Performance Analysis of Deflection Routing with Virtual Circuits in a Manhattan Street Network

Emmanouel A. Varvarigos and Jonathan P. Lang

Department of Electrical and Computer Engineering
University of California
Santa Barbara, CA 93106-9560

Abstract— We propose a new communication protocol for gigabit networks, which we call the virtual circuit deflection (abbreviated VCD) protocol, that combines some of the individual characteristics of virtual circuit switching and deflection routing. Its advantage over previous deflection routing schemes is that deflections in the VCD protocol occur on a per session basis, instead of on a per packet basis, making packet resequencing at the destination considerably easier to accomplish. The VCD protocol exploits the storage arising from the high bandwidth-delay product of optical fibers, and it provides lossless communication with minimal buffering at the switches and without the need for advance reservations. The VCD protocol appears to be particularly suitable for networks that use optical switching, where buffers are expensive to implement with current optical technology. Indeed, the VCD protocol requires only limited buffering, which can be implemented using a minimal number of optical delay lines. We analyze the performance of the VCD protocol for the MS network topology by using new analytical models.

I. Introduction

The VCD protocol is a virtual circuit switching protocol of the *tell-and-go* variety (see [VaS95]), where data starts being transmitted shortly after the set-up packet of the session is sent. A preferred path is selected at the source, and a set-up packet is sent on that path to establish the connection, followed after a short delay (much shorter than the end-to-end round-trip delay required by wait-for-reservation type of protocols [ACG90], [CiG88]) by the data packets, avoiding in this way the pre-transmission delay associated with end-to-end reservations. If the capacity available at a preferred intermediate link is insufficient to accommodate the session, the set-up packet and the data packets that follow it may have to be routed over a different, longer path; we then say that the session is *deflected*. As long as the total outgoing link capacity is (greater than or) equal to the total incoming link capacity of a node, we will see that adequate capacity can always be made available on the outgoing links of an intermediate node to accommodate a new incoming session. This, however, may happen at the expense of interrupting (preempting) existing sessions that originate at that node, and/or splitting the new session into smaller subsessions, each of which follows a different path.

An important advantage of the proposed VCD protocol over previous deflection routing protocols (such as packet-by-packet [Max90] and loop deflection schemes [HaC90]) is that it significantly reduces the need for packet resequencing at the destination. This is because deflections in the former occur on a session-by-session basis while in the latter they occur on a packet-by-packet basis. Consequently, message reassembly at the destination, which is one of the main problems of deflection schemes [Max90], is easier with the VCD protocol. When a session is split, blocks of data have to be resequenced, instead of individual packets. This is important for multigigabit networks, where a session may involve the transfer of millions of packets. Moreover, in the VCD protocol, data packets are routed through a switch without involving the processor, based on the virtual circuit identifier (VCI) they carry and the routing tables established by the set-up packet. By contrast, in the deflection protocols proposed to date, routing decisions are made individually for each data packet, making the switch processor a potential bottleneck of the design.

Traffic in high-speed networks can be switched either optically, or electronically. Optical switching is generally considered incompatible with packet switching because efficient packet switching requires substantial packet storage, which is difficult to achieve with current optical technology. The VCD protocol provides lossless communication for data streams that are nearly uniform with minimal buffer space at the intermediate nodes. A particular implementation of the VCD protocol for networks using optical switching, which employs a small number of optical delay lines to perform the buffering function, is presented in [VaL96].

We analyze analytically and through simulation the performance of the VCD protocol for the Manhattan Street (MS) network topology, under the assumption that all sessions have equal rates, and their source and destination nodes are uniformly distributed over all nodes of the network. We obtain results on the throughput and the average number of deflections, as a function of the network load, the size of the network, and the link capacities. Deflection routing protocols have previously been analyzed by several researchers, under a variety of assumptions on the underlying network topology (see [GrG88], [GrH92], [Max89], [Max90], [Bra91], and [HaC90]). Our model, analysis, and results are considerably different than those presented in previous works, where only packet-by-packet (datagram) deflections, instead of session (virtual circuit) deflections, were considered. As a result, session durations and rates played no role

0-7803-3336-5/96 $5.00

in these works, and packet arrivals at a node and their destinations could be assumed to be independent. This is very different from our model, where we focus on sessions (virtual circuits) instead of packets (datagrams), and the previous assumptions are no longer valid.

