The Representation of Multistage Interconnection Networks in Queuing Models of Parallel Systems

PETER G. HARRISON AND NARESH M. PATEL

Imperial College, London, England

Abstract. A major component of a parallel machine is its interconnection network (IN), which provides concurrent communication between the processing elements. It is common to use a multistage interconnection network (MIN) that is constructed using crossbar switches and introduces contention not only for destination addresses but also for internal links. Both types of contention are increased when nonlocal communication across a MIN becomes concentrated on a certain destination address, the *hot-spot*. This paper considers analytical models of asynchronous, circuit-switched INs in which partial paths are held during path building, beginning with a single crossbar and extending recursively to MINs. Since a path must be held between source and destination processors before data can be transmitted, switching networks are passive resources and queuing networks that include them do not therefore have product-form solutions. Using decomposition techniques, the flow-equivalent server (FES) that represents a bank of devices transmitting through a switching network is determined, under mild approximating assumptions. In the case of a full crossbar, the FES can be solved directly and the result can be applied recursively to model the MIN. Two cases are considered: one in which there is uniform routing and the other where there is a hot-spot at one of the output pins. Validation with respect to simulation for MINs with up to six stages (64-way switching) indicates a high degree of accuracy in the models.

Categories and Subject Descriptors: C.1.2 [**Processor Architectures**]: Multiple Data Stream Architectures (Multiprocessors)—*interconnection architectures*; C.4 [**Performance of Systems**]: *modeling techniques, design studies, performance attributes*; I.6.3 [**Simulation and Modeling**]: Applications; I.6.3 [**Simulation and Modeling**]: Model Validation and Analysis

General Terms: Design, Parallelism, Performance

Additional Key Words and Phrases: Closed queuing network, crossbar switch, delta network, flow-equivalent server, Markov process, multistage interconnection network, performance evaluation

1. *Introduction*

The complexity of large-scale parallel computer architectures is such that their effective design requires guidance from some form of predictive modeling prior to construction, and analytical models based on queuing network analysis provide a powerful tool for this purpose. In many parallel machines, for example ALICE [8, 12], interconnection networks (INs) are required by direct memory access (DMA) processors to provide fast communication between the many processing and memory elements. It is normally necessary for any element of one type to be

Parts of this paper appeared in *Proceedings of the Performance '87 Conference* and in *Proceedings of the 1988 ACM Sigmetrics Conference.*

Authors' address: Department of Computing, Imperial College, London SW7 2BZ, England.

able to communicate with any of the other type, for example, each processor must have access to any memory. Typically, the memory elements are not uniformly utilized by programs, for example, when a frequently accessed data structure is stored entirely in one part of the distributed memory, and the network traffic becomes concentrated on certain destination addresses. These are often referred to as *hot-spots*. In principle, the interconnection can be provided by a crossbar switch in which the contention for destination addresses is relatively easy to model as a degradation of the transmission service rates. However, in practice, large crossbars cannot be fabricated cost-effectively and a multistage interconnection network (MIN) is normally used, comprising a number of stages of parallel banks of small, say 2-way, switches. This provides the required connectivity, but the sharing of internal links by different paths across the network introduces extra contention. Furthermore, when there is a hot-spot at one of the destination addresses, all paths to it (hot paths) become busier than in the uniformly utilized case and other paths sharing links with the hot paths are also adversely affected.

There are two transmission protocols that require entirely different model types—circuit switching and packet switching. In the case of circuit switching (or unbuffered packet switching), the network is modeled as a *passive* resource in a queuing network; in order to operate a data-transmission server must hold a path comprising internal links and crossbars at each stage through to the destination addressed by the task at the front of its queue. For packet switching, such paths need not be held throughout a message's transmission, and buffered internal crossbars can be modeled as conventional servers since their transmission times are nonnegligible because of the intelligence required to provide the buffering. However, similar problems arise if the buffers become exhausted and cannot be reasonably assumed infinite. We consider the alternative protocol—circuit switching—in which the interconnection network is a passive resource, providing no direct service in the sense of advancing tasks holding network links towards their completion. (A passive resource is one that must be held by a task in order to permit some associated *active* resource to provide service to that task, executing units of applied workload at its particular service rate; the most common example is computer memory.)

Assuming infinite buffers, if the departure processes from the output pins of each stage can be determined in terms of the stage's arrival processes, the arrival processes to the next stage can be derived by simply permuting the outputs according to the interconnection topology. Shuffle-exchange is therefore a particularly convenient representation of this topology, since the permutation is then the same between every pair of stages [28]. Certain approximations regarding the internal arrival processes at successive stages simplify the analysis, providing an efficient algorithm, and such an analysis is given in [18] and [22]. An analysis of a packet-switched crossbar in discrete time is given in [26] and an approximate analysis of a buffered MIN with nonuniform routing is given in [9]. Nonuniform traffic models in circuit-switched MINs are also analyzed in [3] and [21].

Our approach to this problem relies on the *recursive structure* of the MINs we consider, which is in contrast to analogous research reported by Kelly on telephone networks [16]. In either case, we could in principle consider all routes through the network individually and solve the associated Markov process directly to obtain the Erlang loss formula—assuming, that is, that partial paths are not held but released on a collision, resulting in lost transmissions. Because of the large number of routes, such a direct solution is computationally intractable. Kelly's analysis relies on the network behaving as if the steady-state probabilities that each link is blocked were independent, which property is shown to hold asymptotically as link capacities and network traffic approach infinity jointly so that the traffic offered per link remains constant. In fact, the approximation is very accurate for telephone networks because of the large capacity links and large number of nodes. Our analysis differs from this in that links have *small* capacities (namely, one) and also that partial paths are held during path building. Moreover, partly as a result of the former difference, there is a strong dependence between individual links that do not behave remotely as if they were independent, and it is our recursive analysis that captures this dependence in a simple way. In this sense, our analysis addresses the opposite end of the spectrum of switching networks and their control protocols.

In the next section, we describe some of the switching networks commonly used in multiprocessor systems, together with their operational characteristics, and discuss the problems encountered in modeling such networks. Our general approach to modeling a queuing subnetwork that represents a collection of DMA servers together with a circuit-switched MIN is considered in Section 3. First, we apply decomposition techniques to obtain the service rate of a *flow-equivalent server* (FES) by determining the throughput of the short-circuited subnetwork as a function of its population, according to the method of [4]. This throughput is obtained by defining a simple Markov process, under appropriate assumptions, in which some function, μ_n, for throughput is assumed to be available if the number of active *input* pins, n, is known, that is, the number of DMA servers currently wishing to transmit. In Sections 4 and 5, we derive expressions for μ_n in the cases that the interconnection network is respectively a *full crossbar switch* and a *delta network*, which is itself constructed from and analyzed in terms of crossbars. The degree of the degradation introduced by these networks on the DMA servers' throughput is illustrated by numerical predictions in Section 7. In these analyses, the networks are *uniform*, which means that all output destination addresses (pins) are selected with equal probability by any transaction, but in Section 6 we consider the case where the destination address space may have a *hot-spot* [25], that is, one pin that is selected with a higher probability than the rest, which are still all selected with the same (reduced) probability. Our conclusions and suggestions for future work are laid out in Section 8.

2. *Interconnection Networks for Multiprocessors*

Tightly coupled or shared-memory multiprocessor systems require some kind of switch to connect their processing elements (memories included), so allowing them to communicate, and being a unique shared resource in the parallel machine, the switch is a potential source of contention. The simplest type of switch for this purpose is the common bus that connects to every element. However, the bus derives its simplicity at the expense of increased complexity in each component that it connects. For example, bus request and receive pins are needed by each element, together with several others for control and synchronization, and nontrivial logic is necessary to implement the bus's protocol. This increases the overall cost of a large-scale multiprocessor and, moreover, each additional processor that is added to the bus is faced with more contention and so delivers a diminishing increment in performance.

