Traffic studies of unbuffered Delta networks

by P. Heidelberger
P. A. Franaszek

This paper analyzes the performance of unbuffered Delta networks under a nonuniform ("hot-spot") traffic pattern. Particular attention is paid to characterizing the overflow traffic of unsuccessfully transmitted packets. Analytic techniques are used to show that the overflow traffic from an unbuffered packet-switched Delta network is (fractionally) hotter than the offered load. Simulation techniques are used to extend this result to an unbuffered circuit-switched network with limited retrials. In addition, the distribution of the number of trials until a "cold" packet is successfully delivered is shown to have a decreasing hazard rate, which means that it becomes less and less likely with each successive trial that a packet is delivered successfully. The implications of these results for hierarchical networks, a class of networks for interconnecting a highly parallel, shared-memory multiprocessor computer system, are discussed.

1. Introduction

This paper considers performance issues relevant to the design of hierarchical interconnection networks, a class of interconnection networks which were first described by Franaszek in [1]. Such networks have been proposed for use in highly parallel, shared-memory computer architectures. The basic idea is to provide a hierarchy of paths in which the first (Level 1) path is extremely fast, i.e., has very low latency, but does not provide guaranteed message delivery. Other paths in the hierarchy have (potentially) higher latency, but provide an increased probability of successful message delivery. A final path in the hierarchy provides guaranteed delivery. A properly designed hierarchy is one in which messages are successfully delivered on the fast (Level 1) path with high probability, so that the slower paths are used infrequently. Further rationale for such networks is given in [1], and implementation issues are discussed in [2]. In its simplest form, the hierarchy consists of two levels that are constructed from two physically separate interconnection networks. Delivery of packets is tried first on the Level 1 network. If the packet is not delivered successfully (because of contention in the network), it may be retried on the Level 1 network, according to some retrial protocol, or delivered on the Level 2 network.

The performance of a system with a hierarchical interconnection network in which the Level 1 network is a crossbar was analyzed in [3]. Because of the growth limitations of large crossbars, we focus on issues related to the performance of the Level 1 network under the assumption that it is a multistage interconnection network

(MIN), specifically an unbuffered Delta network (see, e.g., [4] or [5]). Briefly, a Delta network is a multiple-stage network with a unique path between each source and destination pair. The switches in each stage are crossbars. Actually, the particular network we consider is a baseline network. Since a baseline network is topologically equivalent to a Delta network (see the discussion and references in [5]), and because the term *Delta network* is more commonly found in the literature, we refer to this network as a Delta network. A four-stage baseline network using 2×2 switches is shown in **Figure 1**.

Some justification for selecting such an unbuffered network is in order. Because this type of network provides a fairly simple function, it can be made quite fast. For example, since no internal buffers or queueing logic need be implemented, each stage of the network can be implemented with a relatively few levels of logic. This allows the network to be built with fast $k \times k$ switches, where k is fairly large; therefore, the number of stages grows slowly with system size. (In such a switch, messages enter the switch on one of k input ports and exit on one of k output ports.) For example, in this paper we consider $k = 8$ and $m = 3$ stages, corresponding to a 512×512-way system. In contrast, a buffered MIN requires more levels of logic to implement the necessary queueing-discipline logic (resulting in slower switches), as well as more circuits to implement the buffers (resulting in a smaller value of k and a greater number of stages m). The combined effect is that the buffered network has a (potentially) much greater latency than the unbuffered network. For example, to overcome this problem, the (buffered) switch of the IBM RP3 computer [6] was designed to use a faster technology than RP3's processors. We note that the BBN Butterfly* parallel processor uses an unbuffered MIN constructed of 4×4 switches [7, 8].

Performance studies of MINs abound. Analyses of *buffered* MINs may be found in [9–15]. Of particular recent interest is the effect of nonuniform traffic patterns in MINs, with the so-called "hot-spot" model introduced in [16] receiving special attention (see [17–20]). We assume that, in the MIN, sources correspond to processors and destinations correspond to memory modules. In the hot-spot model, traffic is uniformly distributed over the memories, except that a particular (hot) module is selected with higher probability. We call the uniformly distributed traffic "cold" and the additional traffic destined for the hot module "hot." With finite buffers, the hot spot leads to an effect called "tree saturation," in which buffers quickly become filled and the performance of the entire network is severely degraded. When the hot spot is due to interprocess synchronization (e.g., barrier synchronization or allocation of iterations in parallel do-loops), messages

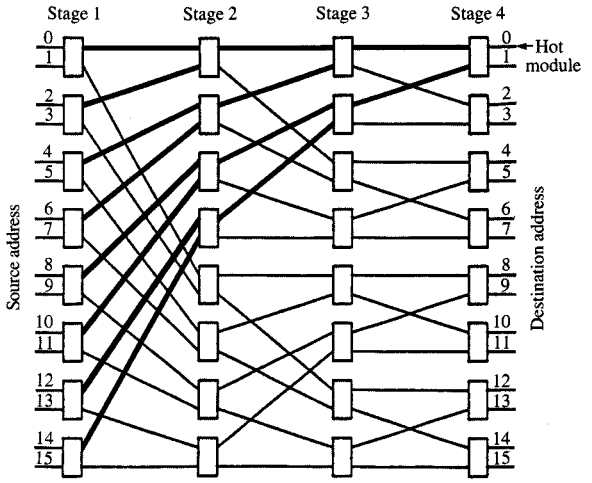


Figure 1

A 16×16 baseline network with 2×2 switches.

can sometimes be combined within the network to reduce the load on the hot memory module and ease the tree-saturation effect (see [6] or [21]). Performance studies on the effectiveness of combining may be found in [16, 22, 23]. Software combining techniques that avoid the use of complex hardware combining switches have been described in [24].

The first performance analysis of *unbuffered* MINs was by Patel [25], who considered a packet-switched unbuffered Delta network with uniform traffic. Patel's analysis has been extended to handle a number of other situations (see for example [26–31]). In particular, Liu [30] studied the packet-switched Delta network with a hot spot. In [30] and [31] it is shown that the hot-spot effect is quite different in unbuffered networks and that only packets destined for the hot module (or its close neighbors) are adversely affected by the hot spot. These analytic results have been confirmed by measurements on the BBN Butterfly [7, 8]. Several configurations of circuit-switched MINs under a uniform traffic model are analyzed in [32].

In most of the above models of unbuffered networks, it is assumed that blocked packets are rejected, although it is sometimes assumed that they are resubmitted and that the offered load therefore includes both new and resubmitted packets. Thus, while the overflow traffic is not typically of interest in these analyses, in the case of hierarchical networks, the overflow traffic is of great interest, since it forms the offered load to the higher-latency Level 2 network. (The Level 1 overflow traffic may form only part

*Butterfly is a trademark of BBN Advanced Computers, Inc., 10 Fawcett St., Cambridge, MA 02238.

of the Level 2 offered load if, for example, all synchronization traffic is routed through the Level 2 network.) In this paper, besides the standard issues of determining the throughput, probability of successful transmission, average delay, expected number of trials, and similar characteristics, we focus on characterizing the overflow traffic of packets that are not successfully transmitted over the Level 1 path in the hierarchy. We limit our discussion to the effects of spatial nonuniformity in the traffic pattern and do not consider temporal nonuniformity. Under the hot-spot model, we show that the Level 1 network effectively filters out the "cold" (i.e., uniformly distributed) traffic from the arrival process to the Level 2 network. The result is that, while the total throughput of overflow traffic is relatively low, the overflow traffic is extremely hot—much hotter, in fact, than the original offered load. More precisely, the fraction of the overflow traffic that is hot is much greater than the fraction of the offered load that is hot.