II. The Virtual Circuit Deflection Protocol

In this section, we describe the VCD protocol and show how it can be combined with other techniques to meet its objectives.

A path with adequate residual capacity is first computed at the source based on the topology and link utilization information available at the source at that time. A set-up packet is then transmitted over the path to set the routing tables, followed after a short delay by the data packets. If the set-up packet is successful in reserving capacity on all of the links on the path to the destination, the VCD protocol looks like the usual reservation protocols, with the difference that the reservation (set-up) phase and the transmission phase overlap in time. If the residual capacity on a link is not sufficient to accommodate the new session, the session may have to be deflected and/or split into smaller subsessions, as described below.

We focus on a particular intermediate node, where a set-up packet arrives requesting rate r. We let R, F, G, be the total capacity occupied by transit, terminating, and initiating sessions, respectively, at that node that are in progress when the set-up packet arrives. We also let A and B be the total unused capacity on the outgoing and the incoming links of the node, respectively. Since the total incoming capacity is equal to the total outgoing capacity of a node, we have $C_{total} = r + R + B + F = R + A + G$, which implies

$$A + G \geq F + r \geq r. \tag{1}$$

Therefore, a set-up packet that arrives at an intermediate node requesting rate r, can always find capacity equal to r to reserve on the outgoing links of the node. This may, however, require the interruption (preemption) of one or more of the existing sessions that initiate at that node. It is possible that the outgoing capacity that is available, or that may become available through the preemption of existing sessions originating at a node, may not all belong to the same outgoing link of the node. In that case the session may have to be split into two or more subsessions of smaller rates, each of which is routed over a different path to the destination. Sessions that are interrupted may resume transmission when the session that preempted them ends (either because it is completed, or because a control packet is sent to its source requesting it to pause). When a session is split into a total of, say, k subsessions, packets belonging to different subsessions may arrive at the destination out of order; packets, however, belonging to the same subsession will always arrive in the correct order. Resequencing k blocks of packets (each of which is ordered) at the destination is much easier to accomplish than resequencing individual packets. This is one of the main advantages of the VCD protocol over other deflection protocols, where packets are deflected independently of each other, and the order of the packets in a session may be completely destroyed.

The VCD protocol is designed to provide lossless communication for sessions that have constant rate, or sessions that have certain smoothness properties (see [Gol91]), or sessions that have variable rate but can tolerate the delay induced when transforming them into smooth sessions through the use of input flow control. Constant-rate sessions can clearly be switched with minimal buffer space at the nodes. If more burstiness is allowed, additional buffer space, which depends on the degree of burstiness, is required to provide lossless communication. It can be shown that the VCD protocol can be combined with stop-and-go queueing (see [Gol91]) to provide lossless communication with minimal buffer requirements for sessions that have some particular smoothness properties (see [VaL96]).

III. Analysis for the Manhattan Street Network

We will assume that the underlying topology is a square Manhattan Street (abbreviated MS) network, with $X = Y = \sqrt{N}$ nodes along each dimension. External session (connection) requests are generated at each node over an infinite time horizon according to a Poisson process of rate λ, and their destinations are uniformly distributed over all nodes of the network. All sessions have rate equal to one unit, and their holding times are independent and exponentially distributed with mean $1/\mu$. The capacity of each link is taken to be equal to m units. Since the session rates are equal to one unit and the uncommitted capacity on a link is always an integer number of units, sessions do not have to be split, and all packets of a session arrive at their destination in the correct order. It is still possible, however, for sessions to be deflected and/or preempted.