At the other end of the spectrum, the switch might be a full crossbar that provides parallel communication between any number of distinct pairs of processors connected through it. For example, a crossbar may have two types of links—for processors and for memory elements—and a way of dynamically connecting any pair of the two types. This allows parallel access to the shared memory provided that there are no *memory conflicts*, that is, two or more processors attempting to access the same memory element.

An $a \times b$ crossbar can switch any of its a input pins to any of its b output pins on the appropriate clock-cycle, so that the only contention possible is for the destination processors, that is, is internal in the switch for the output pins. We therefore have the configuration shown in Figure 1 when $a = b = 4$.

Each additional processor that is added to the crossbar requires a new parallel link to each of the output pins. Thus, for an N-way crossbar the complexity and cost grow as $O(N^2)$ and the very large crossbars required for large-scale multiprocessors would become hopelessly complex and expensive. Moreover, large crossbars are physically infeasible to construct using contemporary electronic devices.

However, we can interconnect a matrix of small crossbars to form a MIN that provides the same connectivity (any input pin can be dynamically connected to any output pin if there is a free path), but introduces contention for internal links (*path conflicts*) in addition to memory conflicts. The cost of an N-way MIN grows almost linearly, $O(N \log N)$, compared with $O(N^2)$ for the equivalent crossbar. There are several types of MIN, many with similar structure and properties, but here we will consider a subclass of *banyan networks.* A banyan network is a MIN with a unique path from each input pin to each output pin. This criterion implies that the network has a tree-like fan-in and fan-out structure; a formal definition of banyan networks in terms of graph theory may be found in [10]. A banyan network comprising J stages of crossbars in which outputs of one stage connect directly to the inputs of the next stage, is called a *J-level banyan.* There are two main types of J-level banyans: regular banyans and irregular banyans. Irregular banyans connect any N inputs to M outputs through J stages of crossbars where the ith stage employs c_i $n_i \times m_i$ crossbars. Since no crossbar pin is left unconnected, the number of outputs from one stage have to be the same as the number of inputs to the next stage, that is, $c_i m_i = c_{i+1} n_{i+1}$ for $1 \le i < J$. This is satisfied by

$$c_i = \prod_{j=1}^{i-1} m_j \prod_{k=i+1}^{J} n_k \qquad \text{for} \quad 1 \le i \le J.$$

Hence, the number of inputs $N = n_1 \times c_1 = n_1 \times n_2 \times \cdots \times n_J$ and the number of outputs $M = c_J \times m_J = m_1 \times m_2 \times \cdots \times m_J$. Regular banyans are constructed from a single type of basic crossbar switch and examples of specific structures include CC banyans and SW banyans. The CC in CC banyan is an acronym for cylindrical cross-hatch since the network can be laid out as a cross-hatch pattern on a cylindrical surface. In this paper, we consider SW banyans which can be built by recursively extending the crossbar as defined in the next section.

2.1 Delta Networks. A well-known example of a regular banyan network is the delta network defined in [23]. The definition of delta networks includes the additional property that routing in the network is digit-controlled, that is, the choice of which output pin to select at a particular crossbar in the network can be determined by a single digit in the destination address of the packet. A rectangular delta network is constructed from b-way crossbars with the same number of input pins as output pins and is often called a delta-b network. There are a number of different ways of connecting the outputs of one stage of crossbars to the inputs of the next so as to obtain the desired connectivity. The corresponding permutation functions define the network's topology. All these topologies are equivalent in that one can be obtained from another by permuting the switches in each stage of the

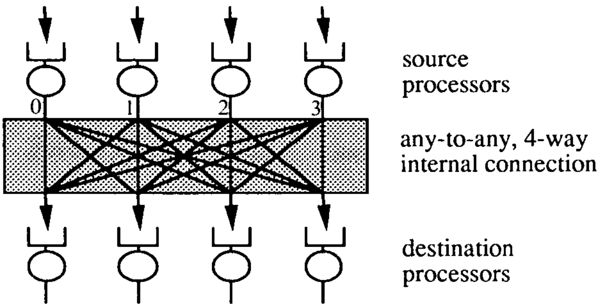

FIG. 1. The 4-way crossbar switch.

network and so all have the same performance characteristics, such as equilibrium throughput. For our recursive analysis we adopt the *partial shuffle* variant of the *cube-topology* [23], which may be defined recursively as follows for networks with 2-way crossbars (the definition can be generalized easily to networks constructed from b-way crossbars with $b > 2$):

(i) A one-stage network, Π_1, is the single 2-way crossbar.

(ii) An s-stage network, Π_s, $(s > 0)$ has 2^s inputs and outputs, that is, 2^{s-1} switches in each stage, and is constructed by connecting a *head stage* of 2^{s-1} switches to the right of 2 *tail networks* of $(s - 1)$ stages according to the partial shuffle topology shown in Figure 2. The ith switch in stage s takes its top input and bottom input from the ith pin of the upper subnetwork and the ith pin of the lower subnetwork, respectively. This property of the topology is particularly useful in obtaining the recurrence formulas in Sections 5 and 6.

2.2 OPERATIONAL CHARACTERISTICS OF SWITCHING NETWORKS. MINs have three main operational characteristics: control, timing, and switching protocol. Typically, MINs have decentralized control with implicit digit-controlled routing as in delta networks. However, timing characteristics depend on the type of parallel machine that uses the MIN. For SIMD machines, the MIN is normally used in synchronous mode whereas in MIMD machines the MIN is used in asynchronous mode. As mentioned in the Introduction, there are two main types of switching protocols—circuit switching and packet switching—as well as hybrid protocols. In circuit switching, a complete path has to be established across the network before data can be transferred from the input buffer to the destination buffer. In the process of setting up this path, there may be a required link already in use which causes the path to be blocked. The partial path already established may then be held until the link becomes free or may be released, whereupon the source processor may have to retry after some time. In a packet-switched MIN, there are buffers at each crossbar switch and packets of data are transferred from one buffer to the next in a single hop fashion. This protocol incurs higher transfer delays than circuit switching but introduces greater potential for parallelism and higher throughput. Hybrid protocols can also be used in buffered MINs in which packets can bypass many buffers by forming circuits across any number of stages when possible.

In the ALICE machine [13], the interconnection is provided by a 64-way delta-4 network. This network operates in asynchronous mode with a circuit-switching protocol in which partial paths are held. Similar protocols have been proposed for a new generation of INs constructed from optical crossbar switches [15], which can be fabricated with larger dimensions than electronic crossbars and so deliver higher

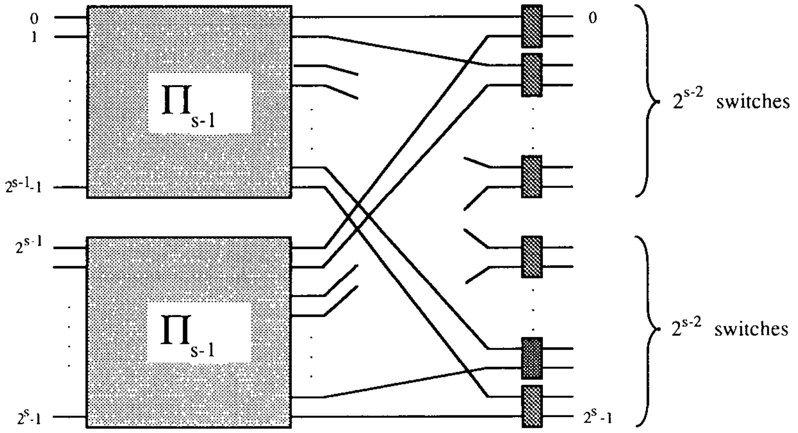

FIG. 2. The partial shuffle topology.

throughput. At present, switching times in optical switches are nonnegligible in contrast to the corresponding electronic devices and so it is preferable to hold a partial path rather than release the links when a request is blocked.