In Section 2 this result is established analytically for a simple model of a packet-switched network without retrials. In Sections 3 and 4, we present the results of simulation studies that demonstrate this effect for a more realistic circuit-switched network with retrials. Other performance issues are also discussed in Section 4, including the observation that with hot spots, the (conditional) probability of successful transmission on the jth trial is a decreasing function of j, corresponding to a Decreasing Failure Rate (DFR) distribution (see [33]). Thus, in a hierarchical network, there is only limited value in employing a retrial protocol that relies on a large number of retrials. These results are summarized and their implications are discussed in Section 5.

2. Analytic result for a packet-switched network

In this section we consider the overflow traffic from an unbuffered packet-switched Delta network without retrials. Patel [25] considered the case of uniformly distributed traffic. In his discrete time model, the arrival traffic is Bernoulli, and rejected packets are lost; i.e., on each cycle a packet is independently generated at each processor with probability $p = p_0$. As packets move through the network, if a collision occurs, one of the packets is randomly selected as the winner, and the other colliding packets are rejected. The winning packet is then forwarded to the next stage, and the port at the current stage is released. For a system consisting of $k \times k$ switches, Patel determined a recursion for p_n, the probability that a packet is emitted from a port in stage n on any given cycle: $p_{n+1} = 1 - (1 - p_n/k)^k$. Kruskal and Snir [11] developed an asymptotic expansion for p_n (to first order, $p_n \propto 2k/[n(k-1)]$). Liu [30] developed recursions for the Delta network with a hot spot.

Building on Liu's recursions, we show that in such a network consisting of 2×2 switches, the overflow traffic is (fractionally) hotter than the offered load. To make this statement more precise, we need to introduce some notation. We again assume Bernoulli arrivals; i.e., a new packet is generated with probability $p = p_c + p_h$, where p_c is the probability of generating a "cold" packet whose destination is uniformly distributed over all the memory modules, and p_h is the probability of generating a "hot" packet that is destined for a particular (hot) module, say module 0. We assume there are m stages and P $(= 2^m)$ processors and P memories. Note that the total probability of generating a packet for memory module 0 is $p_h + (p_c/P)$. Let $\lambda_c^m(i)$ be the steady-state throughput (arrival rate) of cold traffic at module i (in packets per cycle), and define $\lambda_c^m = \sum_{i=0}^{P} \lambda_c^m(i)$. Then $\alpha_c^m = Pp_c - \lambda_c^m$ is the steady-state overflow throughput of (rejected) cold packets. Similarly, define λ_h^m to be the steady-state throughput of hot traffic at module 0 and $\alpha_h^m = Pp_h - \lambda_h^m$ to be the steady-state overflow throughput of (rejected) hot packets. The fraction of the offered load that is hot is $p_h/(p_c + p_h)$, and the fraction of the overflow traffic that is hot is $\alpha_h^m/(\alpha_c^m + \alpha_h^m)$. We establish that

$$\frac{\alpha_h^m}{\alpha_c^m + \alpha_h^m} \geq \frac{p_h}{p_c + p_h}. \tag{1}$$

As shown in Figure 1, we assume that hot (destination 0) traffic is always routed up, according to the baseline network topology. As discussed in [17–19, 30], the memories can be partitioned into different classes corresponding to the number of stages shared with the hot traffic. Let C_j be the set of memories sharing output ports with the hot traffic at j stages. Then $|C_0| = P/2$, $|C_1| = P/4$, etc. For example, in Figure 1 ($P = 16$) the paths from any processor to the hot memory module are indicated by darkened lines. Traffic destined for memory module 1 shares three stages of output ports with the traffic destined for the hot module (the output ports in stages 1, 2, and 3); thus $C_3 = \{1\}$. Similarly, traffic destined for memory modules 2 and 3 shares two stages of output ports with the traffic destined for the hot module (the output ports in stages 1 and 2); thus $C_2 = \{2, 3\}$. We similarly obtain $C_1 = \{4, 5, 6, 7\}$ and $C_0 = \{8, \cdots, 15\}$.

Now consider a switch at stage n along a path toward the hot module. Let c_{n-1} be the probability that a cold packet is output from an output port along this path at stage $n-1$, and let h_{n-1} be the probability that a hot packet is output from an output port along this path at stage $n-1$. Let c_n be the probability that a cold packet is output from the top (hot) port at stage n, and let c_n' be the probability that a cold packet is output from the bottom (cold) port at stage n. Starting with $c_0 \equiv p_c$ and $h_0 \equiv p_h$, we have the recursions

$$c_n = c_{n-1}\left(1 - \frac{h_{n-1}}{2} - \frac{c_{n-1}}{4}\right),$$

$$h_n = h_{n-1}\left(2 - h_{n-1} - \frac{c_{n-1}}{2}\right),$$

$$c'_n = c_{n-1}\left(1 - \frac{c_{n-1}}{4}\right). \quad (2)$$

Note that the expression for c'_n is the same as Patel's recursion for uniform traffic and $k = 2$. The throughput of each of the memories in class C_{n-1} can then be obtained by applying Patel's recursion to a network with $(m - n)$ stages and initial probability $p_0 = c'_n$. This formulation is somewhat different from Liu's, but the two yield the same results. In the above notation, $h_m = \lambda_h^m$ and $c_m = \lambda_c^m(0)$. By using the recursions of Equations (2), it is straightforward to show that if $p_h > 0$ and $p_c > 0$, then $h_n > h_{n-1}$, and $\lambda_c^m(i)$ is a nondecreasing function of i (actually, $\lambda_c^m(i)$ takes on the same value for all $i \in C_j$, and $\lambda_c^m(i) < \lambda_c^m(i')$ if $i \in C_j$ and $i' \in C_{j-1}$).

We now prove Inequality (1), which is equivalent to $p_h\alpha_c^m \le p_c\alpha_h^m$. Since $\alpha_c^m = Pp_c - \lambda_c^m$ and $\alpha_h^m = Pp_h - \lambda_h^m$, this in turn is equivalent to $\lambda_h^m p_c \le \lambda_c^m p_h$. Since $p_c = c_0$, $p_h = h_0$, $\lambda_h^m = h_m$, and $2^m c_m = 2^m \lambda_c^m(0) \le \lambda_c^m$, Inequality (1) will be true, provided $h_m c_0 \le 2^m c_m h_0$. The proof is by induction. Using Equations (2) with $m = 1$, we actually obtain equality; i.e., $h_1 c_0 = 2c_1 h_0 = h_0 c_0(2 - h_0 - c_0/2)$. For $m > 1$, assume that $h_m c_0 \le 2^m c_m h_0$. Using the recursion for h_{m+1}, we obtain

$$h_{m+1}c_0 = c_0 h_m\left(2 - h_m - \frac{c_m}{2}\right) \le 2^m c_m h_0\left(2 - h_m - \frac{c_m}{2}\right)$$

$$= 2^{m+1} h_0 c_m\left(1 - \frac{h_m}{2} - \frac{c_m}{4}\right) = 2^{m+1} c_{m+1} h_0, \quad (3)$$

where the inequality comes from the induction hypothesis, thereby completing the induction step and establishing the result.

