selecting good trees for networks of reasonable size, we propose in this paper a heuristic to select a nearly optimal set $\mathcal{S}$ in the context of network design. This heuristic is based on the traffic concentration principle described in [14].

III. TRAFFIC CONCENTRATION

For the network design problem, in all the numerical examples we have considered (about 50), we systematically found that at the optimal solution, *a single tree carries all the traffic of a given call type* and that the trees that carry the traffic of the different call types have *many links in common* [13]–[15]. We also found that the more we reduce the number of links in the network, the more the cost of the network is reduced. This result is not surprising and it is due to a very well known result: systems with large capacities have better marginal performance than systems with small capacities. Hence, traffic concentration covers two aspects.

1) The network should have as few links as possible.

2) Links should be able to carry as much traffic as possible.

In the multiclass case, we expect that traffic concentration will be economical in the asymptotic regime of large traffic and capacity where we know that multiclass calls behave as if the circuits are seized and liberated independently [11].

The conclusions given above are based solely on the traffic i.e., we assume that the cost is a linear function of capacity and therefore, we ignored this function in our discussion. In general, this function is concave which further increases the savings made by traffic concentration.

Finally, for networks with low value of GoS (less than 1% call loss probability), traffic concentration minimizes the cost of the network while it almost does not affect the revenue. In fact, when we optimize the dimensioning of the network, the call loss probabilities satisfy the GoS constraints and therefore, when $Q_m^k(c) \cong 1$, the revenue generated by the connections is almost constant, whatever the selected trees (numerical results also confirm this statement (see Section III-D).

A. Tree Selection Heuristic

Since we know that the solution is on a single tree, we now describe a heuristic algorithm [15] for selecting trees based on the principle of traffic concentration where: 1) we concentrate the traffic on a small number of links and 2) we install the capacity on the links that can carry as much traffic as possible.

Traffic concentration is done in two steps. The first is a global concentration where we consider all the call types simultaneously and where we want to make the trees generated for all the call types share as many links as possible. In the second step, we consider each type individually in order to have more traffic concentration where we want each tree of a given call type to use as few links as possible (see Section III-C).

B. Global Traffic Concentration

The global traffic concentration is performed by the following algorithm.

- For each call type k and class c, construct $\mathcal{E}^k(c)$, the set of nodes participating in this connection (source and destinations).
- For each call type k and class c, construct the set $\mathcal{L}^k(c)$ of all the two-way links (i, j) between all pairs of source and destination nodes of type k. If calls of type k and class c are connected using a tree composed of links exclusively in $\mathcal{L}^k(c)$, then we have a spanning tree for that call type.
- Construct an arc-type incidence matrix

$$\mathcal{J}_{s,(k,c)} = \begin{cases} 1, & \text{if link } s \in \mathcal{L}^k(c) \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

- Compute $\ell_s = \sum_{k,c} A^k(c)W(c)\mathcal{J}_{s,(k,c)}$, the total bandwidth requested between the two end-points of the link, for each link s.
- Using ℓ_s as link lengths, construct a maximum spanning tree over all call types.
- For each call type, construct the subtree connecting the destination and origin nodes. This is the tree that will be used to connect the calls.

The result given by this algorithm is a tree spanning all the nodes involved in at least one call type. For each call type, we have a subtree that uses exclusively links found in the tree. Once we have selected the trees, we optimize the link capacities using the dimensioning algorithm of [14].

We select the links with the largest values of ℓ_s which tends to minimize the number of links and thus maximize the traffic concentration. Based on the results of [15], we then expect to have a *Nearly Optimal Tree Network* (NOTN). In practice, the optimal network may not be a tree, especially when we have point-to-point call types (see Section III-C), but when we have exclusively point-to-multipoint call types, numerical results show that often, a tree network is optimal.

Since the network is a tree, its reliability is very low. Nevertheless, this is not a problem if we consider a network of logical links (ATM VPs). In case of a physical link failure, the VPs can be rerouted transparently over other physical links. Moreover, as we will see in Section III-C, when we have both point-to-point and point-to-multipoint calls in the network, the optimal network topology is not necessarily a tree. (The problem of network
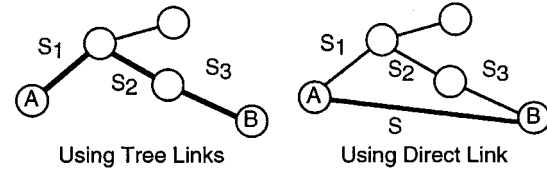

Fig. 1.   Using links in the NOTN versus the direct link.

reliability with multipoint communications is an open issue and is outside the scope of this paper.)

Finally, the global traffic concentration algorithm can produce more than one solution since there can be different maximum spanning trees depending on the order in which the links are considered in the tree generation algorithm. This introduces some flexibility in choosing the network topology. For example, while keeping optimality, we can reduce the degree of the nodes in order to minimize the number of copies generated by the switches or reduce the number of links in the branches of the tree in order to reduce the data units delivery delay from the source to the destinations.