2.3 TERMINOLOGY. In order to associate various features of the delta network with more familiar terms, used, for example, in describing graphs and trees, we use certain words synonymously: the full crossbar switch is sometimes referred to as a *node* and connections between switches in adjacent stages are called *links*. A sequence of connected nodes and links between the network input pins and output pins is called a *path*, and any contiguous portion of a path which begins at an input pin is called a *partial path*. The *decode tree* of a network *output* pin comprises all paths to that pin, although in this paper we only need to consider the nodes in those paths. Likewise, the *decode tree* of a network *input* pin is composed of all paths originating from that pin. The pins in any stage of the network are numbered consecutively, starting at zero for the top pin, as shown in Figure 2. In addition to this numbering, each output pin belongs to a *class* such that class 0 contains the pin numbered 0 and class k contains all pins numbered 2^{k-1} to $2^k - 1$ inclusively ($k > 0$), as shown in Figure 3. These classes will be used to distinguish different degrees of hot-spot contention in Section 6.

In the analysis of nonuniform routing within delta networks, the top pin is taken to be the hot-spot, that is, the top pin address is chosen by an arriving customer with some probability ρ and all other addresses are chosen with the same probability $(1 - \rho)/(w - 1)$ in the case of w-way connection. The top pin will be referred to as the *hot pin* and the other pins as *cool pins*.

3. *Representation of Interconnection Networks in Queuing Models*

The interconnection network is a passive resource in that paths in it need to be held by active resources (DMA processors) for them to provide their service. This is an example of service blocking and is typically found in circuit-switched networks. There are several types of blocking that can occur in queuing networks and approximate solutions for such networks can be found in [1]. However, such methods become impractical when there are large numbers of servers and, in the following sections, we describe how aggregation techniques can be applied to a queuing network with an IN.

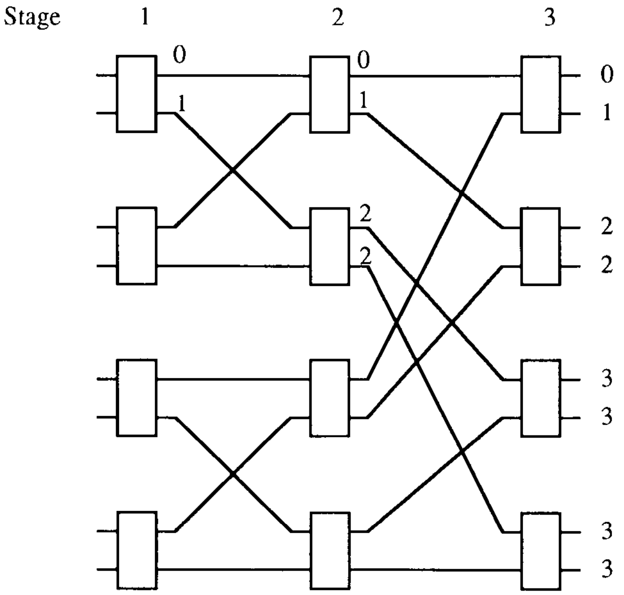

FIG. 3. Pin classes in an 8-way network.

3.1 FLOW-EQUIVALENT SERVER FOR THE INTERCONNECTION NETWORK. In a locally balanced queuing network, it is a standard result that any subnetwork can be replaced by a FES with service rate equal to the throughput of the *short-circuited* subnetwork when its population is equal to the FES queue length [4]. We apply the same method to subnetworks that comprise a bank of DMA servers together with some interconnection network, the remainder of the enclosing network satisfying the conditions for local balance. Under appropriate assumptions, the behavior of the short-circuited subnetwork can be represented by a stationary Markov process, and for each valid population we find its throughput and hence FES-rate by solving the balance equations of this process. The interconnection network is represented by a function that determines the effective service rate of the bank of DMA devices when they compete for network paths, and the appropriate functions for full crossbar and delta networks are derived in Sections 4.1 and 5.1, respectively; in the form of a recurrence relation that exploits the recursive structure of the network in the latter case. In Section 3.3, by considering the limiting case in which there is *no* degradation, that is, every active input pin of a network is always connected to a destination, the well-known expression for the throughput of a multiple server is obtained, and later we also derive a simple formula in the case that the interconnection network is a crossbar (Section 4.2).

3.2 UNDERLYING MARKOV PROCESS. The flow-equivalent server for b parallel DMA channels connected through an interconnection network is defined by the short-circuited network shown in Figure 4 for populations $N > 1$. When $N = 1$, the customer experiences no contention for links and so the throughput is the same as the DMA service rate. The shaded box represents the switching network that limits the number of active outputs, m, to some value between 1 and the number of active inputs, n, according to its internal connection topology.

The switching network may be a full crossbar or a delta network with circuit switching and partial paths held. In the steady state, when there are n active inputs

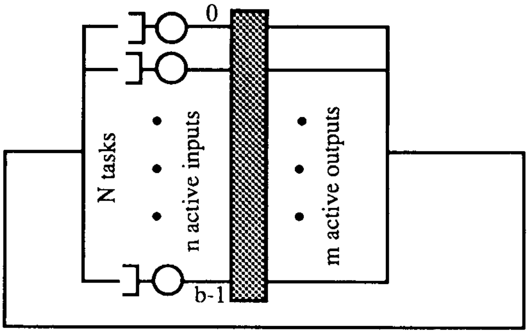

FIG. 4. The short-circuited subnetwork defining the FES rate.

to the network we assume that they are uniformly distributed over the network input pins. We also make the simplifying assumption that path set up and release times are negligible compared to service durations, that is, the path across the network or partial path to the first blockage will be established instantly. In practice, this is a reasonable assumption in networks where message lengths (and so DMA transfer times) are large when compared to crossbar switching times. For example, in the ALICE machine, the crossbar can switch in 85 ns whereas the mean transfer time is 14.4 μs.

The state space, Ω_1, for this closed system is defined by $\Omega_1 = \{(\mathbf{n}, m) \mid \sum n_i = N, \#(\mathbf{n}) \geq m \geq 1\}$, where the population is $N > 1$, $\mathbf{n} = (n_1, \ldots, n_b)$ is a vector of queue lengths at the source servers ($n_i \geq 0$, $1 \leq i \leq b$), m is a corresponding number of active output pins, and $\#(\mathbf{n})$ is the number of nonzero components of $\mathbf{n}$; we use n to denote $\#(\mathbf{n})$ where there is no confusion. Now let the random variable $N_i(t)$ denote the number of tasks at server i ($1 \leq i \leq b$), and $M(t)$ the number of active outputs at time $t \geq 0$. $X(t) = \{(\mathbf{N}(t), M(t)) \mid t \geq 0\}$ has finite state space Ω_1 and is a nonnull recurrent Markov process, with generators defined by appropriate state transition probabilities, under the assumption that all servers have negative exponentially distributed service times; this assumption can be relaxed for various queuing disciplines in standard ways, for example, [2].

We could now try to obtain the balance equations for the state space probabilities of $X(t)$ in the steady state by considering every possible state transition that may occur on a service completion. This results in a large number of equations and the approach is impractical even in the simple case of a full crossbar. Moreover, in a similar analysis of a delta network, the state space Ω_1 must be extended to represent every path and partial path established, vastly increasing computational complexity.

We therefore consider a simpler process, $Y(t)$, in which the switching network is represented by an expression ν_n for the expected number of active outputs when there are n active inputs in the steady state and we assume (approximately) that this relationship holds at all times. Thus, in the case of a direct connection (no switching network) every "input" will always be connected to an "output" and we have $\nu_n = n$, which does always hold exactly. We also use $\mu_n = \nu_n \mu$ to denote the *effective service rate function* of a bank of servers, each with rate μ, connected through such a network when n servers have nonempty queues. Thus, it is no longer necessary to include the number of active outputs in the state of the process $Y(t)$, since we can approximate this by ν_n when there are n active inputs. When the population of the short-circuited subnetwork is N, the state space is $\Omega_2 = \{\mathbf{n} \mid \sum n_i = N\}$ and we denote the stationary probability of $\mathbf{n} \in \Omega_2$ by $\pi(\mathbf{n})$.