As an example, consider a 512 × 512-way system (nine stages) with $p_c = 0.10$ and $p_h = 0.0015$. In this case 18.9% of the cold traffic is rejected, while 40.2% of the hot traffic is rejected. While only 1.48% of the offered load is hot [0.0015/(0.01 + 0.0015)], 3.08% of the overflow traffic is hot. However, the overall throughput of the network is little affected by the hot spot, since the total cold-packet throughput is 99.6% of the total throughput of the same network without any hot traffic (i.e., with $m = 9$, $p_c = 0.10$, and $p_h = 0$). As discussed above and in [30], the throughput at the memory modules close to the hot spot is less than at those modules far away from the hot spot. For example, the throughput at memory module 1, which shares eight stages of output ports with the hot module, is only 85.4% of the throughput at a module that shares no output ports with the hot module.

3. Simulation model of a circuit-switched network

In this section, we describe a simulation model of a more realistic circuit-switched unbuffered Delta network. The network operates synchronously in discrete units of time; we call the unit a network cycle. There are m stages of $k \times k$ switches and a total of $P = k^m$ processors attached to P memories, and there is a unique path connecting each processor to each memory. (Basically, in the absence of conflicts, a message header can advance one stage in one network cycle.) Each processor has a network-interface buffer of length B messages. If this buffer is not full, a new packet is generated on a cycle with probability $p = p_c + p_h$, where p_c is the probability of generating a cold packet whose destination is uniformly distributed over the memories and p_h is the probability of generating a hot packet whose destination is memory module 0. If a processor's buffer is full, no new packets may be accepted by that processor's buffer until a space in the buffer becomes available. Thus, from the point of view of the network, the source is idle, i.e., "turned off," whenever its buffer is full. This corresponds (approximately) to a system in which a processor can have at most B outstanding memory references at any given time. Alternatively, in the context of a system with a hierarchical interconnection network, if a processor generates a new packet during such a buffer-full period, either the new packet or one of the packets waiting in the buffer may be selected for delivery on the Level 2 network.

We assume that as a message is routed through the network on each network cycle, it can either establish a connection to the appropriate output port ($k \times k$ switch) at the next stage in the MIN or, if a conflict occurs, the contention is resolved during the cycle. If two or more packets request the same output port on the same cycle, one packet is (randomly) chosen as the winner and the other colliding packets are declared losers. The winner establishes the connection to the output port on that cycle. On the next cycle, the losers begin to release (free up) the output ports they have acquired so far. (We assume that D stages of ports can be dropped per cycle.) Similarly, if a packet requests a port that is already held by another packet, the requesting packet is declared a loser and, on the next cycle, it begins releasing the output ports it has acquired so far. If a packet successfully arrives at a memory, the packet begins releasing the ports along its path L cycles later (the parameter L can be thought of as a way of representing message length). A successfully transmitted packet is removed from the network-interface buffer at the end of the cycle on which all of the ports have been released.

Unsuccessfully transmitted packets are retried according to a retrial protocol. We consider two retrial protocols.

Table 1 Effect of retrial protocol on cold traffic performance [512 processors, 8 × 8 switches, infinite retrials ($T = \infty$), $p_c = 0.03$, $p_h = 0.001$].

Retrial protocol	*Protocol parameters*	*Throughput* (packets/cycle)	*Mean response time* (cycles)	*Mean number of trials*	$F_c(1)$
Geometric	$p_r = 0.10$	8.80	9.89	1.42	0.800
Geometric	$p_r = 0.05$	8.76	11.52	1.30	0.810
Geometric	$p_r = 0.01$	7.53	22.90	1.17	0.857
Backoff	$p_r = 0.10$, $b_r = 0.5$, $m_r = 0.005$	7.46	8.95	1.17	0.858
Backoff	$p_r = 0.10$, $b_r = 0.5$, $m_r = 0.01$	8.37	9.51	1.22	0.835
Backoff	$p_r = 0.10$, $b_r = 0.25$, $m_r = 0.01$	8.28	10.50	1.22	0.837

The first protocol is a geometric retrials protocol. A packet is resubmitted to the network after waiting a geometrically distributed amount of time (with parameter p_r). After a maximum number of trials T (T, which includes the first trial, could be infinite), the packet is rejected. For example, if $T = 2$, an unsuccessful packet is retried once, and if it is still unable to establish a connection with the memory, it is rejected. In the hierarchical network context, such a packet would then be delivered on the Level 2 network. We are interested in properties of the overflow traffic consisting of rejected packets. In this study, we do not include in the overflow traffic packets that arrive during a buffer-full period (and hence cannot enter the network), since we are interested in the characteristics of the rejected traffic. The second protocol is a geometric-backoff protocol. Let $p_r(j)$ be the probability that a packet is resubmitted on the next cycle, given that it has already been unsuccessfully tried a total of j times. Then for $j = 1$ we set $p_r(1) = p_r$, and for $j > 1$, $p_r(j) = \max\{b_r \times p_r(j-1), m_r\}$, where b_r is the backoff factor ($b_r < 1$) and m_r is the minimum retrial probability. By increasing the waiting time between trials, this protocol attempts to adapt to the level of contention. Because the input-buffer length B is finite, our model is not subject to the instabilities associated with retrial protocols in certain open queueing systems (see [34]).

Since our primary interests in this paper are related to the effects of the hot-spot and retrial protocol, we generally fix some of the above-mentioned parameters at the following values: $k = 8$, $m = 3$, $B = 1$, $L = 1$, and $D = 3$. With these parameters, the system has $P = 512$ processors. The assumption that $B = 1$ corresponds to an assumption that the system behaves like a closed queueing network with population P (see for example [35]). With $L = 1$ and $D = 3$ we are (perhaps optimistically) assuming that the full path is released on the cycle after a packet arrives at a memory.

We also fix probability of generating a cold packet at $p_c = 0.03$. This is a representative choice for the following reasons. Since the switches are very simple, if the network and processors use the same technology, the network cycle time can be expected to be faster than the processor cycle time (in addition, it may not be unreasonable to design the network in a faster technology than the processors; see [6]). If the network is from three to six times faster than the processors, this choice of p_c spans the global memory-access rate of the three parallel scientific applications described in [36] (see specifically Figure 12 of [36]). Also, the value of p_c may be small if the network actually consists of multiple independent networks over which the traffic is distributed (in which case our model is an approximation of one of the independent networks).