C. Individual Traffic Concentration

When the algorithm described in Section III-B is used, a *small number* of call types are sometimes routed over large Steiner trees that do not use many of the links in $\mathcal{L}^k(c)$, with a resulting high network cost. To avoid these rather rare occurrences, we introduce the notion of *individual traffic concentration*. Basically, we keep the links of the NOTN and add some links to this network so that it becomes possible to generate a *small* tree for *each* call type. This often has to be done only for a small number of call types and the additional savings is rather small compared to the savings that were already made by means of the global traffic concentration.

The individual traffic concentration is more likely to be required for the call types that have a small number of destination nodes. A typical case is that of point-to-point calls. Assume that we have a network with many point-to-multipoint connections and a small number of point-to-point connections. If we use only the global traffic concentration, in the resulting tree network, some point-to-point connections may have to use paths involving several links which is likely to be costly in terms of capacity. A significant cost reduction can be realized if we use the direct link instead of using those several links.

An example is depicted in Fig. 1 where we have two cases. One case is where we use three links, s_1, s_2, and s_3, to connect point-to-point calls of type k from node A to node B (links s_1, s_2, and s_3 are within the NOTN). In the other case, we use the direct link s (link s corresponds to link (A, B) and is assumed not to be in the NOTN). The cost of the network is reduced by using the direct link s in the case where

$$N_{s_1} + N_{s_2} + N_{s_3} > N'_s + N'_{s_1} + N'_{s_2} + N'_{s_3} \tag{7}$$

where the primed variables correspond to the case where the traffic from A to B is sent over the direct link s, instead of the path composed of links s_1, s_2 and s_3. Inequality (7) is likely to be satisfied if the traffic demand from A to B is large enough so that we have $N'_{s_i} \ll N_{s_i} \; \forall i \in \{1, 2, 3\}$. However, if this

traffic demand is very large, the direct link s is very likely to be already in the NOTN which means that the individual traffic concentration is not needed.

Starting from the trees generated over the NOTN (we have one tree for each call type), the procedure used for the individual traffic concentration is as follows.

1) For all the unused links, set $\ell_s = -1$. Set $k = 0$.
2) $k = k + 1$, if $k = K + 1$, stop.
3) If, for call type k, the tree generated over the NOTN is not a spanning tree (i.e., we have a Steiner tree), generate a maximum spanning tree using ℓ_s as link lengths for this call type. Else, go to (2).
4) If, when using the spanning tree for call type k, the net value of the network is improved, replace the Steiner tree by the spanning tree for call type k. Go to (2).

The initial value of ℓ_s can be set to any arbitrary negative value in order to prioritize the selection of the links already in the NOTN. Once we have selected the trees, we optimize the link capacities using the dimensioning algorithm of [14]. The routing consists of using a single tree (often a tree using exclusively links of the NOTN) for each point-to-multipoint call and a single path (often the direct link) for each point-to-point call.

To summarize, dimensioning for point-to-multipoint connections yields a lightly connected network while dimensioning for point-to-point connections yields a densely connected network. If both types of traffic are mixed in the network, the optimal network topology can have any connectivity between that of a tree and that of a fully connected network. This depends on the ratio given by the number of the point-to-point call types to the number of the point-to-multipoint call types, on the number of nodes in each call type, on the relative intensity of the traffic (values of $W(c)$ and $A^k(c)$) and on the number of call types overall (see Section III-D.2 for numerical examples).

D. Numerical Results

We now present numerical results to show the impact of traffic concentration on the network performance and the validity of the proposed tree selection heuristic. We first evaluate the savings made by traffic concentration and then study the effect of the mixture of point-to-point and point-to-multipoint calls on the optimal topology of the network.

1) Traffic Concentration: In this section, we quantify the advantage of concentrating the traffic. We choose a star network as our reference since this is often used in practice because of its simple structure and compare its cost with that of an optimized network. To simplify the presentation, we consider a single class of calls and remove the class index.

We consider the nine-node network depicted in Fig. 2. This network represents all the possible links between the switches. Some of them will be allocated a capacity of 0 in the final solution and will not be present in the final network. Data corresponding to the call types are given in Table I. We have $K = 10$ and the w^ks were chosen such that each destination produces one unit of revenue. For each call type, we assume that we use a star to connect the calls. We then optimize the link capacities with the algorithm of [14] where we assume that $C_0 = 0.1$ and
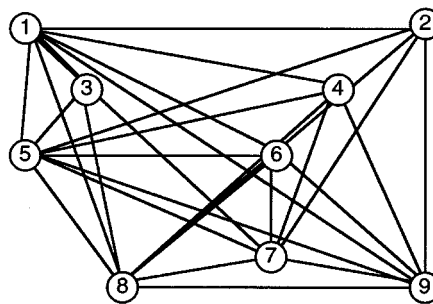

Fig. 2. Reference network topology: fully connected nine-node network.