3.3 UNDERLYING BIRTH AND DEATH PROCESS. We now aggregate all states $\mathbf{n}$ with the same number of nonzero components and consider the integer-valued, stationary Markov process $Z(t)$, which describes the number of active inputs to the switching network at time t and has state space $\Omega_3 = \{n \mid 1 \leq n \leq \min(b, N)\}$ with stationary probability p_n for $n \in \Omega_3$, that is, p_n is the steady-state probability that n input pins are active. The aggregate state n, therefore, represents the subset of states $S_n = \{\mathbf{n} \mid \#(\mathbf{n}) = n, \sum n_i = N\}$ of the process $Y(t)$, where the sets S_n form a partition of Ω_2.

For a given n, each state in S_n has the same steady-state probability with respect to the process $Y(t)$, that is, each arrangement of customers in any n nonempty queues is equally likely. This follows because the generators of the Markov process $Y(t)$ depend only on n and not directly on the component values n_i. The balance equations for $Z(t)$ then follow by expressing $\pi(\mathbf{n})$ in terms of p_n for each $\mathbf{n} \in \Omega_2$ [11, 24]. Of course, a stronger result holds in a simple Markovian queuing network (with no switching network) when the visitation rate to service rate ratio is the same for all servers, namely, that *all* states are equi-probable (cf. the multiple server result discussed below).

We use the following elementary combinatorial results in several places in this paper:

(i) The number of ways of arranging n balls in m boxes is

$$^{n+m-1}C_{m-1},$$

where the combinatorial function, C, is defined by

$$^{n}C_{m} = \frac{n!}{m!(n-m)!}. \tag{1}$$

(ii) The number of ways of arranging n balls in m *nonempty* boxes, that is so that there is at least one ball in each of the m boxes, is the same as the number of ways of arranging $n - m$ balls in m boxes (m having already been accounted for), that is,

$$^{n-1}C_{m-1}. \tag{2}$$

(iii) $$(n/m){}^{n-1}C_{m-1} = {}^{n}C_{m} \quad \text{for integers } n \geq m > 0. \tag{3}$$

Thus, we deduce from (ii) that for $n \geq m$, $\sum_{k=1}^{m} {}^{m}C_{k} {}^{n-1}C_{k-1} = {}^{m+n-1}C_{m-1}$ since if n balls are arranged in m boxes, they must occupy k nonempty boxes for some k, $1 \leq k \leq m$, and there are ${}^{m}C_{k}$ ways of selecting k boxes from m. This result follows formally from Lemma 4.1, which with its corollary we will find useful a number of times in this paper.

We can model the original network (Section 3.2) by $Z(t)$ provided we assume that:

—the effective service rate function is known,

—all arrangements of customers on n nonempty queues are equally likely, which has been shown to hold exactly when the IN is a crossbar as discussed above.

By analyzing process $Z(t)$ in the steady state, we can obtain the following theorem:

THEOREM 3.1. *Consider a b-way IN with known effective service rate function* μ_n *in a cyclic network comprising a parallel bank of DMA servers with* *population N. Under the assumption that all arrangements of customers on n nonempty queues are equally likely, the throughput is given by:*

$$T(N) = \sum_{n=1}^{b} \mu_n p_n, \tag{4}$$

where

$$p_n = \frac{p_1 \mu_1 \prod_{j=1}^{n-1} \{(b-j)(N-j)\}}{\mu_n (n-1)!^2} \tag{5}$$

and

$$p_1^{-1} = \sum_{n=1}^{b} \frac{\mu_1 \prod_{j=1}^{n-1} \{(b-j)(N-j)\}}{\mu_n (n-1)!^2}. \tag{6}$$

PROOF. State transitions of the process $Z(t)$ can only occur on completion of service by a task which then recycles to join one of the b queues. Thus there are only two possible types of state transitions: $n \longrightarrow n+1$ and $n+1 \longrightarrow n$ $(1 \le n < b)$ since transitions $\mathbf{n} \longrightarrow \mathbf{n}'$ of the process $Y(t)$ with $\#(\mathbf{n}) = \#(\mathbf{n}')$ do not cause a change of state in $Z(t)$. For the first type of transition to be possible, the completing task must depart from a queue in which there is at least one other task (length > 1), and must then join one of the $(b-n)$ empty queues. Let $q_{i,j}$ be the probability that the state changes from i to j on a service completion. Since states (of $Y(t)$) with the same number of nonempty queues are equi-probable, the transition $n \longrightarrow n+1$ occurs with probability $q_{n,n+1}$ given by:

$$q_{n,n+1} = \frac{(b-n)}{b}\,\mathrm{Pr}(\text{queue length at a given active input} > 1 \text{ when } n \text{ are active})$$

$$= \frac{(b-n)}{b}\,\frac{\#\,\text{arrangements of } N-1 \text{ tasks in } n \text{ nonempty queues}}{\#\,\text{arrangements of } N \text{ tasks in } n \text{ nonempty queues}},$$

which by (2) becomes:

$$= \frac{(b-n)}{b}\,\frac{{}^{N-2}C_{n-1}}{{}^{N-1}C_{n-1}} = \frac{(b-n)}{b}\,\frac{(N-n)}{(N-1)}. \tag{7}$$

Conversely, for the second type of transition, the completing task must leave from a queue in which there are no other tasks and must then join one of the n nonempty queues. The transition $n+1 \longrightarrow n$, therefore, has probability $q_{n+1,n}$ given by:

$$q_{n+1,n} = \frac{n}{b}\,\frac{\#\text{ arrangements of } N-1 \text{ tasks in } n \text{ nonempty queues}}{\#\text{ arrangements of } N \text{ tasks in } n+1 \text{ nonempty queues}}$$

$$= \frac{n}{b}\,\frac{{}^{N-2}C_{n-1}}{{}^{N-1}C_{n}} = \frac{n^2}{b(N-1)} \tag{8}$$

Now from (7) and (8), given that the effective service rate function of the switching network is μ_n, we have the following balance equations for the process $Z(t)$ with $1 \le n < b$:

$$\mu_n p_n (b-n)(N-n) = \mu_{n+1} p_{n+1} n^2. \tag{9}$$

Solving this for p_n gives (5), with normalizing constant given by (6). Equation (4) follows from this distribution and the effective service rate function. □

When the effective service rate function μ_n is a simple expression, it is not necessary to compute the distribution p_n directly to obtain the throughput $T(N)$ by Theorem 3.1. In such cases an expression for throughput may be obtained by the following theorem:

THEOREM 3.2. *For a b-way IN with effective service rate function μ_n, in a closed system of a parallel bank of DMA servers with population N, the throughput is given by*:

$$T(N) = \frac{bN}{b+N-1} H(1), \tag{10}$$

where

$$H(z) = \sum_{n=1}^{\infty} \frac{t_n z^n}{n} \tag{11}$$

and t_n is given by the following recurrence formula and boundary condition, respectively:

$$(b-n)(N-n)t_n = n^2 t_{n+1} \qquad 1 \le n < b \tag{12}$$

and

$$\sum_{n=1}^{b} \frac{t_n}{\mu_n} = 1. \tag{13}$$

PROOF. Substituting $t_n = \mu_n p_n$ in the balance eq. (9) and the normalizing condition for p_n we obtain the recurrence formula (12) and the boundary condition (13). We can solve this by defining the generating function

$$G(z) = \sum_{n=1}^{\infty} t_n z^n \qquad t_n = 0 \qquad \text{for} \quad n > b.$$