A session using a given link l is called an *originating session* if l is the first link on the session's path, and a *transit session* if l is an intermediate link. A session that reaches its destination over link l is called a *terminating session* for link l. When both of the outgoing links of a node lie on a shortest path to the destination, then the node is called a *don't care* node for that destination; otherwise it is called a *preference* node. A set-up packet selects a preferred link according to the following rule:

Persistent rule: If the current node is a "don't care" node, one of the links is chosen with equal probability as the preferred one. If the current node is a "preference" node, the preferred link is the one that lies on the shortest path.

A transit set-up packet attempts to reserve capacity on its preferred link, preempting if necessary a session originating at that link. If this is not possible, the session is routed over the other link of the node, preempting if necessary some session originating on that link. An originating session is accepted only if there is capacity available on its preferred link to accommodate it; that is, sessions are never deflected on their first hop. An originating session that is not accepted is said to be blocked. Sessions that are preempted or blocked

are randomly mixed back into the input queues so that the combined process of exogenous and retrial set-up attempts can be approximated by a Poisson process.

We focus on sessions with destination $(0,0)$, and let $\overline{D}(i,j)$ [or $D(i,j)$] be the average number of additional links that will be used by a transit [or originating, respectively] set-up packet currently located at node (i,j), whose destination is node $(0,0)$. We let p be the probability that an arriving transit set-up packet fails to reserve the required capacity on its preferred outgoing link (therefore, such a set-up packet is deflected if the current node is a preference node). We then have

$$\overline{D}(i,j) = 1 + \begin{cases} \frac{1}{2}[\overline{D}(i_1,j_1) + \overline{D}(i_2,j_2)], \\ \quad \text{if } (i,j) \text{ is a don't care node;} \\ (1-p)\overline{D}(i_1,j_1) + p\overline{D}(i_2,j_2), \\ \quad \text{if } (i,j) \text{ is a preference node, and} \\ \quad (i_1,j_1) \text{ is the preferred next node,} \end{cases} \tag{2}$$

and

$$D(i,j) = 1 + \begin{cases} \frac{1}{2}[\overline{D}(i_1,j_1) + \overline{D}(i_2,j_2)], \\ \quad \text{if } (i,j) \text{ is a don't care node;} \\ \overline{D}(i_1,j_1), \\ \quad \text{if } (i,j) \text{ is a preference node, and} \\ \quad (i_1,j_1) \text{ is the preferred next node,} \end{cases} \tag{3}$$

where (i_1,j_1) and (i_2,j_2) are the outgoing neighbors of (i,j). Also, we clearly have $D(0,0) = \overline{D}(0,0) = 0$. If the deflection probability p is known, the preceding equations can be applied iteratively on the MS network to calculate $D(i,j)$ and $\overline{D}(i,j)$ for all nodes (i,j). The total average number of links used by a session can then be obtained as

$$D = \frac{1}{N-1} \sum_{(i,j)\neq(0,0)} D(i,j). \tag{4}$$

Transit set-up packets that arrive on a horizontal (or vertical) link and select according to the persistent rule the horizontal (or vertical) link as their preferred outgoing link are called *straight-through* set-up packets. We let $\overline{\theta}(i,j)$ be the average number of additional nodes at which a transit set-up packet currently at node (i,j) will have a straight-through horizontal preference until it reaches its destination node $(0,0)$. Using the symmetry of the MS network, we have

$$\overline{\theta}(i,j) = \begin{cases} \frac{1}{2}[1 + \overline{\theta}(i_1,j_1) + \overline{\theta}(j_2,i_2)], \\ \quad \text{if } (i,j) \text{ is a don't care node;} \\ 1 + (1-p)\overline{\theta}(i_1,j_1) + p\overline{\theta}(j_2,i_2), \\ \quad \text{if } (i_1,j_1) \text{ is the preferred next node;} \\ p\overline{\theta}(i_1,j_1) + (1-p)\overline{\theta}(j_2,i_2), \\ \quad \text{if } (i_2,j_2) \text{ is the preferred next node,} \end{cases} \tag{5}$$

