Routing of Voice and Data in Burst-Switched Networks

BASIL MAGLARIS, MEMBER, IEEE, ROBERT R. BOORSTYN, FELLOW, IEEE, SHIVENDRA PANWAR, MEMBER, IEEE, THEODORE SPIRTOS, MEMBER, IEEE, PETER O'REILLY, SENIOR MEMBER, IEEE, AND CAROLYN JACK

Abstract—**This paper addresses the static and centralized routing of voice and data traffic in burst switched networks. Since voice and data have markedly different traffic characteristics and performance requirements, their interaction in a unified switching and transmission structure greatly impacts their performance. Voice performance in a burst switched network is primarily measured by end-to-end freeze-out while figures-of-merit for data include end-to-end delay and stability that may be jeopardized due to the interaction with the higher priority voice traffic. We assume that our routing allows random bifurcation in voice and data paths and preemptive priorities for voice requirements.**

We first study routing of voice only, by using a multicommodity flow model with linearized link losses and average network loss as a minimization objective. Solving the resulting linear program, we observe that optimal routing strategies prefer to freeze a requirement at an early stage of its path rather than those requirements that are close to their destinations. We then study the voice/data interaction at the link level using an available fluid-flow model, and translate the combined link performance as a maximum flow constraint on a link. This constraint may have undesirable effects on the voice, such as introducing routes with flow absorbing loops, and unfair freezing of some requirements. We include all conflicting multiple objectives and constraints in a linear programming formulation and show how parameters can be tuned to produce desirable voice and data paths.

I. Introduction

THE new switching technologies that have been proposed for the Integrated Services Digital Network (ISDN) range from combinations of traditional circuit and packet switching, to more radical proposals such as fast circuit switching [1], and fast packet switching [2]. We concentrate in this paper on routing optimization issues for burst switching (BS) [3], [4], a technology that provides a unified switching and transmission structure for both voice talkspurts and data messages. In burst switching, talkspurts and data messages are statistically multiplexed onto TDM transmission links. Congestion in a BS network is resolved by clipping (and/or buffering) speech and buffering data; voice has nonpreemptive priority over data.

We use the BS technology as a baseline to investigate the effect of integrating in one network digitized voice and bursty data. These two traffic components have markedly different statistical characteristics and performance objectives. Thus, their performance in a common switching and transmission structure may be very sensitive to their interaction. Such correlations were found to have serious effects on data delays in movable boundary multiplexing of synchronous voice streams and buffered data packets, e.g., [5].

In what follows, we briefly describe burst switching for voice and data, discuss network performance models, and formulate the routing problem as a multiple objective multicommodity flow optimization. In particular, we simplify the optimization by using piecewise linear approximations of the voice freeze-out function, and model the link-level voice-data interaction as a linear constraint. We present numerical solutions of the resulting linear programs in small sample networks and discuss parameter tuning that produce reasonable compromises among the conflicting performance objectives.

We assume that our routing allows random bifurcation in voice and data paths and preemptive priorities for voice requirements. Furthermore, the network traffic is assumed to be at steady state, and routing decisions are based on a global, static network state. We make no attempt to include adaptive or distributed routing.

II. Modeling Burst Switched Networks

A. Burst Switching

Burst switching is a technology based on TDM (time division multiplexing) transmission which can carry both voice and data traffic. Digital speech interpolation (DSI), the digital equivalent of TASI (time assigned speech interpolation), is used on voice calls to statistically multiplex many voice calls onto a small number of TDM channels. We shall consider here only 64 kbit/s PCM encoded voice multiplexed onto $T1$ carriers, which thus provide 24 channels per link.

At the interface to the network, speech or silence detectors delimit talkspurts and silence periods. Typically, each off-hook caller in a conversation is in a talkspurt 30–40% of the time. On the average, talkspurts last between 250 and 350 ms. The interface creates bursts out of each talkspurt by adding a four-byte header (three bytes for address and control information and one for error control of the header) and a terminating flag. The burst format is indicated in Fig. 1.

After call control has established a connection through the network from the source to the destination, the address information in each burst is used to switch it through the network of multiplexed $T1$ links. Note that all links are full-duplex (FDX). Whenever there are more than 24 bursts present, the excess bursts are frozen-out (clipped) until a channel becomes available, as shown in Fig. 1 where three voice calls are multiplexed onto two channels. Only PCM encoded voice is clipped from the burst. The header of the burst is stored and reinserted at the head of the remaining part of the burst. It should be noted that burst switching has the capability to store up to 4 ms of speech at any node before clipping is enacted. However, in this study, we assume that there is no buffering of speech. The average link freeze-out Φ is given by Weinstein's TASI formula, [6]

$$\Phi = \frac{1}{Na} \sum_{k=C+1}^{N} (k - C) \binom{N}{k} a^k (1-a)^{N-k}. \quad (1)$$

Paper approved by the Editor for Communications Networks of the IEEE Communications Society. Manuscript received March 12, 1987; revised February 18, 1988 and November 16, 1988. This work was supported in part by a grant from GTE Laboratories and by the New York State Foundation for Science and Technology as part of its Centers for Advanced Technology Program.

B. Maglaris, R. R. Boorstyn, S. Panwar, and T. Spirtos are with the Department of Electrical Engineering and Computer Science, Polytechnic University, Brooklyn, NY 11201.

P. O'Reilly and C. Jack are with GTE Laboratories, Waltham, MA 02254.

IEEE Log Number 9035779.

0090-6778/90/0600-0889$01.00

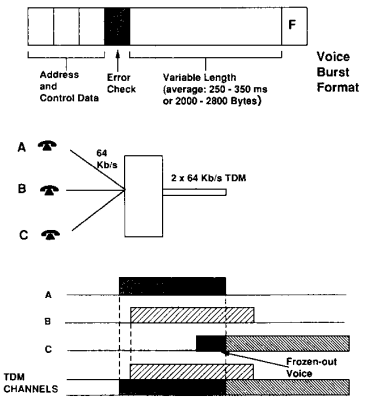


Fig. 1. Example of voice freeze-out.

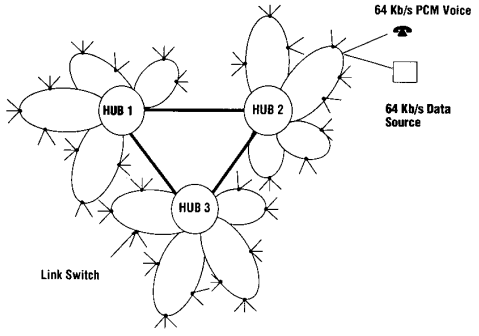


Fig. 2. A burst-switched network.