$G(z)$ has first derivative

$$G'(z) = \sum_{n=1}^{\infty} n t_n z^{n-1}$$

and we define

$$H(z) = \int_0^z \frac{G(u)}{u}\, du.$$

We now have $T(N) = G(1)$ and we can rewrite the recurrence formula (12) as

$$Nb\left(\frac{t_n}{n}\right) - (N+b)t_n + nt_n + t_{n+1} - (n+1)t_{n+1} = 0.$$

Multiplying by z^n and summing yields

$$NbH(z) - (N+b)G(z) + zG'(z) + z^{-1}(G(z) - t_1 z) - G'(z) + t_1 = 0,$$

which simplifies to

$$(1-z)G'(z) + (N+b-z^{-1})G(z) - NbH(z) = 0.$$

Thus, setting $z = 1$, we obtain (10), the required expression for the throughput $T(N)$. □

In Theorem 3.2, $H(1)$ is derived from the boundary condition (13), which depends on the specific form of μ_n given by the characteristics of the switching network. In the simplest exact case where we have direct connection (no switching network), $\mu_n = n\mu$ and the boundary condition (13) becomes $\sum_{n=1}^{b} t_n/n\mu = 1$, that is, $H(1) = \mu$, giving the well-known "multiple server" formula for $T(N) = bN/(b + N - 1)\mu$ derived by a number of authors, for example, [12]. There is a similarity between the expression for $T(N)$ when there is a direct connection and the expression for μ_n when there is a full crossbar (given in the next section). In fact, the two expressions can be unified by the substitution, $N = n$. As we see shortly, this is to be expected because both expressions can be derived by purely probabilistic arguments that use the fact that each arrangement of tasks over a fixed number of nonempty queues occurs with equal probability in the steady state.

3.4 THE EXACT RESULT FOR FULL CROSSBARS. Rather than assume that the output pin service rate is always the mean value μ_n and solving the simple process $Z(t)$, we may be a little more precise and consider the process $X'(t)$, defined by $X'(t) = \{\#(\mathbf{N}(t)), M(t)\}$, instead of $X(t)$. The state space of $X'(t)$ is then $S' = \{(n, m) \mid 1 \le m \le n \le b\}$ and for full crossbars, we may solve for the throughput of the short-circuited subnetwork considered in the previous sections by solving for the equilibrium state space probabilities of S' directly for any population N.

We define the steady-state probability distribution of the stationary Markov process $X'(t)$, $P(n, m)$ for $(n, m) \in S'$, by

$$P(n, m) = \lim_{t\to\infty} \Pr(\#(\mathbf{N}(t)) = n, M(t) = m)$$
$$= \sum_{\substack{\#(\mathbf{n})=n \\ (\mathbf{n},m)\in\Omega_1}} P(\mathbf{n}, m),$$

where $P(\mathbf{n}, m)$ is the corresponding equilibrium probability for the state $(\mathbf{n}, m)$ of the process $X(t)$.

Now, the number of arrangements of N tasks over n given nonempty queues is ${}^{N-1}C_{n-1}$ (by (2)) and we know that given $(n, m) \in S'$, each state $(\mathbf{n}, m) \in \Omega_1$ with $\#(\mathbf{n}) = n$ has the same steady-state probability,

$$\frac{P(\mathbf{n}, m)}{{}^{N-1}C_{n-1}{}^{b}C_{n}}$$

for a b-way crossbar since all transition probabilities depend only on n and m (recall the argument in the previous section).

In this way, we can obtain the following equations for the stationary probabilities, $P(n, m)$, of the Markov process $X'(t)$, for $(n, m) \in S'$:

$$\begin{aligned}
&P(n,m)mb(N-1)\\
&\quad= (b-n+1)(n-1)(N-n+1)\\
&\qquad\cdot \{P(n-1,m+1)\theta_1(n-1,m+1) + P(n-1,m)\theta_2(n-1,m)\\
&\qquad+ P(n-1,m-1)\theta_3(n-1,m-1) + P(n-1,m-2)\theta_4(n-1,m-2)\}\\
&\qquad+ n^2(n+1)\{P(n+1,m+1)\phi_1(n+1,m+1) + P(n+1,m)\phi_2(n+1,m)\}\\
&\qquad+ n\{n(N-n) + (n-1)(b-n+1)\}\\
&\qquad\cdot \{P(n,m+1)\varphi_1(n,m+1) + P(n,m)\varphi_2(n,m) + P(n,m-1)\varphi_3(n,m-1)\},
\end{aligned}$$

for appropriately defined parameters $\boldsymbol{\theta}$, $\boldsymbol{\phi}$, $\boldsymbol{\varphi}$ given in [11] and [24]. Notice that in general, after a service completion, the state (n, m) can transit to a state (n', m')

with $m' = m - 1$ (recycling task joins a nonempty queue and all queues are blocked or transmitting already through the $m - 1$ servers), m, $m + 1$ or $m + 2$ (another task becomes unblocked by the service completion, the next task in the completing task's queue enters service at a *different* free server, and the recycling customer enters an empty queue and is not blocked). The state space S' is only of order b^2 and so it is quite feasible to compute performance measures for full crossbars by solving the above equations directly. Moreover, as we shall see in later sections, once computed, these results may be used as base cases in the recursion that yields the corresponding performance measures for delta networks.

In the case of a 2-way crossbar switch, $b = 2$, with $N > 1$, the values of these parameters are $\theta_2(1, 1) = \theta_3(1, 1) = \frac{1}{2}$, $\phi_1(2, 2) = 1$, $\phi_2(2, 1) = \frac{1}{2}$, $\varphi_1(2, 2) = \frac{1}{2}$, $\varphi_2(1, 1) = 1$, $\varphi_2(2, 1) = \frac{1}{4}$, $\varphi_2(2, 2) = \frac{1}{2}$, $\varphi_3(2, 1) = \frac{1}{4}$, and the (dependent) equations become:

$$\begin{aligned} (N - 1)P(1, 1) &= 2P(2, 2) + P(2, 1), \\ (2N - 1)P(2, 1) &= (N - 1)P(1, 1) + 2(2N - 3)P(2, 2), \\ 2(2N - 1)P(2, 2) &= (N - 1)P(1, 1) + (2N - 3)P(2, 1). \end{aligned}$$

These equations have normalized solution:

$$P(1, 1) = \frac{4}{3N + 1}, \qquad P(2, 1) = \frac{2(N - 1)}{3N + 1}, \qquad P(2, 2) = \frac{(N - 1)}{3N + 1}, \tag{14}$$

giving a throughput

$$T(N) = 1.P(1, 1) + 1.P(2, 1) + 2.P(2, 2) = \frac{4N}{3N + 1}. \tag{15}$$

Thus, throughput is always less than $\frac{4}{3}$, and approaches this value as $N \to \infty$. This is as expected for a full 2-way crossbar with both of its inputs active (see Section 4.1 for example). For finite N, there is always a nonzero probability that an input is inactive, giving lower throughput.

4. *FES for DMA Servers with Full Crossbar*

4.1 EFFECTIVE SERVICE RATE FUNCTION FOR CROSSBARS. For an $a \times b$ crossbar, where outputs are randomly selected and inputs are equally utilized, given n active inputs, the probability that m outputs are active, $p_b(m \mid n)$, is easily determined by ball-in-box arguments since by symmetry every valid arrangement of the n and m (active) input and output pins over b pins is equally likely in the steady state. From this follows the mean number of active outputs, conditional on n active inputs, and hence conditional throughput, μ_n.

We require the following lemma, which has appeared in [17].

LEMMA 4.1. *For* $n \geq 1, 0 \leq h \leq X - n + 1$,

$$\sum_{k=1}^{n} {}^{n-1}C_{k-1}\,{}^{X}C_{k+h-1} = {}^{n+X-1}C_{X-h}$$

PROOF. For $n = 1$, the left-hand side, lhs $= {}^{X}C_h = {}^{X}C_{X-h} =$ rhs, the right-hand side if $0 \leq h \leq X$. Now assume inductively that the result is true for $1 \leq n \leq N$.