where (i_1,j_1) and (i_2,j_2) are the horizontal and vertical neighbors of (i,j), respectively. Also, we clearly have $\overline{\theta}(0,0) = 0$. The average probability of a straight-through preference is

$$\theta = \frac{1}{(N-1)(D-1)} \sum_{(i,j)\neq(0,0)} \theta(i,j),$$

where

$$\theta(i,j) = \begin{cases} \frac{1}{2}[\overline{\theta}(i_1,j_1) + \overline{\theta}(j_2,i_2)] \\ \quad \text{if } (i,j) \text{ is a don't care node;} \\ \overline{\theta}(i_1,j_1), \\ \quad \text{if } (i_1,j_1) \text{ is the preferred next node;} \\ \overline{\theta}(j_2,i_2), \\ \quad \text{if } (i_2,j_2) \text{ is the preferred next node,} \end{cases} \tag{6}$$

where (i_1,j_1) and (i_2,j_2) are the horizontal and vertical neighbors of (i,j), respectively.

We denote by B the probability that a new session is blocked, and by E the probability that a session is interrupted (preempted) before it is completed. We assume that the retransmissions of sessions that are blocked or preempted are sufficiently randomized so that the total arrival rate of originating sessions requesting a particular outgoing link of a node is a Poisson process with rate

$$\lambda_1^* = \frac{\lambda}{2(1-B)(1-E)}. \tag{7}$$

Since the average number of intermediate links used by a session is equal to $D-1$, the average rate with which transit set-up packets are emitted on a link is

$$\lambda_2 = \lambda_1^*(1-B)(D-1) = \frac{\lambda(D-1)}{2(1-E)}. \tag{8}$$

Also, the average rate with which terminating set-up packets arrive at a node is $\lambda_3 = \lambda_1$.

We will say that a node is in state $\overline{X} = (X_a, X_b, X_c, X_d, X_{ab}, X_{ad}, X_{cb}, X_{cd})$, if there are X_a (or X_c) sessions terminating over its horizontal (vertical, respectively) incoming link, X_b (or X_d) sessions originating on its horizontal (vertical, respectively) outgoing link, X_{ab} (or X_{cd}) transit sessions arriving over the horizontal (or vertical) incoming link and leaving over the horizontal (or vertical, respectively) outgoing link, and X_{ad} (or X_{cb}) transit sessions arriving over the horizontal (or vertical) incoming link and leaving over the vertical (or horizontal, respectively) outgoing link. We also let $\pi(\overline{X})$ be the steady-state probability that a node is in state $\overline{X}$, and we will approximate $\pi(\overline{X})$ as the stationary distribution of an auxiliary system $\hat{\mathcal{Q}}$.

We also ask that the rate λ_2 at which transit set-up packets are emitted on a link of the MS network is the same as the rate at which transit customers are accepted in system $\hat{\mathcal{Q}}$. For this to hold, we should have

$$\lambda_2^* = \frac{\lambda_2}{1 - \Pr(\overline{X} :\ X_a + X_{ab} + X_{ad} = m)}.$$

Furthermore, we ask that the rate at which terminating packets are received at an incoming link of the MS network is the same with the rate at which terminating customers are accepted in system $\hat{\mathcal{Q}}$. This happens when λ_3^* is defined as

$$\lambda_3^* = \frac{\lambda_3}{1 - \Pr(\overline{X} :\ X_a + X_{ab} + X_{ad} = m)}.$$

The blocking probability for the first hop is

$$B = \sum_{\overline{X}:\ X_b+X_{ab}+X_{cb}=m} \pi(\overline{X}). \quad (9)$$

Assuming that an arriving transit set-up packet finds a node in a typical state, except for states where $X_a + X_{ab} + X_{ad} = m$, the deflection probability p is given by

$$p = \sum_{\substack{\overline{X}:\ X_{ab}+X_{cb}=m,\\ X_a+X_{ab}+X_{ad}\neq m}} \frac{\theta\pi(\overline{X})}{\sum_{\overline{X}:\ X_a+X_{ab}+X_{ad}\neq m}\pi(\overline{X})} + \sum_{\substack{\overline{X}:\ X_{ad}+X_{cd}=m,\\ X_a+X_{ab}+X_{ad}\neq m}} \frac{(1-\theta)\pi(\overline{X})}{\sum_{\overline{X}:\ X_a+X_{ab}+X_{ad}\neq m}\pi(\overline{X})}. \quad (10)$$