Here N is the number of off-hook sources connected to a link of C subchannels, and a is the activity factor, i.e., the average fraction of time an off-hook source is actively transmitting a talkspurt. Note that the formula applies for any distribution of talkspurt and silent period lengths. With $N = 50$ sources, $C = 24$ channels, and $a = 0.4$, the freeze-out Φ is of the order of 0.5%, which is the CCITT objective for acceptable voice quality on a TASI link.

Data users are connected via a 64 kbit/s interface that produces data bursts (i.e., low bit rate data is accumulated and then transmitted at 64 kbits/s). Typically, a data burst is less than 80 bytes long on the average or 10 ms at DS0 rate. The data burst format is similar to the voice burst format except that an additional two CRC bytes are appended to the information stream for the purpose of end-to-end error control. Voice bursts have nonpreemptive priority over data bursts and are switched via cut-through switching: if there is a channel available (a time slot in the TDM frame) in tandem links, the burst is pipelined as in circuit switching, with minimal buffering per link for address processing. Thus, under low load conditions, burst switching behaves as a form of fast circuit switching. If a carrier is filled, the data burst is stored at this stage until a channel is released.

A burst switched network is envisioned in [3] and [4] to consist of two-level hierarchy. One possible local access subnet connects subscribers via link switches and $T1$ lines arranged like the petals of a flower around a hub switch. The backbone interhub network interconnects via $T1$ lines several such local access areas, as illustrated in Fig. 2. Our work on routing optimization concerns the backbone subnet. For analyses and simulations of the local access subnet we refer the reader to [8].

B. Voice Independence Approximations

For a given backbone topology and traffic, routing optimization involves assignment of paths (possibly bifurcated and involving priorities) that minimize some network performance measures. In our case, such measures include end-to-end freeze-out for voice and queueing delays for data (with the assumption of no buffering for speech). The end-to-end performance analysis is extremely hard because of link dependencies for both voice and data burst switching. We have established that independence can be safely assumed in distributed networks with several independent end-to-end requirements. To illustrate our independence assumption, consider the 3 node chain in Fig. 3. N_{ij} denotes the number of off-hook sources with origin i and destination j. Φ_{ij} denotes the average freeze-out of traffic originating at i and destined for j while Φ_k denotes the average freeze-out for link k. We assume independent binomial sources that generate voice traffic at any random time instant with probability a, the activity factor. All links have 24 subchannels. An exact evaluation of the end-to-end freeze-out Φ_{ij} can be achieved by convolving at each node the distributions of traffic components carried from upstream and generated locally at the node [9]. Such an approach is extremely complex and approximations may be required for mesh-type networks. We used instead an approximate method that models the total offered traffic into a link as an equivalent binomial source and uses (1) to evaluate the link freeze-out. The end-to-end freeze-out from node 1 to 3 in the figure is approximately

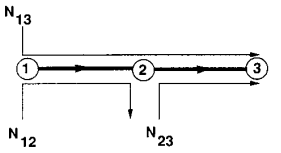


Fig. 3. A three-node chain network.

$$\Phi_{13} \approx 1 - (1 - \Phi_1)(1 - \Phi_2).$$

The equivalent binomial method (EBM) is described next. Consider a node that receives upstream random traffic with mean $E(t)$ and variance $\mathrm{var}(t)$. The tandem variable t represents the number of upstream channels occupied by talkspurts at steady state. Let N sources be directly connected to the same node, with activity a. The node multiplexes the tandem and new traffic onto the downstream link. In case the total number of active calls exceeds 24, it clips the excess bursts by selecting at random from both the tandem and new traffic. This can be modeled by a hypergeometric distribution. The EBM bundles both input streams into M independent sources with activity b, and chooses the parameters M and b so as to match the first two moments of the total traffic from both input streams. Thus,

$$Mb = E(t) + Na \qquad \text{(mean matching)}$$

$$Mb(1 - b) = \mathrm{var}(t) + Na(1 - a) \qquad \text{(variance matching)}.$$

The two equations above determine the equivalent binomial source (M, b). The TASI formula, (1), is used to obtain the freeze-out and carried traffic statistics of the link downstream. In case of noninteger M, we use a linear interpolation of results obtained with the floor and ceiling of M. This process is repeated for all links in a chain. Traffic that exists from a link without being offered to the next stage, is accounted in evaluating $E(t)$ and $\mathrm{var}(t)$ by randomly splitting the carried traffic in the correct proportions via a hypergeometric distribution. A simpler version of the EBM, which does not require hypergeometric splits, involves first moment matching only. In this case, the equivalent binomial source (M, a) does not reflect the variance reduction (smoothing) of tandem traffic as it proceeds through the 24 channel links. Nevertheless, for link freeze-out values less than 2% such smoothing does not seem to have a major impact (recall that the CCITT objective is 0.5%).

For the chain in Fig. 3, we compared the two and single moment EBM results to exact calculations using the method described in [8]. We assume 35 off-hook sources from 1 to 2, 25 from 1 to 3, and 30 from 2 to 3. All sources have activity $a = 0.4$ and all links consist of 24 channels. In Table I we give the end-to-end freeze-out under the three methods. The two-moment matching EBM modeled the second

TABLE I
EBM RESULTS FOR CHAIN NETWORK (FIG. 3)

ETE Freeze-out	2-Moments	1-Moment	Exact [9]
Φ_{13}	13.79%	14.30%	13.54%
Φ_{12}	6.28%	6.28%	6.28%
Φ_{23}	8.02%	8.60%	8.30%

TABLE II
EBM RESULTS FOR TRIANGULAR NETWORK (FIG. 4)

N_{13}	N_{21}	N_{32}	2-Moments	1-Moment	Simulation
25	25	25	7 Iter.	4 Iter.**	
	Φ_{13}		1.65%	2.02%	1.82%
	Φ_{21}		1.65%	2.02%	1.82%
	Φ_{32}		1.65%	2.02%	1.82%
20	30	40	9 Iter.	6 Iter.**	
	Φ_{13}		1.88%	3.41%	2.66%
	Φ_{21}		15.91%	16.04%	15.90%
	Φ_{32}		16.21%	17.33%	16.41%
30	30	30	3 Iter.	7 Iter.**	
	Φ_{13}		8.75%	10.01%	9.32%
	Φ_{21}		8.75%	10.01%	9.32%
	Φ_{32}		8.75%	10.01%	9.32%

** Number of Iterations for Convergence

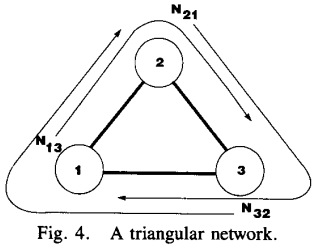


Fig. 4. A triangular network.

link as serving 48.5 sources with activity 0.518. The single-moment EBM considered instead 62.8 equivalent sources of activity 0.4. This demonstrates the smoothing effect of the first link in the two-moment method (higher activity corresponds to lower variance of the offered traffic).