Then,

$$\sum_{k=1}^{N+1} {}^{N}C_{k-1}{}^{X}C_{k+h-1} = \sum_{k=2}^{N} ({}^{N-1}C_{k-2} + {}^{N-1}C_{k-1}){}^{X}C_{k+h-1} + {}^{X}C_{N+h} + {}^{X}C_{h}$$
$$= \sum_{k=1}^{N} ({}^{N-1}C_{k-1}{}^{X}C_{k+h} + {}^{N-1}C_{k-1}{}^{X}C_{k+h-1})$$

(we have replaced the dummy summation variable, k, by $k + 1$ in the first sum, and extended each summation domain to include the two loose terms).

Therefore, using the inductive hypothesis twice, for $0 \le h + 1 \le X - N + 1$ and $0 \le h \le X - N + 1$, that is, for $0 \le h \le X - N$,

$$\text{lhs} = {}^{N+X-1}C_{X-h-1} + {}^{N+X-1}C_{X-h} = {}^{N+X}C_{X-h} = \text{rhs}. \qquad \Box$$

Corollary 4.2. *For $n \ge 1, X \ge 1, 0 \le h \le X - n$,*

$$\sum_{k=1}^{n} {}^{n-1}C_{k-1}{}^{X}C_{k+h}.(k + h) = X.{}^{X+n-2}C_{X-h-1}.$$

Proof. Since ${}^{X}C_{k+h}.(k + h) = X.{}^{X-1}C_{k+h-1}$,

$$\text{lhs} = X. \sum_{k=1}^{n} {}^{n-1}C_{k-1}{}^{X-1}C_{k+h-1} = \text{rhs}$$

by the lemma. □

Now let the random variables M, N denote the numbers of active outputs and inputs, respectively. Then,

$$p_b(m \mid n) = \Pr(M = m \mid N = n) = \frac{{}^{b}C_{m}{}^{n-1}C_{m-1}}{{}^{b+n-1}C_{b-1}}.$$

Then by Lemma 4.1, $\sum_{m=1}^{\min(n,b)} p_b(m \mid n) = 1$ (rearranging as above when $b < n$), and by Corollary 4.2 with $h = 0$, the expected number of active outputs when there are n active inputs is given by:

$$\sum p_b(m \mid n).m = b.\left(\frac{{}^{b+n-2}C_{b-1}}{{}^{b+n-1}C_{b-1}}\right) = \frac{bn}{(b + n - 1)},$$

which, in the terminology of the previous section, gives a degraded service rate function

$$\mu_n = \frac{bn}{b + n - 1}\mu. \tag{16}$$

4.2 Throughput Function for a Crossbar. Now that we have a simple expression for μ_n in the case for a full crossbar, we can use Theorem 3.2 to obtain a simple expression for the throughput $T(N)$.

Corollary 4.3. *For closed system with population N, consisting of b parallel DMA servers transmitting across a $b \times a$ crossbar network, the throughput is given by:*

$$T(N) = \frac{abN\mu}{(a + b - 1)N + (a - 1)(b - 1)}.$$

PROOF. For a $b \times a$ crossbar network, we obtain the following conditional throughput function from (16),

$$\mu_n = \frac{an}{a + n - 1}\mu$$

from which the boundary condition (13) becomes

$$G(1) + (a - 1)H(1) = a\mu.$$

Substituting $T(N) = G(1)$ and using (10) from Theorem 3.2 we obtain

$$\left\{\frac{bN}{b + N - 1} + (a - 1)\right\}H(1) = a\mu$$

from which the result follows by substituting $H(1)$ in (10). □

In the case of a 2 × 2 crossbar for example, substituting $b = 2$ and (16) in (5) and simplifying gives

$$p_1 = \frac{4}{3N + 1} \quad \text{and} \quad p_2 = \frac{3(N - 1)}{3N + 1},$$

which are the marginal probabilities of (14), the solution obtained by solving the balance equations of the process $X'(t)$ directly (see Section 3.4). Also using the probabilities we derived in Section 3.4,

$$T(N) = \frac{4N}{3N + 1}\mu$$

which agrees with the above corollary. Hence, our simpler analysis of INs using process $Z(t)$ and effective service rate function μ_n is exact with respect to the more precise analysis in Section 3.4 for a 2 × 2 crossbar.

In the case of a multistage delta network, the expression for μ_n is given by a recurrence formula as derived in the following section, and so a compact expression for throughput would appear impossible to derive. We have obtained our numerical predictions (see Section 7) of throughput $T(N)$ in this case from the degraded rates μ_n by directly computing the set of probabilities p_n (using (5) and (6)), and hence $T(N)$ from (4) in Theorem 3.1.

5. *Conditional Throughput Function for a Uniform Delta Network*

We analyze delta networks by considering the behavior of the uppermost rightmost crossbars in each of the subnetworks involved in the recursive definition. This analysis is based upon applications of Little's result [20], and so initially requires no assumptions about the distributions of task service times, only that the crossbar is in stochastic equilibrium. In order to obtain a recursive solution, however, we make certain approximating assumptions to determine the blocking probability of an arriving task and the probability that it "sees" the other input pin of the crossbar already held by another task. We see that these assumptions are very mild, especially in the case of a saturated delta network.

First, we define some terms, noting that a crossbar connected to servers with appropriate service-time distributions can always be described by a stationary stochastic process (i.e., one with a steady-state or equilibrium state space distribution).

5.1 DEFINITIONS FOR A 2-BY-2 CROSSBAR. Let the state of a 2×2 crossbar be the binary 4-tuple (I_0, I_1, U_0, U_1) describing the states (active or inactive) of its upper (I_0) and lower (I_1) input pins and output pins (U_0, U_1), similarly. In the steady state, we define the following:

—A pin's *mean holding time* (*MHT*) is the expected elapsed time between the pin becoming active and the departure of the task holding it after its service completion (i.e., the pin next becoming inactive, even if instantaneously).

—An active output pin's *mean residual holding time* (*MRHT*) is the expected elapsed time between an arrival at the crossbar and the pin next becoming inactive (possibly instantaneously).

—The equilibrium state probabilities are denoted by

$$\pi(i_0 i_1 u_0 u_1) \qquad i_0, i_1, u_0, u_1 \in \{0, 1\}.$$

We make the following abbreviations for the marginal probabilities over an $(n+1)$-dimensional state space (here $n = 3$):

$$\pi(x_0 \cdots x_n) = \sum_{\substack{y_i \in \{0,1\}(x_i = *) \\ y_i = x_i \text{otherwise}}} \pi(y_0 \cdots y_n)$$

$$\pi(x_0 \cdots x_m) = \pi(x_0 \cdots x_m * \cdots *) \qquad \text{for} \quad 0 \le m \le n$$

(where there are $n - m$ *'s on the right-hand side).

—The *blocking probability*, b_i, for input pin i ($i = 0, 1$) is the steady-state probability that, at the instant a task arrives on that pin, the output pin it requires is already held by a task currently holding the other input pin.

—The crossbar is said to possess the *arriving observer property* (*AOP*) if the steady-state probability that an input pin is active at the instant a task arrives on the other input pin is equal to the equilibrium (marginal) probability that the former input pin is active.

5.2 A 2-BY-2 CROSSBAR IN EQUILIBRIUM.

LEMMA 5.1. *For a 2×2 crossbar with output pins that*

(*a*) *are selected with the same probability by incoming traffic*
(*b*) *have the same MHT, m, and MRHT, d,*

the probability that either output pin is active is

$$p = \frac{m}{2}\left\{\frac{\pi_0}{m + b_0 d} + \frac{\pi_1}{m + b_1 d}\right\},$$

where π_0, π_1 are the utilizations of the upper and lower inputs, respectively.