In the simulation runs described below, we investigate the performance as the hot-spot probability p_h and the retrial-protocol parameters (p_r, b_r, m_r, and T) are varied.

After examining the output of pilot studies, we select the run lengths for the simulations as follows. The network is simulated for 5000 cycles if $p_h = 0$ and for 20 000 cycles if $p_h > 0$. The statistics from the first 10% of each run are discarded in order to reduce the effects of initialization bias. We replicate each experiment five times if the maximum number of trials $T < \infty$ and 25 times if $T = \infty$, since the relevant estimates from the simulation typically have a lower variance if there is a limit on the number of trials. The results of the replications are then averaged, and standard deviations are calculated in the usual manner. The ratio of a standard deviation to its corresponding point estimate (a measure of the estimate's relative accuracy) is typically found to be less than 0.01, indicating very stable estimates. We use the well-tested random-number generator described in [37].

We collect a variety of statistics for each destination and for each packet type, i.e., hot or cold. These statistics

Table 2 Effect of retrial protocol on hot traffic performance [512 processors, 8 × 8 switches, infinite retrials ($T = \infty$), $p_c = 0.03$, $p_h = 0.001$].

Retrial protocol	*Protocol parameters*	*Throughput* (packets/cycle)	*Mean response time* (cycles)	*Mean number of trials*	$F_h(1)$
Geometric	$p_r = 0.10$	0.29	475	41.11	0.027
Geometric	$p_r = 0.05$	0.29	433	20.53	0.050
Geometric	$p_r = 0.01$	0.25	380	4.68	0.213
Backoff	$p_r = 0.10$, $b_r = 0.5$, $m_r = 0.005$	0.27	630	6.41	0.154
Backoff	$p_r = 0.10$, $b_r = 0.5$, $m_r = 0.01$	0.28	568	8.42	0.119
Backoff	$p_r = 0.10$, $b_r = 0.25$, $m_r = 0.01$	0.27	560	7.65	0.130

include the throughput, the mean number of trials, the mean response time, and the mean stage at which a packet is rejected. We also collect statistics related to the overflow traffic and to the distribution of the number of trials required until a packet is successfully delivered.

Let $f_c(j)$ be the (estimated) probability that a cold packet is successfully delivered on the jth trial, and let $F_c(j) = f_c(1) + \cdots + f_c(j)$ be the (cumulative) probability that a cold packet is successfully delivered in j or fewer trials. Note that $F_c(T)$ is the probability that a packet is eventually delivered successfully over the network. We define the conditional success probability $h_c(j) = f_c(j)/[1 - F_c(j-1)]$; $h_c(j)$ is the discrete analog of the hazard rate (see [33]) and represents the probability that a packet is successfully delivered on the jth trial, given that it is tried at least j times. The trials distribution for hot packets is characterized by $f_h(j)$, $F_h(j)$, and $h_h(j)$, which are defined similarly.

4. Experimental results

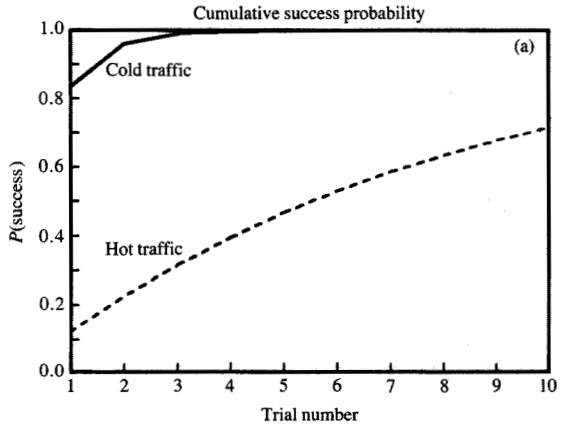
Tables 1 and **2** show the effect of retrial protocol on a variety of performance measures for cold and hot traffic, respectively, with $T = \infty$. In these tables, the effect of varying p_r, the parameter of the geometric retrials protocol, is shown and compared with various geometric-backoff protocols. For cold packets, the probability of successful delivery on the first trial [$F_c(1)$] is high (≥ 0.8) with any of the protocols.

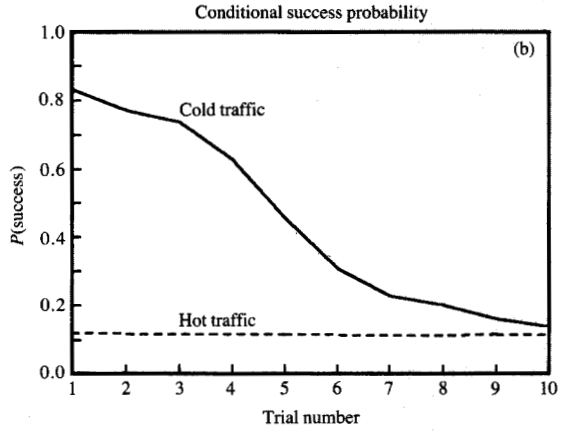
Consider the geometric-retrials protocol. The cold-packet throughput (in packets per cycle) decreases as p_r decreases, and the mean response time increases, since the expected time that an unsuccessful packet spends waiting until resubmission increases as p_r decreases. For hot packets, on the other hand, for large p_r, an extremely large number of trials is required, and the probability of successfully delivering a packet on the first trial is extremely small. Over the range of values shown, the mean hot-packet response time decreases as p_r decreases, since the reduced contention more than offsets the increased waiting time. But note that the mean hot-packet response time is in the hundreds of cycles.

The proper selection of parameters for the backoff protocol can keep the cold-packet mean response time low (in fact, lower than the best geometric-retrials protocol tried). The mean hot-packet response time is increased only by about 10% when m_r, the minimum retrial probability, is reduced from 0.01 to 0.005. There is also little difference in hot-packet response time when the backoff rate b_r is reduced from 0.5 to 0.25. Although the mean number of trials remains small, the mean hot-traffic response time is still in the hundreds of cycles.

To examine certain performance issues in more detail, we choose the geometric-backoff protocol and fix its parameters at $p_r = 0.1$, $b_r = 0.5$, and $m_r = 0.01$. Although the numerical results depend upon these parameters, the general shapes of the performance curves plotted and the conclusions we draw from them are quite insensitive to the retrial protocol and its parameters.