We also implemented the EBM (with 2 and 1 moment matching) for networks of general topologies that include meshes. In such cases, we use an iterative procedure that first loads the links assuming no freeze-out and then applies the EBM by iterating on offered and carried link flows. In Table II we provide end-to-end freeze-out results for the triangular network of Fig. 4 where N_{ij} and Φ_{ij} are as defined earlier. The voice activity factor here is 0.4 and again there are 24 channels on each link. This network is a worst case example with heavily correlated traffic. In the table we compare the EBM to two moment and single moment matching with simulation results. These results, consistent with many more comparisons that we have made, convinced us that we may use (1) with 24 channels and uniform activity a to model voice performance in networks of tandem burst switches.

C. Data Independence Approximation

The results above refer to networks handling voice only. In cases where data bursts are integrated with voice, even single link models become considerably complex. We concentrated on the fluid-flow technique reported in [10]. This analysis assumes that voice bursts have preemptive priority over data bursts, an assumption that was found satisfactory for data packet lengths at least 15 times shorter than voice bursts. Typically, voice bursts last about 300 ms while interactive data bursts are less than 10 ms. Furthermore, the fluid-flow model assumes that voice talkspurts and silent periods are exponentially distributed, and that the data input flow is a continuous stream, so that statistical fluctuations of the data input are ignored. Albeit its approximate nature, the model captures the basic correlation between voice load and data queueing, which is the major contribution to delays on integrated links.

TABLE III
VOICE-DATA INTERACTION FOR TANDEM LINKS (FIG. 3)

N_{13}	S	D_{12}	D_{23}
0	0	9.8	10.3
14	0.33	10.4	9.6
21	0.49	9.7	7.7
29	0.67	8.7	7.0
36	0.84	9.3	4.7
43	1.00	9.8	0

In the case of tandem integrated links, voice performance can be analyzed using the EBM as before, since the effect of data on voice is negligible for short data packets and links that primarily convey voice. As far as average data queueing delay is concerned, we studied via simulation the viability of assuming link independence. We considered two tandem links as in Fig. 3 with both end-to-end (two hop) data traffic and a mix of one-hop and two-hop voice traffic. Table III illustrates a typical set of results. In this case, we kept the voice link loading the same for both links

$$N_{12} + N_{13} = N_{23} + N_{13} = 43.$$

Other parameters are a voice activity factor of 0.3717, 24 channels (all shared by voice and data), exponentially distributed talkspurts of average length 283 ms, 350 Poisson data sources with an average data burst length of 10 ms (exponentially distributed), and an aggregate data rate of 224 kbits/s. In the table, D_{ij} represents the average queueing delay (in ms) on link (i, j) while S is the fraction of voice traffic that goes over 2 links, i.e., $S = N_{13}/(N_{12} + N_{13})$. As observed in [10], data performance given by simulation of an integrated burst-switched link is sensitive to the average talkspurt and silence intervals generated in the simulation run; this is noticeable in the case of independent voice traffic ($S = 0$), although some increase in delay is due to the fact that data arrivals to the second link are no longer exactly Poisson. In fact, a general conclusion from a range of such simulation studies is that data traffic on the links become less "Poissonian" as the voice load (per link) is increased. These studies also showed that, for average data delay, link independence was a viable and appropriate assumption until the two-hop voice traffic exceeded 40% of the link load: to be more precise, the error due to this approximation is less than 20% for $S < 0.55$. In a distributed backbone network with a reasonable degree of connectivity, it is expected that voice and data requirements are interwoven in such a way that link independence is easily justified for both voice and data.

III. ROUTING OPTIMIZATION—VOICE ONLY

Consider a BS network of 24 channel FDX links, with given end-to-end voice traffic requirements in off-hook sources or, multiplying by the voice activity, average end-to-end offered traffic in Erlangs. Routing can be formulated as a multicommodity flow problem with link losses (the traffic frozen in links). Commodities correspond to end-to-end requirements and may be split in any proportion. Such splits assume randomly bifurcated routes, and constitute the variable space for optimization. A reasonable optimization objective is the minimization of the average flow loss. Such problems have been studied for linear link losses using linear programming or augmentation algorithms [11]. The link losses in our case are convex functions of the offered and tandem traffic.

Assuming link independence and using the EBM with first moment matching, each link performs according to Weinstein's TASI formula

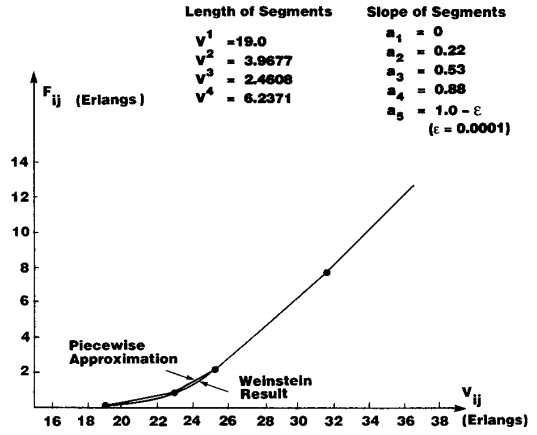


Fig. 5. Link freeze-out versus offered traffic.

(1). The average amount of lost voice F (in Erlangs) as a function of the average offered voice traffic V (in Erlangs) is $F = \Phi \times V$. The freeze-out fraction Φ is given from (1) with $C = 24$ channels; N the population of the equivalent binomial source and the voice activity $a = 0.4$ are assumed to be the same throughout the network. In Fig. 5, we plot F versus V for the parameters of interest. For $V \leq 9.6$ Erlangs, corresponding to less than 24 off-hook sources, $F = 0$. For large V the F curve asymptotically assumes a slope of 45° (all traffic in addition to V is completely cutout). As can be seen in the figure the curve can be accurately linearized with a small number of segments. A linearization with 5 segments is shown in Fig. 5. With piecewise linear link losses the routing problem can be formulated as a linear program (LP). Similar linearization of convex functions has been suggested in network flow problems, and in particular in communication networks, e.g., [12].

A. LP Formulation

We proceed to the LP formulation. Let $r_{s,d}$ be the average end-to-end offered traffic in Erlangs, V_{ij} the average offered traffic to the directed link (i, j) and F_{ij} the frozen traffic in (i, j), provided by the piecewise linear function in Fig. 5. The objective is to minimize the average end-to-end network freeze-out, which is equal to the sum of all link frozen traffic divided by the sum of end-to-end requirements. With the latter being constant, the objective becomes

$$\text{minimize} \left\{ \sum_{i,j} F_{ij} + \alpha \sum_{i,j} V_{ij} \right\}. \tag{2}$$

The first summation in (2) is the total average traffic loss. The second, with α a is small multiplier (e.g., $\alpha = 0.0001$), enforces minimum hop paths in case of ties, and avoids looping as it penalizes against unnecessary link offered loads. The constraints to the optimization are deduced from flow conservation in each node, and from the freeze-out curve of Fig. 5 for each link. We establish node conservation constraints by dividing offered and frozen link loads into distinct portions (commodities) destined to each eventual destination node d. Thus, it is assumed that the routing plan is destination dependent (source independent) and the number of commodities is equal to the number of nodes. The conservation equation for node i and commodity d, is

$$\sum_{n} [V_{ni}^d - F_{ni}^d] + r_{i,d} = \sum_{j} V_{ij}^d \quad \forall (i, d)\ i \neq d. \tag{3}$$

The superscript d refers to the portion of the traffic that is destined to d.