PROOF. Let the arrival rates (i.e., reciprocals of the mean interarrival times) on the upper and lower input pins be λ_0 and λ_1, respectively. Three applications of Little's result then yields:

(i) For an output pin, $p = [(\lambda_0 + \lambda_1)/2]m$, since the probability that the pin is active is the same as its mean queue length;
(ii) For the upper input pin, $\pi(1) = \lambda_0(m + b_0 d)$ similarly;
(iii) For the lower input pin, $\pi(*\ 1) = \lambda_1(m + b_1 d)$ similarly.

Since $\pi(1) = \pi_0$ and $\pi(*\ 1) = \pi_1$, the result follows by eliminating λ_0 and λ_1. □

COROLLARY 5.2

(*a*) *Assuming the crossbar has the AOP and that $m = d$*

$$p = \frac{\pi_0}{2 + \pi_1} + \frac{\pi_1}{2 + \pi_0}.$$

(*b*) *For a crossbar with $\pi = \pi_0 = \pi_1$, for example, one in which the input processes are the same for each input pin, which has the AOP and $m = d$,*

$$p = \frac{2\pi}{2 + \pi}.$$

(*c*) *If further the crossbar is saturated, that is, $\pi = 1$,*

$$p = \tfrac{2}{3}.$$

PROOF. Since output pins are selected with equal probability, $b_i = \frac{1}{2}\pi_{1-i}$ ($i = 0, 1$) by the AOP. □

LEMMA 5.3. *The steady-state probabilities for a single crossbar with output pins selected with equal probability may be written in terms of marginal probabilities as follows*:

(*a*) $\pi(0\ 1\ 0\ 1) = \pi(0\ 1\ 1\ 0) = \frac{1}{2}\pi(0\ 1)$
(*b*) $\pi(1\ 0\ 0\ 1) = \pi(1\ 0\ 1\ 0) = \frac{1}{2}\pi(1\ 0)$
(*c*) $\pi(1\ 1\ 0\ 1) = \pi(1\ 1\ 1\ 0) = d'/(2m' + d')\pi(1\ 1)$
(*d*) $\pi(1\ 1\ 1\ 1) = (2m' - d')/(2m' + d')\pi(1\ 1)$

where m', d' are, respectively, the MHT and MRHT of each output pin, conditional on both the crossbar's input pins being active.

PROOF. Cases (a) and (b) follow by symmetry. For the other cases, we apply Little's result in the steady state at times when both inputs are active, that is, the total queue length is 2. Let the throughput of each output pin be τ (the same for each by hypothesis), and the steady-state probability that it is active, conditional on both inputs being active, be p'.

Then Little's result applied to an output pin gives $p' = \tau m'$ and to the whole switch $2 = 2\tau\{m' + \frac{1}{2}d'\}$. Thus, $p' = 2m'/(2m' + d')$ and since $1 - p' = \pi(1\ 1\ 0\ 1)/\pi(1\ 1)$ the result (c) follows. Result (d) follows from

$$p' = \frac{\pi(1\ 1\ 1\ 0) + \pi(1\ 1\ 1\ 1)}{\pi(1\ 1)} \qquad □$$

COROLLARY 5.4. *If the MHT = MRHT,*

$$\pi(1\ 1\ 0\ 1) = \pi(1\ 1\ 1\ 0) = \pi(1\ 1\ 1\ 1) = \tfrac{1}{3}\pi(1\ 1),$$

and the throughput of the crossbar is $\frac{4}{3}$ when both its inputs are active.

5.3 EXACT ANALYSIS OF THE DELTA-2 NETWORK. The steady-state probability that a given output pin in any stage of a delta network is active satisfies a recurrence relation derived by considering a single crossbar, using Lemma 5.1, and the network's recursive structure.

Let m_s, d_s be the MHT and MRHT, respectively, for output pins at stage s in a uniform delta network with J stages ($1 \le s \le J$). Thus, $m_J = d_J = \mu^{-1}$ if the servers connected to the last stage are exponential with parameter μ. Let b_s be the blocking probability for an arrival on any input pin at stage s. Then, we have the following:

PROPOSITION 5.5. *The equilibrium probability π_s that an output pin (of any crossbar) in stage s of a J-stage delta network is active satisfies*

$$\pi_s = \frac{\pi_{s-1}}{1 + b_s\alpha_s} \qquad 1 \le s \le J,$$

where $\alpha_s = d_s/m_s$ and π_0 is the equilibrium probability that an input pin in stage 1 *is active.*

PROOF. Any crossbar in stage s $(1 \le s \le J)$ satisfies the conditions of Lemma 5.1 since its output pins are stochastically identical and clearly a steady state exists since the Markov process representing the state of *every* pin in the network is ergodic. □

5.4 SIMPLIFYING ASSUMPTIONS. In the sequel, apart from the next section, we make two simplifying assumptions that, for all pins in the network:

(i) MHT = MRHT, that is, that $m_s = d_s$
(ii) the AOP holds so that $b_s = \frac{1}{2}\pi_{s-1}$

Under these intuitively reasonable assumptions, the recurrence in Proposition 5.5 becomes solvable in closed form for the saturated, uniform case. In general, these assumptions are both approximations in that they do not hold at every stage of the network. This is easily seen in assumption (i) which requires *all* pin holding time distributions to be exponential, although this property does hold in the final stage of a network connected to exponential servers.

If we assume the servers connected to the final stage of the network are exponential, then all holding time distributions are mixtures of Erlang distributions with the same parameter. Thus, if arrivals at any crossbar's input pin are *random* with respect to the holding times of the other input pin,

$$\frac{1}{2} \le \frac{d_s}{m_s} \le 1,$$

since the mean residual service time for an Erlang-k distribution with parameter μ is $(k + 1)/2\mu$. In fact arrivals are not random and the MRHT is much closer to the bound given by our approximation. This is because the service completion that caused the arrival may have also just unblocked another task and so the new arrival may be blocked at a pin that has just become active with significantly nonzero probability. In this case, we have $d_s = m_s$, but we may also have $d_s > m_s$ if an arrival at one pin tends to occur more frequently during long intervals of activity of the other pin. Since the pin-holding time distribution seen by an arrival is not known, it is difficult to determine whether $d_s < m_s$ or $d_s > m_s$. However, the assumption that $d_s = m_s$ certainly appears very reasonable.

The more critical assumption is that the AOP holds. Intuitively, it should not hold exactly, but it should provide a good approximation. In the next section, we prove these two conjectures for a two-stage network with exponential servers. This suggests that the AOP cannot hold everywhere in an arbitrary sized delta network which includes two-stage subnetworks, but this would appear difficult to establish for arbitrary holding time distributions, and not an important issue in view of the other approximating assumption.

In Section 5.5, we compare the approximate method with the exact solution for a two-stage saturated, uniform network and show that the AOP does not hold by deriving explicitly the steady-state probability that an input pin is active at the instant a task arrives on the other input pin of a crossbar in the final stage. The approximating assumptions do turn out to be very mild and yield extremely accurate results when compared with both the exact solution in the two-stage case and with simulations of larger networks.

5.5 ANALYSIS OF ARRIVING OBSERVER. In this section, we consider a two-stage delta-2 network connected to four exponential servers. In the steady state, the probability that a crossbar's input pin is active at the instant a task arrives on the other input pin can be determined by first finding the proportion of time in the long term that the system spends in states, s' say, that *can* provide an arrival to the upper (say) input pin, via a transition into an *input state*, s, say. By the Key Renewal Theorem, see [5] for example, the unnormalized equilibrium probability that the system is in state s immediately after the arrival instant is the expected number of transitions into state s in unit time, which can be determined from the equilibrium probabilities of the states s' and the transition rates from s' to s. This is the approach taken by [27], which considers the distribution of similar input (and output) states in product-form queuing networks.