TABLE I
REFERENCE DATA

k	source	destinations	w^k	A^k
1	1	{2,4,5,6,7,8}	6.00	4.00
2	2	{1,4,5,6,9}	5.00	3.00
3	3	{1,2,4,6,8,9}	6.00	4.00
4	4	{1,3,5,8,9}	5.00	5.00
5	5	{1,2,4,7,8,9}	6.00	5.00
6	6	{3,4,5,8,9}	5.00	2.00
7	7	{1,2,4,5,6,8,9}	7.00	6.00
8	8	{1,2,3,4,6,9}	6.00	5.00
9	9	{1,2,3,5,8}	5.00	3.00
10	5	{1,2,4,7,9}	5.00	1.00

TABLE II
GoS USING THE TRAFFIC CONCENTRATION ALGORITHM

k	1	2	3	4	5
L_0^k	.01	.01	.01	.01	.01
L_1^k	.007	.006	.007	.006	.007

k	6	7	8	9	10
L_0^k	.01	.01	.01	.01	.008
L_1^k	.007	.008	.007	.006	.006

$\bar{L}^k = 0.01$. The network cost is 50.0, the revenue is 217.8 and the net value of the network is $|z| = 167.8$. Recall that the net value is defined in (2) as the difference between the rental cost of VPs and the revenue generated by the carried requests. The actual values of the end-to-end GoS constraints, denoted L_0^k, are shown in Table II.

We now use the algorithm described in Section III-A to optimize the tree selection. The resulting network topology is shown in Fig. 3. The optimal network is a tree and we can easily find the subtree used to connect the calls of a given type and the nodes that should be connected for the considered call type from Fig. 3. The cost of the network is 34.8, the revenue is 218.5 and the net value of the network is 183.7. Hence, using the traffic concentration algorithm, we have reduced the cost of the original network by 30% while maintaining the revenue at the same value. Overall, we increased the net value of the network and improved the value of the the GoS constraints L_1^k as shown in Table II.

In order to test the "optimality" of the solution of Fig. 3, we have tried to improve it by exploring its neighborhood. A neighborhood of a solution is obtained by adding or modifying some
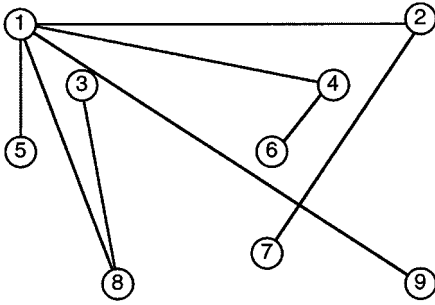
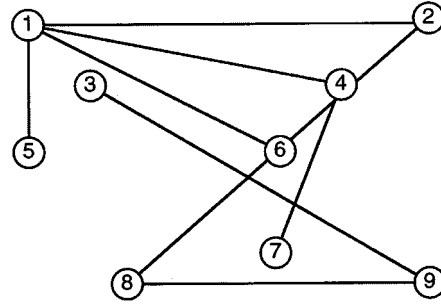Fig. 3. Optimal network topology using the traffic concentration algorithm.

TABLE III
PERFORMANCE OF THE NETWORK WHEN LINK s_1 IS REPLACED BY LINK s_2

Network	worst	random	best	NOTN
s_1	(1,2)	4 :(1,5)	20:(3,8)	–
s_2	(3,7)	17:(3,5)	2:(1,3)	–
$\ell_{s_1} - \ell_{s_2}$	31	17	2	–
Cost	39.5	37.7	34.8	34.8
Revenue	218.4	218.4	218.5	218.5
z	178.9	180.7	183.7	183.7
$\max_k L^k$	.008	.009	.008	.007

TABLE IV
ADDITIONAL POINT-TO-POINT CALL TYPES [(o, d): (ORIGIN, DESTINATION)]

k	11	12	13	14	15
(o,d)	(7,3)	(2,4)	(1,6)	(8,9)	(4,7)
A^k	5.0	4.0	2.0	4.0	3.0
k	16	17	18	19	20
(o,d)	(6,1)	(2,8)	(3,9)	(3,4)	(2,1)
A^k	4.0	5.0	3.0	5.0	4.0

links to the network. We have noticed that the larger the number of links modified or added to the NOTN, the higher the cost of the resulting network and for this reason we have limited the neighborhood to the modification of a few links only. We can see from the results in Table III that the networks in the neighborhood are from 0% to 14% more expensive than the NOTN while they have practically the same revenue and GoS. We can also see that when we replace link s_1 by link s_2, the cost of the resulting network increases by $\ell_{s_1} - \ell_{s_2}$ which illustrates the near optimality of traffic concentration. The fact that the best solution given in Table III is not the original NOTN but has the same net value suggests that the NOTN might not be far from a local optimal solution.

2) Mixing Point-to-Multipoint and Point-to-Point Calls: In this section, we want to evaluate the impact on the optimal network topology of having point-to-point as well as point-to-multipoint calls. Also, we want to illustrate the usefulness of the individual traffic concentration. For this purpose, we introduce ten additional point-to-point call types (see Table IV) while keeping the point-to-multipoint call types we already introduced in Table I. We assume that each point-to-point connection generates one unit of revenue.