To enforce the freeze-out curve, we further divide the offered and frozen traffic per commodity into parcels that fit the five segments of

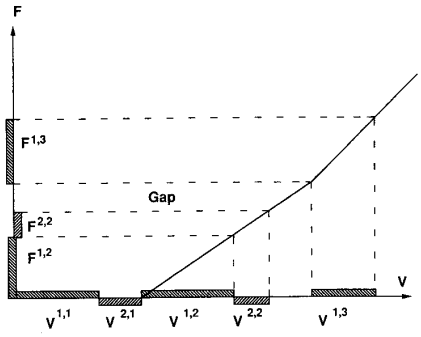


Fig. 6. An example of a gap in the $(V - F)$ curve.

the $(V-F)$ curve. Let a_k be the slope of the kth segment. Obviously, $a_0 = 0$ and $a_5 = 1$. Then, for all links (i, j) and destination nodes d

$$F_{ij}^{d,k} = a_k V_{ij}^{d,k},\ k = 2, \cdots, 5, \tag{4.a}$$

$$V_{ij} = \sum_{d} V_{ij}^{d} = \sum_{d,k} V_{ij}^{d,k}, \tag{4.b}$$

$$F_{ij} = \sum_{d} F_{ij}^{d} = \sum_{d,k} F_{ij}^{d,k}. \tag{4.c}$$

An additional constraint enforces all offered traffic parcels in a link to fit within the space V^k allowed for the kth segment in the V axis

$$\sum_{d} V_{ij}^{d,k} \leq V^k \quad k = 1, \cdots, 5. \tag{4.d}$$

The total number of variables is $s(n-1)L$ where s is the number of segments, n the number of nodes and L the number of links in the network. The total number of constraints is $n(n-1)+(s-1)L$. Both increase polynomially with the number of nodes in the network. The above LP can be easily expressed in terms of only the $V_{ij}^{d,k}$ nonnegative variables. Due to the minimization objective, an optimal solution will tend to fill all lower segments of the convex $(V - F)$ curve, since they are less costly. However, it will not necessarily allocate commodities to all segments in the same proportion and thus may treat preferentially different destinations in the same link. This is by no means an unreasonable strategy as we observed in our computational experiments. It allows an intelligent routing plan with global knowledge of average network flows, to freeze traffic in an early stage of its path rather than let it unnecessarily load links and then freeze it as it nears its destination. Such a strategy was proposed for the long distance circuit switched telephone network. In some instances the LP solution included gaps in the $(V - F)$ curve as shown in Fig. 6. This is a counter-intuitive result given its convexity. The gaps created additional loss than what would be generated by the natural TASI mechanism. Again, they were fully justified by the global nature of the routing strategy. As the additional frozen traffic would have overloaded links downstream at the slope 1 segment, it might as well be frozen out at an early stage (recall that the objective is to minimize losses, regardless of commodity). We were able to avoid creating gaps by making the slope of the last segment slightly less than 1. This increases the incentive to pass as much traffic through any link even at the segment of slope 1, and eliminates the possibility of completely cutting a portion of the traffic earlier in the path.

B. LP Solution for a 5-Node Network

We demonstrate some of the above remarks in the network of Fig. 7, with 5 nodes and 6 FDX links (12 directed links). We assumed 13 Erlangs of end-to-end voice traffic for all source-destination pairs. The corresponding LP has 240 nonnegative variables and 108 con-

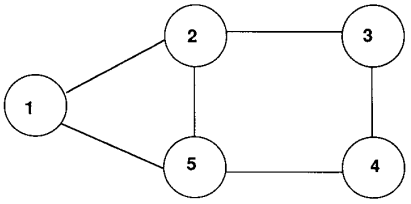


Fig. 7. A five-node network.

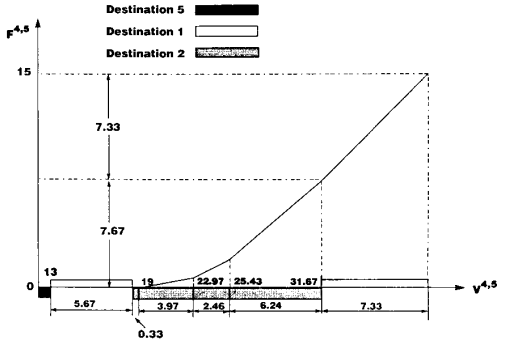
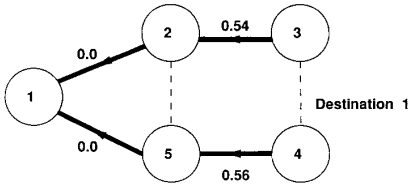


Fig. 8. $V - F$ breakdown for link (4, 5).



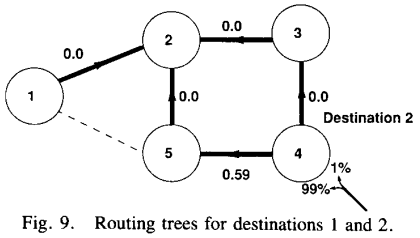


Fig. 9. Routing trees for destinations 1 and 2.

straints. Using the LP 83 package on an IBM PC with math coprocessor, the LP converged to a solution within 3 min. The minimum end-to-end freeze-out fraction was found to be 23%. To demonstrate the allocation of commodities in different segments, we depict in Fig. 8 the $(V - F)$ breakdown for link (4, 5). The total offered flow consists of 39 Erlangs, of which 15 Erlangs were frozen. As can be observed from the figure, there were no gaps in the total link freeze-out. Breaking it into commodities, the LP determined that all 13 Erlangs of traffic destined for 5 (the end node of the link) encountered no freeze-out, while 13 Erlangs of traffic for 1 and 13 Erlangs for 2 (at least another link away) suffered 7.33 and 7.67 Erlangs of loss, respectively. This is an example of how the LP solution indicates that freezing-out commodities further from their destination is preferred. In Fig. 9 we illustrate the routing trees for destination 1 and 2. The numbers next to the links represent the freeze-out fraction for that particular commodity. The traffic at node 4 for destination 2 is randomly split into 99%–1% between the two possible paths. Notice that the tree routed to destination 1 is slightly asymmetric, although the topology and traffic requirements would appear to dictate otherwise. We reran the LP with symmetry enforced via additional constraints and obtained a symmetric solution with exactly the same objective. As in many network problems, there is a large multiplicity of optimal solutions.