Figure 2(a) is a plot of the cumulative trials distributions [$F_c(j)$ and $F_h(j)$], and **Figure 2(b)** shows their associated hazard rates [$h_c(j)$ and $h_h(j)$] for the case of infinite retrials. While most of the cold packets are successfully transmitted in just a few trials, $F_h(j)$ increases slowly. Only about 70% of the hot packets are successfully delivered in ten or fewer trials. In addition, while the hot-packet conditional success probability $h_h(j)$ remains nearly constant, $h_c(j)$, for cold packets, is a decreasing function of j; i.e., with every trial, it becomes less likely that a cold packet will be successfully transmitted. This phenomenon can be explained as follows. As described in Section 2, the probability that a packet is unsuccessful is larger when its destination is closer to the hot module. Thus, rejected





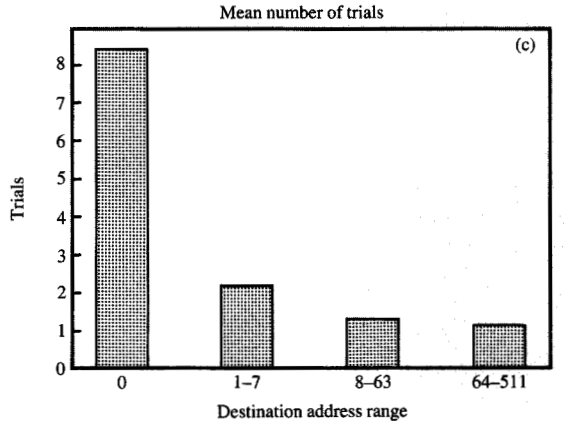


Figure 2

Network performance for infinite trials: Circuit-switched Delta network with geometric-backoff retrials protocol (512 processors, 8×8 switches, $B = 1$, $T = \infty$, $p_c = 0.03$, $p_h = 0.001$, $p_r = 0.1$, $b_r = 0.5$, $m_r = 0.01$).

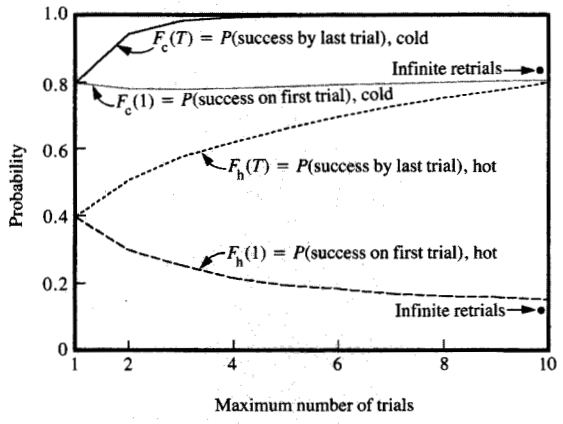


Figure 3

Probability of successful transmission vs. maximum number of trials: Circuit-switched Delta network with geometric-backoff retrials protocol (512 processors, 8×8 switches, $B = 1$, $p_c = 0.03$, $p_h = 0.001$, $p_r = 0.1$, $b_r = 0.5$, $m_r = 0.01$).

packets are more likely to be destined for the hot module or one of its neighbors. This, in turn, means that a rejected packet is less likely to be successful on the second trial than is a typical cold packet on its first trial. With every successive trial, the rejected packets become more likely to be destined for the hot module and thus become less likely to be successfully transmitted. This explains the DFR nature of the cold-packet trials distribution. With a larger network-interface buffer size B and with limited retrials, we also observe that $h_h(j)$ decreases somewhat with j.

Figure 2(c) shows that the mean number of trials is a decreasing function of the destination's distance from the hot module. In this example, besides the hot module itself (module 0), there are three classes of memory modules corresponding to the number of output ports shared with traffic destined for the hot module. These classes are modules 1–7, modules 8–63, and modules 64–511, which share two, one, and no output ports, respectively, with traffic for the hot module. The mean number of trials ranges from 8.42 for the hot-module traffic to 1.18 for the packets destined for modules 64–511. We also note that when conflicts do occur, hot packets tend to conflict closer to the memories than do cold packets; the mean stage at which a hot packet conflicts is 2.31 as opposed to 1.87 for cold packets.

Figure 3 shows the effect that the maximum number of trials T has on performance. The probability of successfully transmitting a cold packet on the first trial, $F_c(1)$, is quite insensitive to T. On the other hand, for hot

packets, $F_h(1)$ is a decreasing function of T, since with increasing T there are more hot packets competing for the paths to the hot memory. The probability of eventually transmitting a cold packet successfully, $F_c(T)$, increases rapidly to 1 as T increases, while, for hot packets, $F_h(T)$ increases slowly with T. For example, 94% of the cold traffic is successfully transmitted when $T = 2$, but even with $T = 10$, only 80% of the hot traffic is successfully transmitted.

Figures 4(a) and **4(b)** show the total throughput of successfully delivered cold and hot packets, respectively, as functions of T. While the hot-packet throughput steadily increases, the cold-packet throughput increases, and then decreases as T increases. The increase in cold-packet throughput when T changes from 1 to 2 is due to the increased probability of successful transmission. The decreasing cold-packet throughput for $T > 2$ is due essentially to a closed queueing-network effect. As T increases, the amount of time the input buffer is full (typically with a hot packet) increases. Since no new cold packets can be introduced during such a period, the cold-traffic throughput decreases with T.

Figure 4(c) shows that as T increases, the fraction of overflow traffic that is hot (i.e., destined for the hot memory module 0) increases very rapidly. This is the analog of Inequality (1), which was established in Section 2 for an unbuffered packet-switched network. This effect is explained by the same reasoning (given above), which showed that the cold-packet trials distribution is DFR-like. As T increases, essentially only traffic destined for the hot module requires more than T trials; therefore, the overflow is predominantly hot. For example, from Figure 4(c), while only about 3% of the offered load is hot, 24% of the overflow traffic is hot for $T = 2$ and 51% of it is hot for $T = 3$.

Figure 5 shows the sensitivity of the cold-packet throughput with respect to T and B, the size of the network-interface buffer. For a given value of T, as B increases, the throughput increases (to a finite maximum value), since the probability that the buffer is full decreases, so the buffer can accept more packets. As does Figure 4(a) (with $B = 1$), Figure 5 shows that, for a fixed B, the cold-packet throughput increases, and then decreases with increasing T. Initially, as T increases, the throughput increases, since more cold packets are successfully transmitted. However, for large T, the throughput decreases, since a packet destined for the hot module (or one of its neighbors) spends a long time at the head of the buffer's queue. This increases the probability that the buffer is full, thereby reducing the expected number of packets accepted into the buffer.

In the context of hierarchical networks, the above results suggest that a low limit should be placed on the maximum number of trials, since almost all of the cold