Without using individual traffic concentration, we optimize the tree selection and the network dimensioning. The NOTN

Fig. 4. Topologies with many point-to-point call types, without individual concentration.
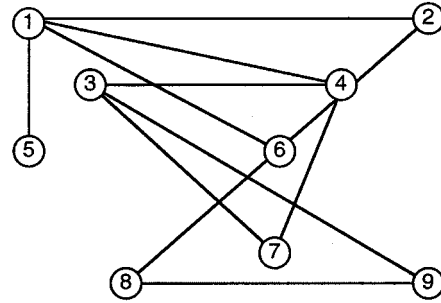

Fig. 5. Topologies with many point-to-point call types, with individual concentration.

topology is given in Fig. 4 and we see that it has changed compared to the one given in Fig. 3. The cost of the network is now 46.5, the revenue is 257.2 and the net value of the network is 210.7. This is to be compared with the star network dimensioned for these new traffic values. Traffic concentration manages both to decrease the network cost from 56 to 46.5 while at the same time increasing the revenue from 256.7 to 257.2.

Next, we use the individual traffic concentration on that network. The resulting network topology is obtained by adding three links to the NOTN of Fig. 4 giving the network of 11 links of Fig. 5. These are the direct links $(2, 4)$, $(3, 4)$ and $(3, 7)$ corresponding to the point-to-point call types 12, 19, and 11, respectively. Hence, the individual traffic concentration was required only for these three call types. For the other point-to-point call types, the direct links were already in the NOTN.

The cost of the network is 42.2, the revenue is 257.2 and the net value of the network is 215.0. Hence, the network cost reduction when the individual traffic concentration is used is about 9% compared to the case where we make only the global traffic concentration. This is summarized in Table V where we can see that even when we have many point-to-point call types, the traffic concentration yields a significant network cost reduction while increasing slightly the revenue.

E. Multiclass Networks

In this section, we evaluate the savings made by the traffic concentration heuristic on more complex cases. We consider the ratio of the number of point-to-point to point-to-multipoint call types, which we define as ρ, the size of the network N (number of nodes), the number of call types K and the number of classes C. As before, we consider a reference case where the calls are connected by a star. We then define η as the total cost reduction

TABLE V
NETWORK DESIGN WITH POINT-TO-POINT AND POINT-TO-MULTIPOINT
CALLS. Nr IS THE REFERENCE NETWORK, Ng WITH GLOBAL TRAFFIC
CONCENTRATION AND Ni WITH INDIVIDUAL TRAFFIC CONCENTRATION

	Cost	Revenue	z	$\max_k L^k$
Nr	56.0	256.7	200.7	.010
Ng	46.5	257.2	210.7	.009
Ni	42.2	257.2	215.0	.010

TABLE VI
% SAVINGS. MIX 1: NO PT-TO-PT, ONE PT-TO-MPT. MIX 2: 1 PT-TO-PT AND 2
PT-TO-MPT. MIX 3: 2 PT-TO-PT AND 2 PT-TO-MPT

Mix	No Nodes	$K = 10$	$K = 15$	$K = 20$
	9	30.6	23	23.3
1	16	31.3	29.8	31
	25	36	40.4	46

Mix	No Nodes	$K = 20$	$K = 30$	$K = 40$
	9	24.6	21.3	15.8
2	16	29.1	26.5	25
	25	33.7	37.6	38.3

Mix	No Nodes	$K = 21$	$K = 30$	$K = 39$
	9	22.2	21	15.4
3	16	26.6	25.1	23.3
	25	30	21.2	33.5

(in percent) obtained when using the traffic concentration algorithm (global plus individual).

We compare in Table VI the cost of the optimized network to the cost of the reference network for networks with different sizes and traffic demand patterns. We can see that traffic concentration yields a significant cost reduction which increases with the size of the network and we have 45% savings for $N = 25$, $K = 20$ and $\rho = 0$ (no point-to-point calls). Also, we can see that η increases when ρ decreases and this is because when ρ is small, we make a significant cost reduction by concentrating the traffic for the large number of point-to-multipoint call types. In addition, η is often larger than 20% except for the cases where we have many point-to-point calls (ρ and K are large) and this, independently of the size of the network and the number of call types. In these cases, using the direct link is optimal for most of the call types, the global traffic concentration does not bring a significant cost reduction and we have a rather small cost reduction by means of the individual traffic concentration. This is illustrated in Table VI for $N = 9$, $K = 40$ and $\rho = 1/2$ and for $N = 9$, $K = 39$ and $\rho = 2/3$ where η is about 15%.

Next we consider some nine-node networks with 15 call types. We give three cases with, respectively one, two, and three classes. Class 1 has $W(1) = 1$, class 2 has $W(2) = 2$ and class 3 has $W(3) = 4$. In each case, we consider two networks with $\rho = 1/3$ and $\rho = 2/3$. We assume that the point-to-multipoint traffic requires more bandwidth than the point-to-point traffic because in practice, point-to-multipoint traffic usually involves video and multimedia traffic while most of the point-to-point traffic is data and voice.