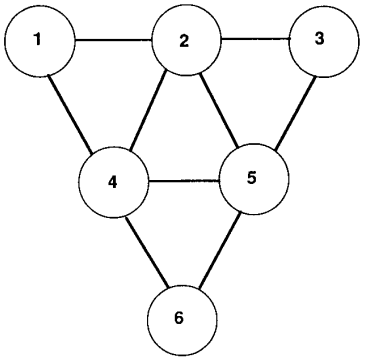


Fig. 10. A six-node network.

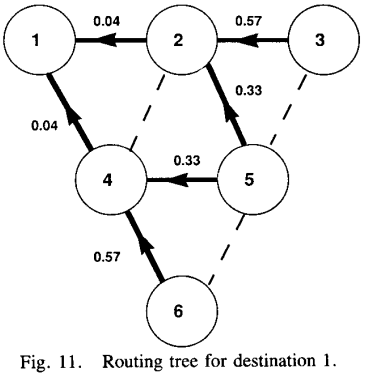


Fig. 11. Routing tree for destination 1.

C. LP Solution for a 6-Node Network

In Fig. 10 we show a six node network. As before, there is a uniform end-to-end requirement of 13 Erlangs. The same linear programming package was used to analyze this network. The number of variables is 450 and the number of constraints is 102, not including the non-negativity constraints. The running time was approximately 6 min. The average network freeze-out was 17.5%. The routing tree for destination 1 is shown in Fig. 11. We see clearly that traffic one hop away from its destination has priority over traffic further away. For example, traffic coming out of nodes 2 and 4 (one-hop traffic) suffers a relatively small freeze-out (about 4%), while traffic two hops away suffers a huge freeze-out, ranging from 33% for source node 5–57% for source nodes 3 and 6.

Let us consider the distribution of voice traffic for links (1,2) and (2,5). No gaps appeared, but for link (1, 2) the LP broke up the traffic destined for node 5 into two parts. The first part of this traffic, which is 5.4 Erlangs, was not frozen at all, while the rest of this traffic, which is 0.29 Erlangs, fell into the fourth segment which corresponds to an incremental freeze-out of 0.88. This is similar to the breaking up of traffic destined to node 1 on link (4, 5) on the 5 node network into two noncontiguous parts on segments 1 and 4, respectively (see Fig. 8). The remaining traffic on link (1, 2) consisted of 13 Erlangs of traffic destined to node 2 which was not frozen at all, while 12.97 Erlangs of traffic destined to 3 separated the two parts destined to node 5. On the other hand, the commodities on link (2, 5) destined to nodes 5 and 6, are contiguous. 18.4 Erlangs of traffic destined to node 5 lies entirely in the first section and is not frozen at all while the 7.03 Erlangs destined to 6 spans sections 1 to 3. In this case

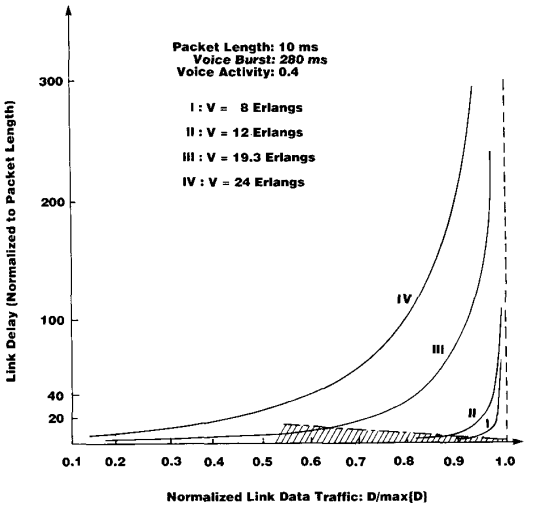


Fig. 12. Data delay curves.

an operational procedure based on a simple priority scheme can be implemented to enforce losses according to the LP results. Traffic destined for node 5 gets first priority, thus suffering no freeze-out, and the traffic destined for node 6 is given second priority. In cases of noncontiguous commodities, as in link (1, 2), a priority scheme based on random partition of a commodity into priority classes can be implemented. For link (1, 2), traffic destined to node 5 should be randomly partitioned into two priority classes with 5.4 Erlangs and 0.29 Erlangs traffic, respectively. A simple priority scheme, giving these two priority classes second and fourth priority, respectively, can now be set up.

IV. Optimal Routing—Voice/Data Integration

We extended the optimization to include end-to-end data traffic by using the fluid-flow voice/data model of [10]. It is assumed that the effect of data on voice is minimal and thus the voice model remains the same as above. For the data, we assumed link independence, and studied the delay performance of a single link under various voice and data average loads. Let V denote the average offered voice load, F the frozen traffic, and D the average offered data traffic intensity (all in Erlangs). For a 24 channel link, $(V - F) + D \leq 24$, and $\max\{D\} = 24 - (V - F)$. In Fig. 12 we plot the average link delay as a function of the data traffic normalized by its maximum value, $D/\max\{D\}$. Direct incorporation of data delays in our optimization is a very difficult task, since there is no analytic expression available for average link delay, and curve fitting would involve two parameters (voice and data flows). We indirectly solved the problem, by observing that the simple linear constraint on the carried traffic of link (i, j)

$$(V_{ij} - F_{ij}) + D_{ij} \leq 22 \tag{5}$$

satisfies all performance objectives in an integrated link of 24 channels. This corresponds to a maximum link offered voice traffic of 22.8 Erlangs. As shown by the shaded region in Fig. 12, all points satisfying (5) operate under the knee of the delay curve, and thus data queueing should remain stable in traffic fluctuations. The knee becomes sharper as the traffic mix includes less voice and more data. In addition to stability, points satisfying (5) exhibit average data delays less than 10 packet lengths and voice freeze-out less than 4%.

We now formulate the integrated LP which minimizes voice ETE (end-to-end) freeze-out under constraint (5) that bounds data delays. We used a modified piece-wise linear approximation of the $(V - F)$ curve, since the offered traffic V cannot exceed 22.8 under (5). The new curve is shown in Fig. 13. The complete formulation is

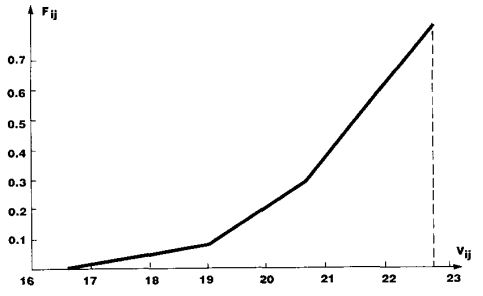


Fig. 13. Modified $(V - F)$ curve.