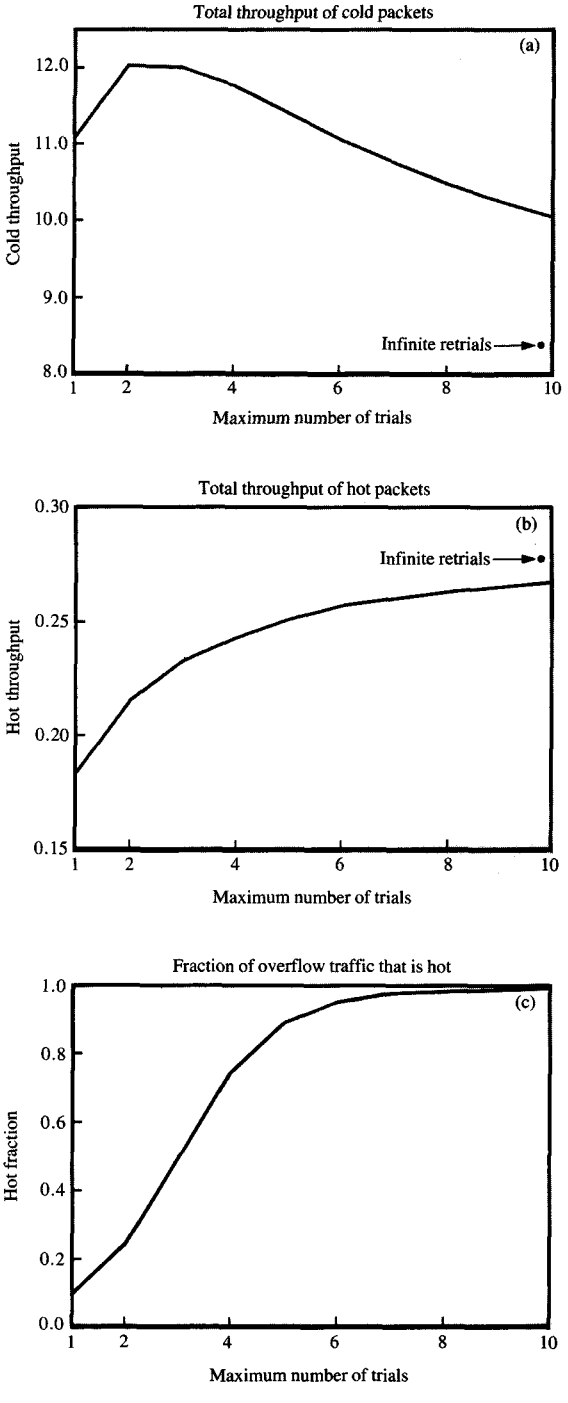


Figure 4

Network performance vs. maximum number of trials: Circuit-switched Delta network with geometric-backoff retrials protocol (512 processors, 8 × 8 switches, $B = 1$, $p_c = 0.03$, $p_h = 0.001$, $p_r = 0.1$, $b_r = 0.5$, $m_r = 0.01$).

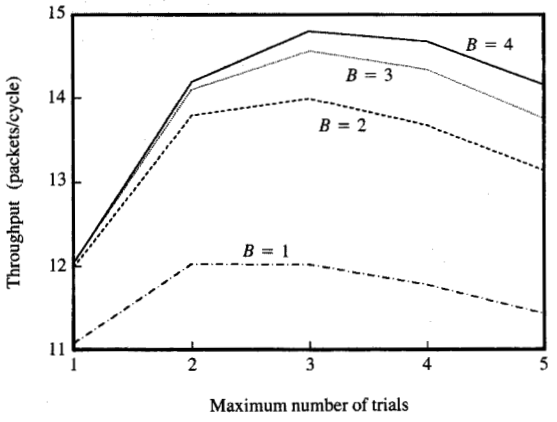


Figure 5

Total throughput of cold packets vs. maximum number of trials and buffer size: Circuit-switched Delta network with geometric-backoff retrials protocol (512 processors, 8 × 8 switches, $p_c = 0.03$, $p_h = 0.001$, $p_r = 0.1$, $b_r = 0.5$, $m_r = 0.01$).

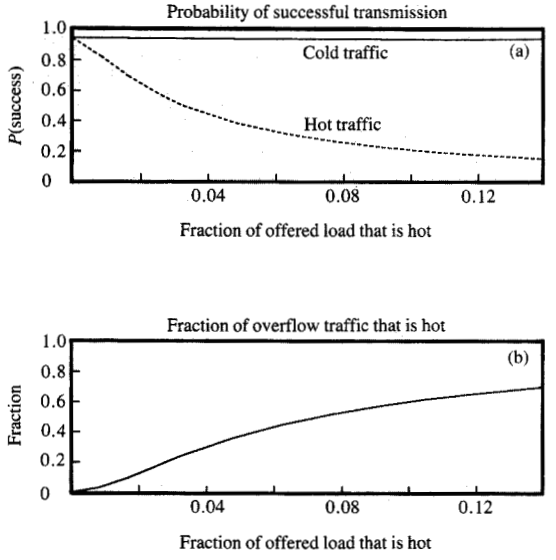


Figure 6

Performance vs. fraction hot load, for maximum of two trials: Circuit-switched Delta network with retrials (512 processors, 8 × 8 switches, $B = 1$, $T = 2$, $p_c = 0.03$, $p_r = 0.1$).

traffic can then be successfully delivered over the fast Level 1 network. In addition, there is little benefit in employing a large number of retrials, since the probability of successfully delivering a hot packet grows slowly and the (conditional) probability of delivering any of the residual cold packets actually decreases. Therefore, we fix the maximum number of trials at $T = 2$ and study the sensitivity of performance as the fraction of hot-spot traffic is varied (specifically, we keep $p_c = 0.03$ and vary p_h). **Figure 6(a)** shows that the probability of successfully delivering a cold packet is little affected by the value of p_h; however, the probability of successfully delivering a hot packet steadily decreases as p_h increases. **Figure 6(b)** shows how the fraction of overflow traffic that is hot increases as the fraction of the offered load that is hot increases.

Figures 3 and 6 illustrate that with $B = 1$, once a cold packet reaches the head of the network-interface-buffer queue, its performance is little affected by the hot-spot traffic. **Figure 7** shows how the hot spot can adversely affect the overall cold-traffic performance by increasing the amount of time that packets spend waiting to get to the head of the network-interface-buffer queue. For $B = 4$ (and all other parameters as before), Figure 7 displays the mean time that a packet spends waiting to get to the head of the queue as functions of T, for two cases:

1. The offered traffic consists of both cold and hot traffic ($p_c = 0.03$ and $p_h = 0.001$).
2. The offered traffic consists of only cold traffic ($p_c = 0.03$ and $p_h = 0$).

The studies reported here have, by necessity, kept certain parameters fixed. However, the effects observed and discussed here hold, in general, for other parameter settings. For example, if 2 × 2 switches are used rather than 8 × 8 switches, the fraction of overflow traffic that is hot is reduced, since the overall level of cold-packet conflicts increases (assuming all other parameters are fixed). For example, with $T = 2$, the fraction of overflow traffic that is hot is reduced from 24% to 8% when 2 × 2 switches are used rather than 8 × 8 switches (and all other parameters are as in Figure 4). In this case, the probability of successfully transmitting a cold packet is reduced from 0.94 to 0.77.

5. Summary and discussion

This paper has examined the performance of a class of unbuffered multistage interconnection networks (specifically Delta networks) under a nonuniform (hot-spot) traffic pattern. While the study of such networks is of broad interest, our specific motivation for these studies is to understand hierarchical interconnection networks, in which there are hierarchies of networks that provide

different levels of performance, e.g., latency and probability of successful packet delivery. In this context, we focused on performance issues related to the (fast) Level 1 network, assuming it is an unbuffered Delta network.