As we can see from these results of Table VII, the cost reduction with respect to the star network made by traffic concentration is significant also in the multiclass case. In addition, the cost reduction increases with the bandwidth in the case where

TABLE VII
SAVINGS DUE TO TRAFFIC CONCENTRATION (η IN %) FOR MULTICLASS
NETWORKS

ρ	One class	Two classes	Three classes
1/3	10.0	28.9	30.0
2/3	23.3	22.9	23.9

ρ is small which is due to the important savings made by the global traffic concentration for the large number of high bandwidth point-to-multipoint calls.

IV. DIMENSIONING AND REAL-TIME DEMANDS

We have shown that when dimensioning the network for a single set of point-to-multipoint demands, concentrating the traffic on a small number of links reduces significantly the cost of the network. In practice, the actual demands will often differ from these forecasted values. In that situation, the network operator can: 1) do nothing; 2) reoptimize the routing; 3) update the link capacities (since the links are ATM VPs); or 4) reoptimize both the network topology and the link capacities. The question is then which of these strategies is best.

In the second case, the network is fixed and we can only change the call routing to maintain an acceptable performance. The results of [15] show that this is possible only if we have a highly connected network in the first place. The question is then what happens in the case where we have a very lightly connected network (a tree) produced by the dimensioning algorithm and if we want to reoptimize the routing because of some traffic change.

When point-to-point traffic shares the same links with point-to-multipoint traffic, and when both types of traffic have comparable bandwidth requirements, the network topology is rather highly connected and routing changes can be made to adapt to the changed demands. The problem occurs only when we have mostly point-to-multipoint call types or when they require a higher bandwidth than the point-to-point calls.

In that case, we must somehow increase the network connectivity in order to make the call routing optimization possible even though this is suboptimal with respect to the solution of the dimensioning based on the nominal demand matrices. One possibility is to modify the model by including a constraint on the connectivity of the network, which will likely yield a complex solution. Another possibility is to select disjoint trees for the call types which is totally in contradiction with the traffic concentration principle. In practice, this can be very costly and as we have seen in Section III-E, the cost of the resulting network can be up to 45% more expensive than the one obtained by traffic concentration.

A. Connectivity Increase Algorithm

A more economical way to generate networks with increased connectivity is to add some suitably chosen links to the NOTN. In order to be able to change the call routing, the links should be selected such that they can be used to connect pairs of nodes participating in as many call types as possible; otherwise, we may have a limited choice to select new trees. We can use the ℓ_s defined in Section III-B to select these links.

Let $h(n)$ be the degree of node n. Suppose that we have generated the NOTN by the algorithm described in Section III-B. We can increase the degree of the nodes by selecting the *links* involving the pair of nodes that have both a small $h(n)$ and a large ℓ_s. This can be done by the following algorithm.

1) Set $h_0 = 1$.
2) Mark all the links corresponding to the pair of nodes (i, j) such that $h(i) = h_0$ *and* $h(j) = h_0$. If there is no such link, go to 3. Else go to 4.
3) Mark all the links corresponding to the pair of nodes (i, j) such that $h(i) = h_0$ *or* $h(j) = h_0$. If there is no such link, go to 5.
4) Select the marked link (a, b) with the largest ℓ_s that is not already selected and add it to the network. Set $h(a) \leftarrow h(a) + 1$ and $h(b) \leftarrow h(b) + 1$. Unmark all links and go to 2.
5) Set $h_0 \leftarrow h_0 + 1$, if $h_0 > \overline{h}$ stop, else go to 2.

$\overline{h}$ is the minimum degree required for each node in the network which we can fix depending on the connectivity we want, and h_0 is the iteration counter.

Once the topology of the network has been modified to increase its connectivity, there still remains the question of what capacities should be allocated to all the selected links, including those in the NOTN. We want to reduce the cost of the network by concentrating the traffic as much as possible but also avoid ending up with a tree since this would severely limit the possibility of rerouting in the presence of error forecast on the traffic demands. In practice, for each call type, we avoid selecting trees that use exclusively links in the NOTN since if we do so, the optimal dimensioning would select only those trees. Instead, we select trees where the traffic offered to all the links should be roughly equal. This of course is suboptimal since we have seen that the least costly topology is the tree but we need to do this to allow some possibility of real-time rerouting.

We use the algorithm of [15] to select disjoint trees with a predefined set of links. We choose a single tree for each call type and then we optimize the link capacities by means of the algorithm of [14].

B. Real-Time Heuristic

One weak point of the model presented here is that it requires knowledge of the point-to-multipoint demand matrices A^k. This is unrealistic since the number of such matrices will grow very fast with the network size. The adjustment techniques that we discussed can be adapted to work without a knowledge of the demand matrices.

By measuring the number of blocked connections on the links, the service provider can decide to increase the capacity of links with high blocking and to decrease it when it is low.