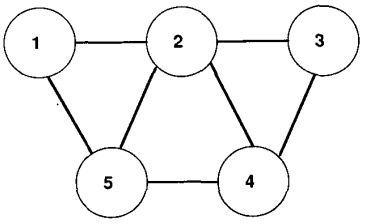


Fig. 14. Another five-node network.

Let $r_{sd}^{(V)}$: Voice ETE requirement (s to d) (Erlangs).
$r_{sd}^{(D)}$: Data ETE requirement (s to d) (Erlangs).
$V_{ij}^{d,k}$: Offered voice traffic, link (i, j), dest. d, segment k.
D_{ij}^{d}: Data traffic, link (i, j), dest. d.
a_k, V^k: Slope and length of segment k in the (V–F) curve.
α, β: Parameters for resolving ties.

$$\text{Objective: } \min \left\{ \sum_{i,j,d,k} (a_k + \alpha) V_{ij}^{d,k} + \sum_{i,j,d} \beta D_{ij}^{d} \right\}. \tag{6.a}$$

$$\text{Constraints: } \sum_{n} D_{ni}^{d} + r_{id}^{(D)} = \sum_{j} D_{ij}^{d} \quad i \neq d \tag{6.b}$$

$$\sum_{n,k} (1 - a_k) V_{ni}^{d,k} + r_{id}^{(V)} = \sum_{j,k} V_{ij}^{d,k} \quad i \neq d \tag{6.c}$$

$$\sum_{d} V_{ij}^{d,k} \leq V^k \tag{6.d}$$

$$\sum_{d,k} (1 - a_k) V_{ij}^{d,k} + \sum_{d} D_{ij}^{d} \leq 22 \tag{6.e}$$

$$V_{ij}^{d,k}, D_{ij}^{d} \geq 0. \tag{6.f}$$

The first summation in the objective (6.a) represents the total voice loss. The parameters α and β help resolve ties in favor of shortest hop paths and eliminate loops. Constraints (6.b) and (6.c) are the flow conservation equations for voice and data commodities. Notice that we assumed source independent routes and did not allow data losses. Constraint (6.d) enforces the fit of voice into the appropriate segments, and (6.e) is the voice-data interaction constraint (5). The total number of variables is $L(n-1)s + L(n-1)$ and the total number of constraints $2n(n-1) + sL + L$, not counting the nonnegativity constraints.

We implemented the above LP for the network shown in Fig. 14. The end-to-end data traffic was set to 3.5 Erlangs, uniform for all source-destination pairs. The voice traffic was set to either 7 or 10 Erlangs in such a way as to impose heavier loads in the links interconnecting nodes 5, 2, and 4. The LP gave a feasible solution with

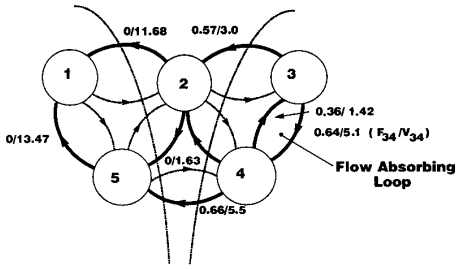


Fig. 15. Voice routing to destination 1.

TABLE IV
VOICE TRAFFIC LOSS (IN ERLANGS), K VARYING

K	Blocked	Frozen	Total	Objective
100.0%	18.0	0.0	18.0	18.03
50.0%	17.9	0.1	18.0	18.03
30.0%	15.9	2.1	18.0	18.03
20.0%	13.6	4.4	18.0	18.03
12.5%	14.3	6.6	20.9	20.93*
11.1%	15.2	6.6	21.8	21.93*

$\alpha = 0.0001$ $\beta = 0.0001$
* Loops observed

average ETE freeze-out of 3.8%. In order to satisfy the total link flow constraint (6.e), it forced some loops within the voice routing. We illustrate this phenomenon in Fig. 15, for destination node 1. The maximum frozen traffic allowed by the freeze-out curve of Fig. 13 is 6.9 Erlangs per link. However, because of heavy congestion in the middle of the network, more loss was required to satisfy the maximum total flow of 22 Erlangs in the inner links. This was achieved by introducing a flow absorbing loop between nodes 4 and 3. Thus, portions of the voice traffic were not offered in the middle of the network; their flow was absorbed as they looped around links with losses.

We eliminated loops by allowing complete loss of traffic before it enters a link. We will refer to this traffic loss as blocking to distinguish it from the TASI induced freeze-out. Obviously, the best place to block traffic is at its source, before it is allowed to unnecessarily load some links. This was verified by running a modified LP that included blocking. Blocking at the source can be viewed as either cutting at random some bursts or parts of them, or as reducing the number of off-hook sources, e.g., by giving busy tones. In general, the LP would block sources in a very asymmetric way, e.g., it might completely block a certain requirement and not block others. This results from the objective formulation that minimizes total network losses. We incorporated fairness in blocking via an additional constraint on the fraction of a requirement that can be blocked. The modified LP includes new parameters B_s^d, the traffic destined for d, blocked at its source s (in Erlangs). In this case the modified LP is

Modified objective:

$$\min \left\{ \sum_{i,j,d,k} (a_k + \alpha) V_{ij}^{d,k} + \sum_{s,d} B_s^d + \sum_{i,j,d} \beta D_{ij}^d \right\}. \quad (7.a)$$

Modified voice conservation:

$$\sum_{n,k} (1 - a_k) V_{ni}^{d,k} + r_{id}^{(V)} - B_i^d = \sum_{j,k} V_{ij}^{d,k} \quad i \neq d. \quad (7.b)$$

Fairness in blocking:

$$B_s^d \leq K \times r_{sd}^{(V)}. \quad (7.c)$$

In addition, the LP requires constraints (6.b), (6.d), (6.e), and (6.f). The parameter K in (7.c) is the maximum allowable source blocking.

We studied the effect of parameter setting in the LP above in the network of Fig. 14. The ETE data traffic was set uniformly to 3.5 Erlangs, while the voice requirements were 13 and 7 Erlangs, the former between nodes 2, 4, and 5, and the latter for all requirements involving nodes 1 and 3. In Table IV we provide results as we vary the blocking percentage K.

For $K = 100\%$ we do not impose any limit to the amount of source blocking. The LP chose to block all traffic instead of freezing it inside the network. The objective is slightly larger than the total traffic loss due to the α and β factors in (7.a). As we restrict individual source blocking, the internal freeze-out increases but the total loss (imposed by the bound on link flows) remains the same. With K less than 13% the LP introduced flow absorbing loops to limit the flow. We observed that some requirements were blocked less than the maximum allowed by K. We then increased the parameter α to weigh the link carried traffic and push blocking of all sources to the maximum allowed. For $K = 11.1\%$, we eliminated loops by increasing α from 0.0001 to 0.1. At this point the total blocked traffic increased to 16.4 (from 15.2) and the frozen traffic dropped to 6.0 (from 6.6 with the loops).