In particular, we studied the overflow traffic of rejected packets, which forms the offered load to the Level 2 network in the hierarchy. For a simple, analytically tractable model of a packet-switched network, we proved that the overflow traffic is fractionally hotter than the offered load. Simulation studies extended this result to a circuit-switched network with limited retrials. As the maximum number of trials increases, the Delta network filters out more and more of the cold traffic. The result of this filtering is that the overflow traffic, while having a relatively low overall throughput, is intensely hot. For example, in a 512-way system with a maximum of only three trials, the network filters and concentrates an offered load that is 3% hot into an overflow traffic that is just over 50% hot. Thus, in a hierarchical network, the Level 2 network must be capable of handling traffic that has a low to moderate input rate but also has an extremely nonuniform destination distribution.

The distribution of the number of trials required before a packet is successfully delivered was also studied in detail. Simulation studies showed that for cold packets, this distribution has a decreasing hazard rate. With each successive retrial, it thus becomes less likely that a cold packet will be successfully delivered on that trial. This is again related to the filtering effect, since with each successive retrial, it becomes more likely that a cold packet is actually destined for the hot module or one of its neighbors. This observation also has implications for analytic modeling approaches to such networks, since a typical modeling assumption is that rejected packets can be treated as being independently resubmitted. Our studies show that the distribution of destinations is not independent of the number of retrials. In fact, it changes dramatically, by becoming more and more concentrated on the hot module with each retrial.

Our studies also confirm other studies and measurements that show that the hot-spot effect in unbuffered networks is quite different from the effect in buffered networks (see [7, 8, 18, 19, 30, 31]. Buffered networks suffer from the "tree-saturation" effect, in which the hot traffic quickly causes buffers to become full, thereby reducing the throughput of the entire network. In an unbuffered network, once a packet reaches the head of the network interface buffer, the hot-spot effect is much more localized since the probability of successfully transmitting a packet is reduced significantly for only the hot module and its neighbors. On the other hand, even with only a slight nonuniformity in the traffic pattern, the probability of successfully transmitting a hot packet on a given trial is dramatically reduced. While the probability of eventually transmitting the hot packet successfully increases with the maximum number of trials, the hot-packet network response time also increases. In addition, the time spent waiting in the interface buffer increases, and the cold-packet throughput decreases (eventually). This corresponds to the tree-saturation effect, but it occurs outside the interconnection network itself.

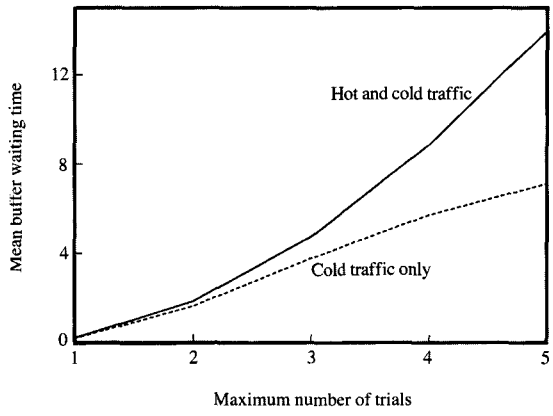


Figure 7

Mean buffer waiting time vs. maximum number of trials, for buffer size 4: Circuit-switched Delta network with geometric-backoff retrials protocol (512 processors, 8×8 switches, $B = 4$, $p_c = 0.03$, $p_h = 0.001$ and 0, $p_r = 0.1$, $b_r = 0.5$, $m_r = 0.01$).

It may be possible to reduce the hot-spot effect by providing alternative paths to the memories. There are many ways to do this. However, we conjecture that such efforts will have only marginal benefit unless multiple ports to memory are provided (or, for example, if multiple independent paths are connected to a multiplexer with a high degree of buffering, which is, in turn, connected to the memory—see [29]). However, such multiple-path networks are much more complex from a hardware standpoint. In addition, at this point, it is unclear how the performance of such a multiple-path, unbuffered network compares to that of a hierarchical network in which the Level 2 network provides guaranteed delivery.

Acknowledgment

We wish to thank an anonymous referee for many helpful comments.

References

1. P. A. Franaszek, "Path Hierarchies in Interconnection Networks," *IBM J. Res. Develop.* **31,** No. 1, 120–131 (January 1987).