The other modification is a change in the network topology. This is based on the values of ℓ_s which estimate the link utilization. They can be calculated from the demand matrices when these are known but it is also possible to estimate them when this is not the case. We count at each switch the number of new connection requests and record their destination sets to determine the type. Inside the network, we also count the number of blocked calls on each link. These measurements can then be
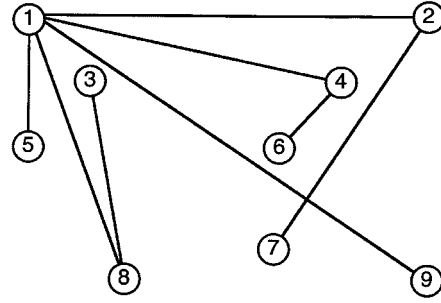

Fig. 6.   Increasing the network connectivity, original network.

TABLE VIII
FORECAST ERRORS: ACTUAL DESTINATIONS AND TRAFFIC DEMANDS. $A+$ SIGN IS A NEW DESTINATION, $A-$ SIGN IS A NONEXISTENT DESTINATION. A_f^k IS THE FORECAST, A_a^k THE ACTUAL TRAFFIC DEMAND

k	source	forecasted dest.	A_f^k	A_a^k
1	1	$\{2,4,5,6,-7,+8\}$	4	2
2	2	$\{1,4,5,6,+8,9\}$	3	2
3	3	$\{1,2,4,6,+7,8,9\}$	4	5
4	4	$\{1,+2,3,5,+6,8,9\}$	5	4
5	5	$\{1,2,+3,4,7,8,9\}$	5	7
6	6	$\{3,4,5,8,9\}$	2	4
7	7	$\{1,-2,4,5,6,8,9\}$	6	6
8	8	$\{1,2,-3,4,6,-9\}$	5	4
9	9	$\{1,2,-3,5,8\}$	3	2
10	5	$\{1,2,4,+6,7,9\}$	1	2

used to estimate a new value of the ℓ_s. If they indicate that the offered traffic has changed significantly, the network operator can react by updating the topology or the routing as described above.

C. Numerical Results

In this section, we compare through numerical results the topology update and the link upgrade for handling real-time demands. We consider the case where we have point-to-multipoint call types only.

Suppose now that we had forecasted the traffic of Table I and built the corresponding NOTN given in Fig. 3 which is also given in Fig. 6. As we can see in this network, we have $h(1) = 5$ and $h(n) = 1 \ \forall n \in \{3, 5, 6, 7, 9\}$. Suppose now that the actual traffic demands and destination sets are the ones described in Table VIII.

The simplest way to react to this unforeseen traffic is to keep the same topology and compute a new set of trees by the algorithm of [15] on which to route the demands. The results are summarized on the first row of Table IX where we can see that the GoS has substantially degraded. This is not unexpected since the topology is too sparse to be able to produce good trees for these new demands.

Next, we try to increase the connectivity of the network by the algorithm of Section IV-A. When we set $\overline{h} = 2$, we found that links $(5, 9)$, $(3, 6)$, and $(4, 7)$ were added to the original NOTN in decreasing order of ℓ_s. The resulting network is given in Fig. 7. Its cost is 38.5 which is 11% more expensive than the network of Fig. 6. Setting $\overline{h} = 3$, we obtain a network where

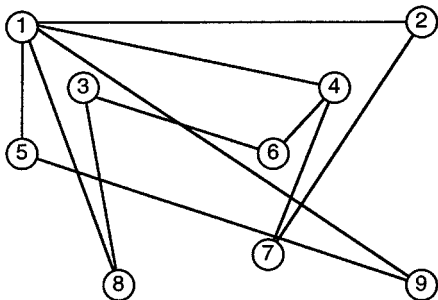|	Cost	Revenue	z	L_{max}^k
TU	34.8	214.3	179.5	.036
CI ($\bar{h} = 2$)	38.5	222.1	183.6	.021
CI ($\bar{h} = 3$)	41.8	224.3	182.5	.009
LU	36.7	224.5	187.2	.008
NOTN	36.1	224.4	187.7	.008

Fig. 7.   Increasing the network connectivity, $h_{\min} = 2$.
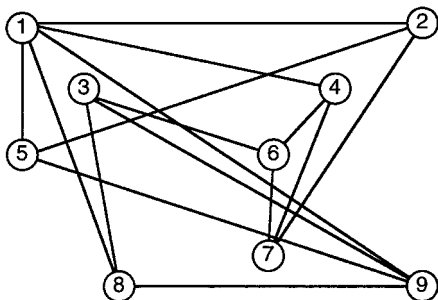

Fig. 8.   Increasing the network connectivity, $h_{\min} = 3$.

links $(8, 9)$, $(2, 5)$, $(6, 7)$ and $(3, 9)$ are further added. The resulting network is given in Fig. 8. Its cost is 41.8 which is 20% more expensive than the network of Fig. 6.