When the voice traffic is increased from 7 to 10 Erlangs there is no solution when K is reduced below 0.33. Clearly, as the voice traffic increases, the minimum value of K must increase since more traffic must be blocked (the hard constraint must be looser). With small values of K, loops have been observed in the voice routing and have been eliminated by increasing the value of α from 0.0001 to 0.1 exactly as before.

Next, we consider the effect of an increase in data traffic from 3.5 to 4.0 Erlangs with the original voice traffic. The minimum value of K for a feasible solution is 0.167. When K is set to this value the solution contains loops. As previously, the required value of α for elimination of the loops is 0.1. We see that the required value of α for eliminating the loops is relatively insensitive to variations of either the voice or data traffic.

The parameter β has enforced min-hop paths in the data routing in most of the cases. When the value of K is set close to 0, or α is increased by an order of magnitude more than β, data routing may not be the min-hop any more. In the first case, this is because the additional voice traffic in the network eliminates some of the routing choices for data traffic. In the second case, data routing may not be min-hop because min-hop routing for voice takes priority over min-hop for data. When an attempt was made to enforce min-hop for data by increasing β to a value equal to that of α, loops in the voice routes that had previously been eliminated reappeared.

On the basis of the previous discussion, the following tuning procedure is suggested. First the value of K is set by the maximum amount of voice loss we are willing to accept for any end-to-end pair. The parameters α and β are initially set at some small value. If a solution exists and does not contain loops the procedure terminates. If there is no feasible solution, the only alternative to terminating the procedure is to increase K to block more voice until we are in the feasible range. If a solution containing loops is found, α is raised until the loops have been eliminated. At this point if the data routing is not min-hop, β can in turn be raised until either loops start appearing again in the voice routing, or min-hop for data is achieved. This parameter tuning will not eliminate loops if no feasible solution exists without flow absorbing loops.

We also studied the effect of the total link traffic hard constraint on the average network freeze-out and delay. The limit of 22 Erlangs we have placed on the link results in an upper limit for voice freeze-out of under 4% and a corresponding limit for data delays of 10 packet lengths. The packet length was kept constant at 10 ms. We parametrized the hard constraints on the link traffic using the parameter X

$$V_{ij} - F_{ij} + D_{ij} \leq X.$$

TABLE V
FREEZE-OUT DELAY INTERACTION, X VARYING

F(%)	B(%)	Total(%)	D(ms)	X
0.0	46.0	46.0	0.00	15.5
0.0	30.0	30.0	0.07	18.0
0.0	15.0	15.0	22.8	22.0
0.0	10.0	10.0	154.25	23.3
0.0	0.0	0.0	∞	26.0

The same five node network was examined with uniform voice and data requirements of 8 and 5 Erlangs, respectively. In Table V we show the effect of variation of X on the freeze-out and delay. The average network delay is obtained from the following formula.

$$D = \sum_l \rho_l^{(D)} D_l \Big/ \sum_{ij} r_{ij}^{(D)}$$

where $\rho_l^{(D)}$ is the data link utilization and D_l is the data link delay. The delay results are based on the fluid flow analysis. The value of the fairness parameter is kept constant at 1. We see that for the whole range of variation of X there is no freeze-out. All the loss is due to blocking, which is consistent with what we observed before. For high values of the fairness parameter blocking is always preferable to freezing. When X is set to 15.5 the total voice loss, entirely due to blocking, amounts to 46.0% while the data delay is negligible. As the constraint is made looser (as we permit more and more traffic to be carried by the link, and therefore less blocking takes place) the total voice loss decreases while the delays increase abruptly (there is more traffic to be carried). When the value of the parameter is increased from 22.0 to 23.3 (a little less than a 6% increase), the voice loss goes down by about 30% but the corresponding increase in delay is on the order of 600%. The delays are extremely sensitive on the value of the parameter. When X is set at 26.0 there is no loss of voice but the delays become unbounded. As we improve one of the objectives the other becomes unacceptably worse. The value of 22 we have chosen is a reasonable compromise between these two conflicting objectives.

V. CONCLUSIONS

Optimization of voice and data routing in a burst switched network was handled via linear programming multicommodity flow models. Routing was assumed to be source independent, with random bifurcation, and allowing nonuniform freezing of voice requirements. It implicitly assumes global knowledge of the network steady-state statistics, and does not account for adaptive rules.

We first validated approximations that analyze voice freeze-out in a network environment and studied routing optimization for voice only. Link losses were modeled via piece-wise linear curve fitting. The linear programming model minimizes the total network loss by selecting optimal paths and freezing requirements (if necessary) early in their path.

Based on fluid-flow queueing results, we incorporated data performance in terms of a bound on the total link carried traffic, and extended the linear programming model to integrate voice and data routing. In addition to link freeze-out, voice traffic can be blocked at the source up to a maximum percentage. The integrated linear programming formulation (7) combines either directly or in its constraints the following multiple objectives:

- Minimum average network loss (source blocking and DSI freeze-out).
- No looping in voice and data paths.
- Stable and bounded data delays. Voice link freeze-out does not exceed 4%.
- Fair source blocking $\leq K$ for all voice requirements.

An additional degree of freedom may be introduced by allowing blocking for data (e.g., data flow control). This can be easily incorporated in the LP models.

Due to the centralized static nature of the routing rules above, and the complexity limitations of linear programming, this study should be seen as providing directions for network optimization and not as a tool for actual network operation.

ACKNOWLEDGMENT

We wish to thank Y. Lim and U. Tillman of GTE Laboratories for providing numerical results used in this paper. We also wish to acknowledge the many discussions the authors have had with Alan Pierce and Changhao Zhou, both formerly of GTE Laboratories, J. Morse of GTE Labs., and P. Sen, P. Sarachik, and N. Rikli of Polytechnic University.