The probabilistic rate ϵ at which a particular transit session $\mathcal{S}$ is preempted due to arrivals of set-up packets at its source can also be found to be

$$\epsilon = \lambda_2^* \Bigg[\sum_{\substack{\overline{X}:\ X_b+X_{ab}+X_{cb}=m,\\ X_a+X_{ab}+X_{ad}\neq m,\\ X_b\neq 0}} \frac{1}{X_b}\theta\frac{\pi(\overline{X})}{\sum_{\overline{X}:\ X_b\neq 0}\pi(\overline{X})} + \sum_{\substack{\overline{X}:\ X_b+X_{ab}+X_{cb}=m,\\ X_a+X_{ab}+X_{ad}\neq m,\\ X_b\neq 0,\\ X_{ad}+X_{cd}=m}} \frac{1}{X_b}(1-\theta)\frac{\pi(\overline{X})}{\sum_{\overline{X}:\ X_b\neq 0}\pi(\overline{X})} + \sum_{\substack{\overline{X}:\ X_b+X_{ab}+X_{cb}=m,\\ X_c+X_{cb}+X_{cd}\neq m,\\ X_b\neq 0}} \frac{1}{X_b}(1-\theta)\frac{\pi(\overline{X})}{\sum_{\overline{X}:\ X_b\neq 0}\pi(\overline{X})} + \sum_{\substack{\overline{X}:\ X_b+X_{ab}+X_{cb}=m,\\ X_c+X_{cb}+X_{cd}\neq m,\\ X_b\neq 0,\\ X_{ad}+X_{cd}=m}} \frac{1}{X_b}\theta\frac{\pi(\overline{X})}{\sum_{\overline{X}:\ X_b\neq 0}\pi(\overline{X})} \Bigg]. \quad (11)$$

The probability E that a session $\mathcal{S}$ that has been accepted is preempted before it is completed can be approximated as

$$E = \frac{\epsilon}{(\epsilon+\mu)}. \quad (12)$$

To calculate the steady-state probabilities $\pi(\overline{X})$ for all feasible states, we write down the global balance equations of the Markov chain that corresponds to the auxiliary system $\hat{Q}$. If the parameters λ_1^*,λ_2^*, λ_3^*, and ϵ are known, then the global balance equations together with the equation $\sum\pi(\overline{X}) = 1$ give the steady-state probabilities. These parameters, however, depend on the values of the steady-state probabilities. Equations (2)-(12) together with the global balance equations give a system of equations that can be jointly solved by using the method of successive approximations.