We can see from Table IX that this has the desired effect to the extent that the GoS is better. The revenue increase for $\bar{h} = 2$ is not sufficient to compensate for the cost increase because there are not enough links to construct new trees on which the calls can be routed. We need to set $\bar{h}$ to 3 in order to achieve an acceptable GoS but at the expense of a 15% cost increase which is much higher, in relative terms, than the 5% increase in revenue. This result is typical of what we have seen for other networks and seems to depend weakly on the cost of capacity C_0 and the revenues w^k. C_0 and w^k should be translated in dollars in order to give a clearer insight into the actual increase of the cost and revenue of the network.

A third possibility is to keep the initial topology of Fig. 6 but to reoptimize the link capacity and the tree selection. The results on the fourth line of Table IX indicate a much lower cost increase and a significant increase in revenue as compared with the first row.

We can estimate the quality of this solution if we finally recompute the NOTN for the actual demands. The results of the last row of Table IX do indeed indicate that tree selection combined with an adjustment of the link capacity is a very good strategy to follow in the presence of unforeseen traffic demands.

Therefore, we can state that if the cost of capacity C_0 and the revenue of a connection w^k are comparable, in the presence of unforeseen traffic demands, *it is more economical to adjust the link capacities and to select new trees than to increase the network connectivity*. This solution is feasible if the capacities can be updated rapidly as for ATM VPs and also for future dynamically adjustable capacity networks such as the ones described in [12].

V. CONCLUSION

In this paper, we have examined the problem of an ISP that needs to provision his network in the presence of point-to-multipoint connections. First, we have defined a model which we think captures some essential features of the network design problem for multipoint communications while being simple enough to give some insight into the structure of good solutions.

Next, we have seen that point-to-multipoint traffic concentration on a small number of links with large capacities significantly reduces the cost of the network (by up to 45%). These savings are expected to be even more important when the cost of capacity function is concave, which is the case in practice.

We have found that using a single tree to connect the calls of a given type is very often near-optimal which means that we need not have more than a single route (tree or path) to connect the calls of a given type. Hence, optimizing the load sharing coefficients is not very important in the network design context. We then proposed a heuristic method for selecting a near-optimal set of candidate trees based on the traffic concentration principle.

We also proposed a method to design the network when point-to-point and multipoint traffics are mixed. We have shown that the optimal network topology can have any connectivity between that of a tree and that of a fully connected network, depending on the intensity of both types of traffic.

We have then examined the problem posed by real-time traffic which may be different from the traffic forecast used for the dimensioning. In [15], we have shown that having a highly connected network is necessary to be able to optimize the call routing in order to achieve an acceptable performance. The question is then: "What happens in the case where we want to optimize the routing and the topology produced by the dimensioning algorithm is a tree?"

We proposed in this paper a procedure to economically increase the connectivity of the network and we have shown that the price to pay in order to be able to reoptimize the call routing may be much more significant than the increase of revenue obtained by optimizing the call routing (this depends on the cost and revenue parameters). We then have shown that in order to take traffic variation into account, it is more economical to adjust the link capacities, which is feasible when they can be updated quickly as for ATM VPs.

We have also explained how this heuristic can be used even when the traffic cannot be forecasted, which makes it of practical interest. In practice, we can measure the link blocking probabilities and the traffic currently carried by the network links

and use these to approximately update the link capacities or re-design the network in order to concentrate the traffic in the case of virtual networks.

This work has given us some insight into the structure of optimal network design and routing for multipoint connections. This, in turn, has yielded a number of questions that still remain to be examined.

One such issue is how to implement and evaluate the performance of a real-time traffic concentration algorithm based on this heuristic (distributed or centralized) that uses traffic measurements.

Another issue arises when the actual traffic demands are different from those for which the network has been dimensioned and when it is not possible to reoptimize easily the capacities. In that case, the only way to improve the GoS and the revenue is to reoptimize the routing, something that is not possible if the network has been designed with a single route tree. An issue would be to verify if the traffic concentration principle holds in the case of a more realistic routing technique such as alternate routing.

Finally, one important question is how to implement and test the performance of a real-time algorithm that adjusts the link capacities in the presence of traffic variations.