REFERENCES

[1] E. A. Harrington, "Voice/data integration using circuit switched networks," *IEEE Trans. Commun.*, vol. COM-28, pp. 781-793, June 1980.
[2] J. S. Turner and L. F. Wyatt, "A packet network architecture for integrated services," in *Proc. GLOBECOM'83*, San Diego, CA, Nov.-Dec. 1983, pp. 2.1.1-2.1.6.
[3] S. R. Amstutz, "Burst switching—An introduction," *IEEE Commun. Mag.*, pp. 36-42, Nov. 1983.
[4] E. F. Haselton, "A PCM frame concept leading to burst switching network architecture," *ICC'83*, Boston, MA, June 1983, pp. E6.7.1-E6.7.6.
[5] C. J. Weinstein, M. L. Malpass, and M. J. Fischer, "Data traffic performance of an integrated circuit- and packet-switched multiplex structure," *ICC'79*, Boston, MA, June 1979, pp. 24.3.1-24.3.5.
[6] C. J. Weinstein, "Fractional speech loss and talker activity model for TASI and for packet-switched speech," *IEEE Trans. Commun.*, vol. COM-26, pp. 1253-1257, Aug. 1978.
[7] J. Morse, "Performance evaluation of burst switched integrated voice-data networks," in *Proc. 11th Int. Teletraffic Congr.*, Kyoto, Japan, Sept. 1985, pp. 155-160.
[8] Y. H. Lim and U. Tillman, "End-to-end speech freeze-out fractions in a network with speech interpolations," *IEEE Trans. Commun.*, vol. COM-34, pp. 1236-1245, Dec. 1986.
[9] ——, private communication.
[10] P. O'Reilly, "Performance analysis of data in burst switching," *IEEE Trans. Commun.*, vol. COM-34, no. 12, Dec. 1986, pp. 1259-1263.
[11] M. Gondran and M. Minoux, *Graphs and Algorithms*. New York: Wiley, 1984.
[12] H. Frank and W. Chou, "Routing in computer networks," *Networks*, vol. 1, pp. 99-112, 1971.

★

Basil Maglaris (S'74-M'79) was born in Athens, Greece, in 1952. He received the undergraduate Diploma degree in electrical engineering from the National Technical University of Athens in 1974, the M.Sc. degree from the Polytechnic Institute of Brooklyn, Brooklyn, NY, in 1975, and the Ph.D. degree in electrical engineering and computer science from Columbia University, New York, NY, in 1979.

From 1979 to 1981 he was with the Network Analysis Corporation, Great Neck, NY, where he was involved in several projects in data and voice networks for both government and industry. In 1981 he joined the Polytechnic Institute of New York, Brooklyn, where he is currently Associate Professor of Electrical Engineering and Computer Science. His research interests focus on the analysis, performance evaluation, and optimization of data, voice and integrated networks, packet ratio, and local area networks.

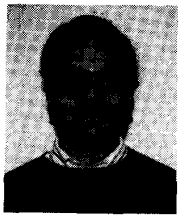
Dr. Maglaris has been involved in various professional activities with the IEEE and the ACM.

Robert R. Boorstyn (S'58–M'59–SM'82–F'86) was born in New York in 1937. He attended the City College of New York and received the B.E.E. degree in 1958. He received the M.S. and Ph.D. degrees in electrical engineering from the Polytechnic Institute of Brooklyn in 1963 and 1966, respectively.

From 1958 to 1961, he worked as an Engineer for the Advanced Studies Department of the Sperry Gyroscope Company. In 1961, he joined the staff of the Department of Electrical Engineering of the Polytechnic Institute of Brooklyn (now Polytechnic Institute of New York), where he is currently a Professor of Electrical Engineering and Computer Science. He is one of the founders of the Polytechnic's New York State Center for Advanced Technology in Telecommunications. From 1977 to 1978 he was on leave at Bell Telephone Laboratories. His research interests are in computer communications networks. Currently he is working on the analysis and design of packet radio networks, on integrated switching technology, on routing in networks, and on network design algorithms.

In 1986 Dr. Boorstyn was named a Fellow of the IEEE for his contributions to the theory and development of multihop packet radio networks. He has been Secretary of the Information Theory Group of the IEEE, Editor for Computer Communications of the IEEE TRANSACTIONS ON COMMUNICATIONS, Chairman of the Computer Communications Committee of the Communications Society of the IEEE, and a member of the Steering Committee of the IEEE INFOCOM Conferences. He is Associate Editor of the *Networks* Journal. From 1972 to 1973 he was a participant in the IEEE Outstanding Lecture Tours Program. He was a member of the delegation to the first joint USSR-IEEE Workshop on Information Theory in Moscow in 1975.

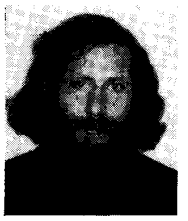
★



Shivendra S. Panwar (S'82–M'85) was born in Delhi, India, on December 15, 1959. He received the B.Tech. degree in electrical engineering from the Indian Institute of Technology, Kanpur, in 1981, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Massachusetts, Amherst, in 1983 and 1986, respectively.

From 1981 to 1985 he was a Research Assistant at the University of Massachusetts. Since 1985 he has been an Assistant Professor in the Department of Electrical Engineering and Computer Science at the Polytechnic University, Brooklyn. He spent the summer of 1987 as a Visiting Scientist at the IBM T. J. Watson Research Center, Yorktown Heights, NY, and has been a Consultant to AT&T Bell Laboratories, Holmdel, NJ. His research interests include multiaccess channels and computer communication networks.

Dr. Panwar is a member of Tau Beta Pi and Sigma Xi. He is currently a member of the IEEE Communications Society Technical Committee on Computer Communications.



Theodore Spritos (S'87–M'90) was born in New York in 1961. He received the B.S. and M.S. degrees, both in electrical engineering from Polytechnic University, Brooklyn, NY, in 1984 and 1985, respectively. From September 1984 until May 1985 he was a Teaching Assistant with the Department of Electrical Engineering at Polytechnic University. Since September 1985 he has been a Research Assistant at Polytechnic University and is working on his Ph.D. degree in the area of integrated networks. His current research interests are in the areas of voice and data communications, queueing theory and signal processing.

★

Peter O'Reilly (S'80–M'83–SM'87) was born in Dublin, Ireland, on June 28, 1948. He received the B.E. degree in electrical engineering from the National University of Ireland in 1969, the M.S. degree in electrical engineering from Colorado State University, Fort Collins, in 1971, and the M.S. and Ph.D. degrees in applied mathematics and electrical engineering, respectively, from the Georgia Institute of Technology, Atlanta, in 1983.

Since 1983 he has been with GTE Laboratories Incorporated, Waltham, MA, where as a member of technical staff he carried out performance and architectural studies of burst switching and other technologies for voice-data integration. He is currently Manager of the Network Performance Analysis Department at GTE Laboratories and is an Adjunct Associate Professor of Electrical Engineering at Worcester Polytechnic Institute. His current research interests include broadband ISDN, network management, local and metropolitan area networks, and protocol/performance tradeoffs. He is a coauthor of the textbook, *Performance Analysis of Local Computer Networks* (Addison-Wesley, 1986).

Dr. O'Reilly is a member of Sigma Xi, SIAM, and the ACM.

★

Carolyn Jack received the B.S. degree in industrial engineering (with highest honors) from Northeastern University in 1984. She is currently working toward the M.S. degree at Northeastern.

Since 1984, she has been a Member of the Technical Staff at GTE Laboratories in Waltham, MA. Her research interests include simulation of networks and queueing systems, operations research, and network capacity planning.

Ms. Jack is a member of the Operations Research Society of America.