IV. Analytical and Simulation Results

We define the *inefficiency ratio* $\eta(\lambda)$ as the ratio

$$\eta(\lambda) = \frac{D(\lambda)}{D(0)} \quad (13)$$

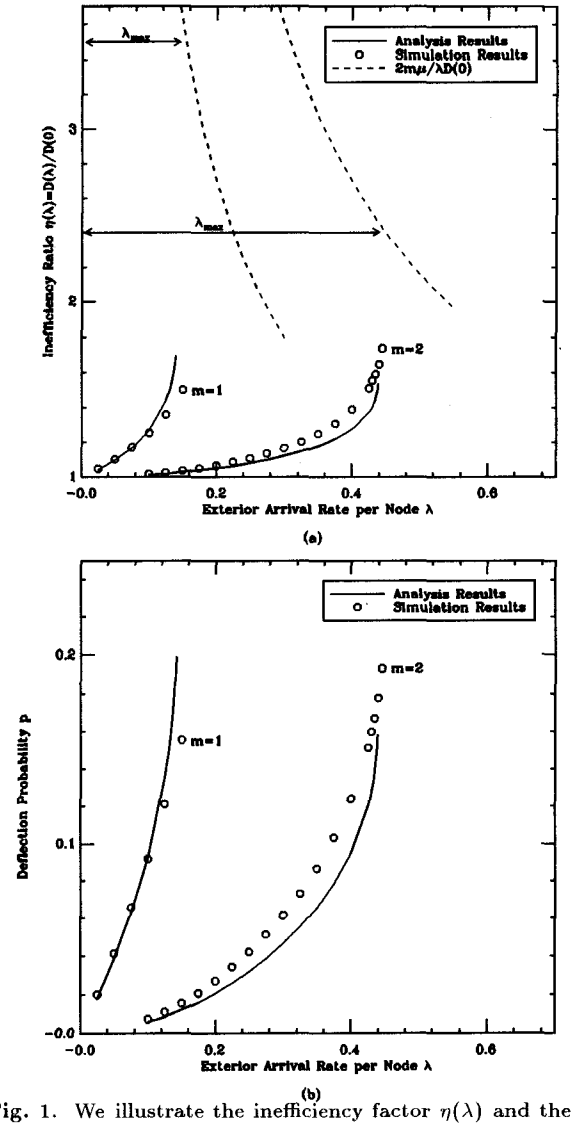

Fig. 1. We illustrate the inefficiency factor $\eta(\lambda)$ and the deflection probability p, as a function of the external arrival rate per node λ, for a 6×6 MS network with capacities $m = 1$ and $m = 2$. We also illustrate the upper bound $\frac{2m\mu}{\lambda D(0)}$ on $\eta(\lambda)$, and the stability region.

of the average path length $D(\lambda)$ taken by a session for a given arrival rate λ, over the average shortest-path length $D(0)$ of the MS network topology. In Fig. 1 we illustrate $\eta(\lambda)$ and the deflection probability p as a function of the external arrival rate λ per node, for a 6×6 MS network, average session duration $1/\mu = 1$, and different values of the link capacity m.

A necessary condition for stability can easily be found to be

$$\eta(\lambda) \leq \frac{2m\mu}{\lambda D(0)}. \quad (14)$$

The dashed lines in Fig. 1a correspond to the upper bounds on $\eta(\lambda)$ given by the right hand side of Eq. (14). The stability region (equivalently, the maximum value of λ) can be approximately obtained by finding graphically the point at which the curves $\eta(\lambda)$ and $\frac{2m\mu}{\lambda D(0)}$ intersect.

The inverse of the inefficiency ratio when operating at maximum load is illustrated in Fig.2 for different values of

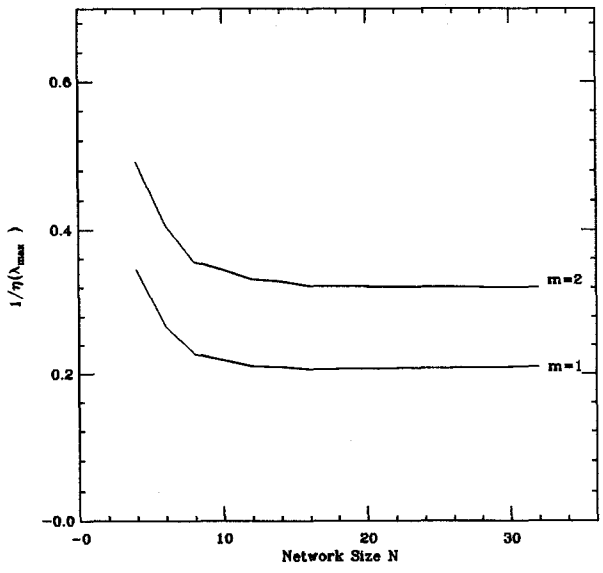

Fig. 2. We illustrate the saturation throughput $1/\eta(\lambda_{max})$ as a function of the network size N, for various values of the link capacity m.