REFERENCES

[1] H. Eriksson, "MBONE: The multicast backbone," *Commun. ACM*, vol. 37, no. 8, pp. 54–60, Aug. 1994.
[2] "ATM user network interface specification," ATM Forum, Version 3.1, Tech. Rep., Sept. 1994.
[3] A. Girard, *Routing and Dimensioning in Circuit-Switched Networks*. Reading, MA: Addison-Wesley, 1990.
[4] ——, "Synthesis models for packet-switched networks with QoS constraints," in *Proceedings of ITC-16*, P. Key and D. Smith, Eds. Amsterdam, The Netherlands, June 1999, pp. 919–928.
[5] A. Girard, P. D. Lansard, B. Liau, and J. L. Thibault, "Optimization of out-of-chain routing," in *Proc. 4th Network Planning Symp.*, 1989, pp. 99–106.
[6] A. Girard and C. Rosenberg, "A unified framework for network design with generalized connections," in *Teletraffic Contributions for the Information Age: Proc. ITC 15, Volume 2 of Teletraffic Science and Engineering*, V. Ramaswami and P. E. Wirth, Eds. Amsterdam, The Netherlands: Elsevier, June 1997, pp. 319–328.
[7] A. Girard and S. St-Georges, "Performance evaluation and routing optimization of ISDN networks with nonhierarchical and alternate routing," presented at the Telecommunication Systems Modeling and Analysis Conf., Mar. 1993.
[8] "Text of draft recommendation i.326: Functional architecture of transport networks based on ATM," International Telecommunication Union-Telecommunication, Tech. Rep., July 1995.
[9] F. P. Kelly, "Blocking probabilities in large circuit-switched networks," *Adv. Appl. Prob.*, vol. 18, pp. 473–505, 1986.
[10] P. Key and D. Smith, Eds., *Proc. ITC-16*. Amsterdam, The Netherlands: Elsevier, June 1999.
[11] J. F. P. Labourdette and G. W. Hart, "Blocking probabilities in multi-traffic loss systems: Insensivity, asymptotic behavior and approximations," *IEEE Trans. Commun.*, vol. 40, pp. 1355–1366, Aug. 1992.
[12] E. Livermore, R. P. Skillen, M. Beshai, and M. Wernik, "Architecture and control of an adaptive high-capacity flat networks," *IEEE Commun. Mag.*, vol. 36, pp. 106–112, May 1998.
[13] A. Meddeb, "Synthèse de réseaux pour des communications multi-point," Ph.D. dissertation, École Polytech. Montréal, Montréal, QC, Canada, 1998.
[14] A. Meddeb, A. Girard, and C. Rosenberg, "The impact of tree selection on the design of networks with multipoint connections," in *Proc. IEEE Globecom'97*, Nov. 1997, pp. 1891–1897.
[15] ——, "Tree selection heuristics for point-to-multipoint call routing in telecommunication networks," in *Proceedings of ITC-16*, P. Key and D. Smith, Eds., Amsterdam, The Netherlands, June 1999, pp. 539–550.
[16] P. Newman, T. Lyon, and G. Minshall, "Flow-labeled IP: A connection-less approach to ATM," in *Proc. IEEE Infocom'96*, 1996, pp. 1251–1260.
[17] P. P. White, "RSVP and integrated services in the internet: A tutorial," *IEEE Commun. Mag.*, vol. 35, pp. 100–107, May 1997.

Aref Meddeb (S'94–M'99) received the "Diplôme d'Ingénieur" in electrical engineering from ENIT, Tunis, Tunisia, and the M.S. and Ph.D. degrees in electrical engineering and computer science from the Ecole Polytechnique, University of Montréal, Montréal, QC, Canada, in 1992, 1995, and 1998, respectively.

From 1992 to 1993, he was with Alcatel Data Networks, Tunisia. From 1999 to 2000, he was a Research Associate with INRS-Télécommunications, Montréal, QC, Canada. He was with Teleglobe Canada from 2000 to 2001. In April 2001, he joined Nortel Networks, Marne-La-Vallée, Bussy-Saint-Georges, France, where he is working on voice over packet solutions. His research interests include network planning, intelligent Internet, and voice over packet networks.

André Girard received the B.Sc. degree from the Université Laval, Québec, QC, Canada, and the Ph.D. from the University of Pennsylvania, Philadelphia, in 1967 and 1971, respectively, both in physics.

From 1971 to 1972, he was with the Centre National d'Etude des Télécommunications, Paris, France, as a Post-Doctoral Student. Since 1972, he has been a member of the faculty of INRS-Télécommunications, Montréal, QC, Canada, where he is currently a Full Professor. He was also Director of the center from 1997 to 2001. His research interests are in the application of the methods of optimization to the design, traffic engineering, and analysis of telecommunication networks. He has done extensive work on telephone systems, ATM, and IP networks. He the author of *Routing and Dimensioning of Circuit-Switched Networks* (Boston, MA: Addison-Wesley, 1990), as well as numerous journal and conference articles. He is an Associate Editor of *Telecommunication Systems*.

Dr. Girard is a member of ORSA and a Member of the Board of the Canadian Institute for Telecommunication Research.

Catherine Rosenberg received the "Diplôme d'Ingénieur" from ENST-Bretagne, Brest, France, the M.S. degree in computer science from the University of California, Los Angeles, and the Doctorat en Sciences degree from the Université de Paris XI, Orsay, France, in 1983, 1984, and 1986, respectively.

From 1988 to 1996, she was with the Department of Electrical and Computer Engineering, Ecole Polytechnique, Montréal, QC, Canada. In September 1996, she joined Nortel Networks, Harlow, U.K., where she created and headed the R&D Department in Broadband Satellite Networking. She was also a Visiting Professor in the Department of Electrical and Electronics Engineering, Imperial College, London, U.K. In 1999, she joined the faculty of the School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN, as an Associate Professor with Tenure. She has authored more than 50 papers on broad-band networking and traffic engineering.