2. P. A. Franaszek and C. J. Georgiou, "Multipath Hierarchies in Interconnection Networks," *Supercomputing, 1st International Conference Proceedings*, Athens, Greece, June 1987; *Lecture Notes in Computer Science*, Number 297, Springer-Verlag, New York, 1987, pp. 112–123.
3. A. N. Tantawi, "Performance and Modeling of a Hierarchically Interconnected Multiprocessor," *Research Report RC-13335*, IBM Thomas J. Watson Research Laboratory, Yorktown Heights, NY, 1988.
4. L. N. Bhuyan, "Guest Editor's Introduction: Interconnection Networks for Parallel and Distributed Processing," *Computer* **20,** No. 6, 9–13 (June 1987).
5. T. Y. Feng, "A Survey of Interconnection Networks," *Computer* **14,** No. 12, 12–27 (December 1981).
6. G. F. Pfister, W. C. Brantley, D. A. George, S. L. Harvey, W. J. Kleinfelder, K. P. McAuliffe, E. A. Melton, V. A. Norton, and J. Weiss, "The IBM Research Parallel Processor Prototype (RP3): Introduction and Architecture," *Proceedings of the 1985 International Conference on Parallel Processing*, IEEE Computer Society Press, Piscataway, NJ, 1985, pp. 764–771.
7. R. Rettberg and R. Thomas, "Contention Is No Obstacle to Shared-Memory Multiprocessing," *Commun. ACM* **29,** No. 12, 1202–1212 (December 1986).
8. R. Thomas, "Behavior of the Butterfly™ Parallel Processor in the Presence of Memory Hot Spots," *Proceedings of the 1986 International Conference on Parallel Processing*, IEEE Computer Society Press, Piscataway, NJ, 1986, pp. 46–50.
9. D. M. Dias and J. R. Jump, "Packet Switching Interconnection Networks for Modular Systems," *Computer* **14,** No. 12, 43–53 (December 1981).
10. J. R. Jump and D. M. Dias, "Analysis and Simulation of Buffered Delta Networks," *IEEE Trans. Computers* **C-29,** No. 9, 791–801 (September 1981).
11. C. P. Kruskal and M. Snir, "The Performance of Multistage Interconnection Networks," *IEEE Trans. Computers* **C-32,** No. 12, 1091–1098 (December 1983).
12. C. P. Kruskal, M. Snir, and A. Weiss, "The Distribution of Waiting Times in Clocked Multistage Interconnection Networks," *IEEE Trans. Computers* **37,** No. 11, 1337–1352 (November 1988).
13. H. Yoon, K. Y. Lee, and M. T. Liu, "Performance Analysis of Multibuffered Packet-Switching Networks in Multiprocessor Systems," *IEEE Trans. Computers* **39,** No. 3, 319–327 (March 1990).
14. M. Kumar and J. R. Jump, "Performance Enhancement in Buffered Delta Networks Using Crossbar Switches and Multiple Links," *J. Parallel & Distributed Computing* **1,** 81–103 (August 1984).
15. D. Mitra and R. A. Cieslak, "Randomized Parallel Communications on an Extension of the Omega Network," *J. ACM* **34,** No. 4, 802–824 (October 1987).
16. G. F. Pfister and V. A. Norton, " 'Hot Spot' Contention and Combining in Multistage Interconnection Networks," *IEEE Trans. Computers* **C-34,** No. 10, 943–948 (October 1985).
17. U. Garg and Y. P. Huang, "Decomposing Banyan Networks for Performance Analysis," *IEEE Trans. Computers* **C-37,** No. 3, 371–376 (March 1988).
18. P. G. Harrison and N. Patel, "The Representation of Switching Networks in Queueing Models of Parallel Systems," *Performance '87*, North-Holland Publishing Co., Amsterdam, 1988, pp. 497–512.
19. P. G. Harrison and N. Patel, "On 'Hot-Spot' Contention in Interconnection Networks," *Proceedings of the 1988 ACM SIGMETRICS Conference on Measurement and Modeling of Computer Systems*, ACM Press, New York, 1988, pp. 114–123.
20. M. Kumar and G. F. Pfister, "The Onset of Hot-Spot Contention," *Proceedings of the 1986 International Conference on Parallel Processing*, IEEE Computer Society Press, Piscataway, NJ, 1986, pp. 28–34.
21. A. Gottlieb and J. T. Schwartz, "Networks and Algorithms for Very Large Scale Parallel Computations," *Computer* **15,** No. 1, 27–36 (January 1982).
22. J. S. Cameron, "A Study of Message Combining Networks for Large Scale Parallel Processor Systems with Memory Hotspots," *Technical Report 88-05-04*, Department of Computer Science, University of Washington, Seattle, 1988.
23. G. Lee, C. P. Kruskal, and D. J. Kuck, "The Effectiveness of Combining in Shared Memory Parallel Computers in the Presence of 'Hot Spots,' " *Proceedings of the 1986 International Conference on Parallel Processing*, IEEE Computer Society Press, Piscataway, NJ, 1986, pp. 35–41.
24. P. C. Yew, N. F. Tzeng, and D. H. Lawrie, "Distributing Hot-Spot Addressing in Large-Scale Multiprocessors," *IEEE Trans. Computers* **C-36,** No. 4, 388–395 (April 1987).
25. J. A. Patel, "Performance of Processor-Memory Interconnections for Multiprocessors," *IEEE Trans. Computers* **C-30,** No. 10, 771–780 (October 1981).
26. L. N. Bhuyan, "An Analysis of Processor-Memory Interconnection Networks," *IEEE Trans. Computers* **C-34,** No. 3, 279–283 (March 1985).
27. L. N. Bhuyan and D. P. Agrawal, "Design and Performance of Generalized Interconnection Networks," *IEEE Trans. Computers* **C-32,** No. 12, 1081–1090 (December 1983).
28. S. C. Kothari, A. Jhunjhunwala, and A. Mukherjee, "Performance Analysis of Multipath Multistage Interconnection Networks," *Proceedings of the 1988 ACM SIGMETRICS Conference on Measurement and Modeling of Computer Systems*, ACM Press, New York, 1988, pp. 124–132.
29. M. Kumar and J. R. Jump, "Performance of Unbuffered Shuffle-Exchange Networks," *IEEE Trans. Computers* **C-35,** No. 6, 573–578 (June 1986).
30. Y. S. Liu, "Delta Network Performance and 'Hot Spot' Traffic," *Ultracomputer Project Technical Report 132*, Courant Institute, New York University, New York, 1987.
31. A. Pombortsis and C. Halatsis, "Performance of Crossbar Interconnection Networks in Presence of 'Hot Spots,' " *Electron. Lett.* **24,** No. 3, 182–184 (February 1988).
32. M. Lee and C. L. Wu, "Performance Analysis of Circuit Switching Baseline Interconnection Networks," *Proceedings of the 11th Annual International Symposium on Computer Architecture*, IEEE Computer Society Press, Piscataway, NJ, 1984, pp. 82–90.
33. R. E. Barlow and F. Proschan, *Statistical Theory of Reliability and Life Testing Probability Models*, Holt, Rinehart and Winston, Inc., New York, 1975.
34. J. R. Aldous, "Ultimate Instability of Exponential Back-off Protocol for Acknowledgement-Based Transmission Control of Random Access Communication Channels," *IEEE Trans. Info. Theory* **IT-33,** No. 23, 219–223 (March 1987).
35. S. S. Lavenberg, Ed., *Computer Performance Modeling Handbook*, Academic Press, Inc., New York, 1983.
36. F. Darema-Rogers, G. F. Pfister, and K. So, "Memory Access Patterns of Parallel Scientific Programs," *Proceedings of the 1987 ACM SIGMETRICS Conference on Measurement and Modeling of Computer Systems*, ACM Press, New York, 1987, pp. 46–58.
37. P. L'Ecuyer, "Efficient and Portable Combined Random Number Generators," *Commun. ACM* **31,** No. 6, 742–749, 774 (June 1988).

Received April 16, 1989; accepted for publication March 26, 1990

Philip Heidelberger *IBM Research Division, Thomas J. Watson Research Center, P.O. Box 704, Yorktown Heights, New York 10598.* Dr. Heidelberger received a B.A. in mathematics from Oberlin College in 1974 and a Ph.D. in operations research from Stanford University in 1978. He has been a Research Staff Member at the IBM Thomas J. Watson Research Center in Yorktown Heights, New York, since 1978. His research interests include modeling and analysis of computer performance and statistical analysis of simulation output. Dr. Heidelberger is currently an Area Editor for the ACM's *Transactions on Modeling and Computer Simulation*; he was an Associate Editor of *Operations Research* and was the Program Chairman of the 1989 Winter Simulation Conference. He is a member of the Operations Research Society of America, the ACM, and the IEEE.

Peter A. Franaszek *IBM Research Division, Thomas J. Watson Research Center, P.O. Box 704, Yorktown Heights, New York 10598.* Dr. Franaszek is manager of Systems Theory and Analysis in the Computer Sciences Department at the Thomas J. Watson Research Center. He received an Sc.B. degree from Brown University in 1962, and M.A. and Ph.D. degrees from Princeton University in 1964 and 1965, respectively. Dr. Franaszek's interests include analytical and design issues in computer system organization, algorithms, and communication networks and coding. He has received IBM Outstanding Innovation Awards for his work in the areas of algorithms, interconnection networks, concurrency control theory, and constrained coding. Dr. Franaszek was also the recipient of two IBM Corporate Awards for his work in the latter area. He was named the recipient of the 1989 Emanuel R. Piori Award of the Institute of Electrical and Electronics Engineers for his contribution to the theory and practice of digital recording codes. During the academic year 1973–1974, he was on sabbatical leave from the Thomas J. Watson Research Center to Stanford University as a Consulting Associate Professor of Computer Science and Electrical Engineering. Prior to joining IBM in 1968, he was a member of the technical staff at Bell Telephone Laboratories. Dr. Franaszek is a member of Tau Beta Pi and Sigma Xi, and a Fellow of the IEEE.

The "Recent publications by IBM authors" section that normally appears here has been omitted due to the size of this issue. It will return in the next issue.