the network size N and the link capacity m. One can view $1/\eta(\lambda_{max})$ as the maximum fraction of the capacity that can be effectively utilized by the VCD protocol when new sessions are always available to enter the network (capacity is not effectively utilized when it is wasted due to deflections or stays idle); we refer to $1/\eta(\lambda_{max})$ as the *(normalized) saturation throughput*. As shown in Fig.2, $1/\eta(\lambda_{max})$ increases when the link capacity m increases. Therefore, an increase in the link capacities does not only increase the available network capacity, but also the efficiency with which this capacity is used (through a reduction in the deflection probability and the average path length).

V. Conclusions

The virtual circuit deflection protocol presented in this paper compares favorably to wait-for-reservation and backpressure-based protocols, since it can provide lossless communication with little buffering at the switches and minimal pre-transmission delay. The VCD protocol is a hybrid of virtual circuit switching and deflection routing, and combines some of their important individual advantages. The VCD protocol alleviates to a large extent the resequencing problem associated with other deflection routing schemes. Its small buffer requirements, make it particularly appropriate for multigigabit networks that use optical switching. We have presented analytical and simulation results on the throughput, the average path length, and other performance parameters of interest for the VCD protocol in a Manhattan Street (MS) network. We believe that the results obtained are indicative of the performance of the protocol for other topologies of interest (provided that they offer a large number of alternative paths between nodes), and they indicate that the VCD protocol is a potentially interesting connection and flow control protocol for multigigabit and general data networks.

VI. Reference

[ACG90] Awerbuch, B., Cidon, I., Gopal, I., Kaplan, M., and Kutten, S., "Distributed control for PARIS," in Proc. 9th Annu. ACM Symp. on Principles of Distributed Comp., 1990, pp. 145-160.

[Bra91] Brassil, J. T., Deflection Routing in Certain Regular Networks, Ph.D. Thesis, UC San Diego, 1991.

[CiG88] Cidon, I., and Gopal, I., "PARIS: An Approach to Integrated High-speed Private Networks," Int. J. Digital and Analog Cabled Syst., Vol. 1, No. 2, pp. 77-86, 1988.

[Gol91] Golestani, S. J., "Congestion-Free Communication in High-Speed Packet Networks," IEEE Trans. on Communications, Vol. 39, No. 12, December 1991.

[GrG88] Greenberg, A. G., and Goodman, J., "Sharp Approximate Models of Adaptive Routing in Mesh Networks," in Teletraffic Analysis and Computer Performance Evaluation, pp. 255-270, Amsterdam, 1988.

[GrH92] Greenberg, A. G., and Hajek, B., "Deflection Routing in Hypercube Networks," IEEE Trans. Communications, Vol. COM-35, No. 6, pp. 1070-1081, June 1992.

[HaC90] Haas, Z., and Cheriton, D. R., "Blazenet: A Packet-Switched Wide-Area Network with Photonic Data Path," IEEE Trans. on Communications, Vol. 38, No. 6, pp. 818-829, June 1990.

[Max89] Maxemchuk, N. F., "Comparison of Deflection and Store-and-Forward Techniques in the Manhattan Street and Shuffle-Exchange Networks," in INFOCOM '89, Vol. 3, pp. 800-809, April 1989.

[Max90] Maxemchuk, N. F., "Problems Arising from Deflection Routing: Live-lock, Lock-out, Congestion and Message Reassembly," Proc. NATO Workshop on Archit. and High Perform. Issues of High Capacity LANs and MANs, 1990.

[VaS95] Varvarigos, E. A., and Sharma, V., "The Ready-to-go Virtual Circuit Protocol: A Loss-free Connection Control Protocol for Gbit Networks," submitted to IEEE/ACM Trans. on Networking.

[VaL96] Varvarigos, E. A., and Lang, J. P., "A Novel Virtual Circuit Deflection Protocol for Multigigabit Networks and its Performance for the MS Topology," submitted to IEEE/ACM Trans. on Networking.