5.5.1 *Notation.* Let $S = \{\mathbf{b} \mid b_j = 0, 1; 0 \leq j \leq 3\}$ be the set of binary quadruples describing the (marginal) states of the output pins (numbered j from the top) in the *first* stage of the network. An active pin is denoted by 1 and an inactive one by 0.

Let $A = \{s \in S \mid s_0 = 1\}$ be the set of states that can exist immediately after an arrival at the top input pin of the second stage.

For $s \in A$, let $B(s) = \{s' \in S \mid \exists$ a one-step transition $s' \longrightarrow s\}$.

Let $T(s', s)$ denote the server at which a service completion can cause the transition $s' \longrightarrow s$.

Since we are considering a renewal process, the expected number of transitions into state $s \in A$ in unit time is

$$\Phi_s = \sum_{s' \in B(s)} \pi(s') p_{s's},$$

where we assume without loss of generality that the servers have rate 1 and

—$\pi(x)$ is the equilibrium probability for state $x \in S$, that is, the proportion of time in the long term that the system spends in state x,

—$p_{s's}$ is the probability that a transition from state s' enters state s, conditional on that transition being the result of a service completion at server $T(s', s)$.

5.5.2 *State Seen by an Arriving Observer.* The steady-state probability that an arrival at the top input pin of the second stage finds the other input pin of the same (top) crossbar active is therefore Φ^*/Φ, where

$$\Phi^* = \sum_{\substack{s \in A \\ s_2 = 1}} \Phi_s \qquad \text{and} \qquad \Phi = \sum_{s \in A} \Phi_s.$$

By calculating Φ_s individually for every $s \in A$, we can indeed determine Φ^*/Φ but the method is extremely laborious and we can do better by considering departures rather than arrivals and exploiting the symmetry of the model as follows.

Φ^* is the long-term arrival rate on input pin 0 of the upper crossbar in the second stage when the other input pin of that crossbar is active. By symmetry, this is a half of the *total* long-term rate of arrivals to the crossbar that result in a state with both input pins active. In the steady state, this is the same as half the departure rate (i.e., throughput) from the crossbar's output pins in states with both its input pins active. Thus, by Corollary 5.4, we have

$$\Phi^* = \frac{1}{2}\frac{4}{3}\pi(1 * 1 *).$$

Similarly, $\Phi^0 \triangleq \Phi - \Phi^*$ is the total long-term rate of arrivals to the same crossbar that result in only the top input pin being active. In the steady state, this is equal to the departure rate from the state with the upper input pin active and the lower one inactive. Thus, we have

$$\Phi^0 = \pi(1 * 0 *) = \pi(1 * 0\ 1) \qquad \text{for a saturated network.}$$

The ratio Φ^*/Φ can therefore be computed using the following equilibrium marginal probabilities (which use the symmetry of this network):

$$\begin{aligned} p &\triangleq \pi(1\ 0\ 1\ 0) = \pi(0\ 1\ 0\ 1), \\ p' &\triangleq \pi(1\ 0\ 0\ 1) = \pi(0\ 1\ 1\ 0), \\ q &\triangleq \pi(1\ 1\ 1\ 0) = \pi(1\ 1\ 0\ 1) = \pi(1\ 0\ 1\ 1) = \pi(0\ 1\ 1\ 1), \\ r &\triangleq \pi(1\ 1\ 1\ 1) = 1 - 2p - 2p' - 4q. \end{aligned}$$

This gives

$$\begin{aligned} \frac{\Phi^*}{\Phi} &= \frac{(2/3)\pi(1 * 1 *)}{(2/3)\pi(1 * 1 *) + \pi(1 * 0\ 1)} \\ &= \frac{2(p + 2q + r)}{2p + 3p' + 7q + 2r}. \end{aligned}$$

Now, if the AOP held, we would have $\Phi^*/\Phi = \pi(* * 1 *) = p + p' + 3q + r$, which we cannot establish since there are no more independent identities. In fact, exact analysis of the Markov process (see [24]) yields $\Phi^*/\Phi = 1454/2179 = 0.667279$ to six decimal places.

If we assume that MHT = MRHT for the first stage as well as the second, as in the approximate analysis for delta networks, this ratio is 2/3. We might therefore expect that our approximation will be very good, and this is borne out in Section 7. In fact for the two-stage case, the exact throughput is 17432/8719 = 1.999312 to six decimal places compared with the approximate result of 2.

5.6 Recursive Analysis of the Delta Network. In this section, we derive the throughput of a delta network with uniform routing, conditional on the number of active inputs to the network. For an s-stage delta-2 network ($s \geq 1$), we define the set $V_s = \{(n_0, \ldots, n_{2^s-1}) \mid n_i \in \{0, 1\}, 0 \leq i < 2^s\}$ and the following random variables:

(i) $\mathbf{N} \in V_s$, which represents the state of the input pins to the network ($N_i = 0$ means that input pin i is inactive and $N_i = 1$ means that it is active, that is, the DMA server connected to it is wishing to transmit).
(ii) $\mathbf{Z} \in V_s$, which represents the state of each network output pin.

We also define the function # for which $\#(\mathbf{V})$ is the number of nonzero components in the vector $\mathbf{V}$ and the analogous function $\#_1$ for which $\#_1(\mathbf{V})$ is the number of nonzero components in the *first half* of the vector $\mathbf{V}$.

We require the following probability distribution: $Q_s(i \mid n) = \Pr(\#_1(N) = i \mid \#(N) = n)$ which is the probability that i active inputs are in the upper part of the s-stage network (i.e., the first 2^{s-1} pins) given that there are n active inputs altogether. By symmetry, when there are n active inputs to the network, they are uniformly distributed over the 2^s input pins. Since all arrangements of the n active inputs are equally likely:

$$Q_s(i \mid n) = \frac{{}^kC_i\,{}^kC_{n-i}}{{}^{2k}C_n} \quad \text{where} \quad k = 2^{s-1}. \tag{17}$$

We also define the following probability distribution:

$$T_s(n) = \Pr(Z_0 = 1 \mid \#(N) = n),$$

which represents the probability that the topmost output pin of the s-stage network is active given that there are n active inputs to the network. This distribution can be defined by using the recursive structure of the delta network with the partial shuffle topology (Section 2).

Consider the rightmost, topmost switch in an s-stage network. This takes its inputs from two separate $(s-1)$-stage subnetworks (Figure 5). When we condition on the number of active inputs to each subnetwork (their sum being $\#(N) = n$), the output pin utilization of either network can be found. Hence, we can determine the utilization of both the inputs to the rightmost, topmost crossbar by using the result obtained for the next smaller networks in the recursion. The distribution of the state of the top output pin then follows by considering a single crossbar and the results of the previous section. This gives the recurrence formula in the following theorem:

THEOREM 5.6. *Under the assumptions that every crossbar in the network has the AOP and outputs with MRHT = MHT, the output pin utilization of an s-stage delta-2 network conditional on the number of active inputs being n is given by the following*:

For $s > 1$,

$$T_s(n) = \sum_{i=max(0,n-2^{s-1})}^{min(n,2^{s-1})} Q_s(i \mid n)U(T_{s-1}(i),\, T_{s-1}(n-i)) \tag{18}$$

$$T_1(0) = 0, \qquad T_1(1) = U(0, 1), \qquad T_1(2) = U(1, 1) \tag{19}$$

where

$$U(\pi_0, \pi_1) = \frac{\pi_0}{2 + \pi_1} + \frac{\pi_1}{2 + \pi_0}.$$

PROOF. For an s-stage delta network ($s > 1$), we define the following random variables (see Figure 5):

$Z \in \{0, 1\}$ the state of output pin 0 of an s-stage network,
$I_0 \in \{0, 1, 2, \ldots, 2^{s-1}\}$ the number of active inputs to the upper $(s-1)$-stage subnetwork,
$I_1 \in \{0, 1, 2, \ldots, 2^{s-1}\}$ the number of active inputs to the lower $(s-1)$-stage subnetwork,

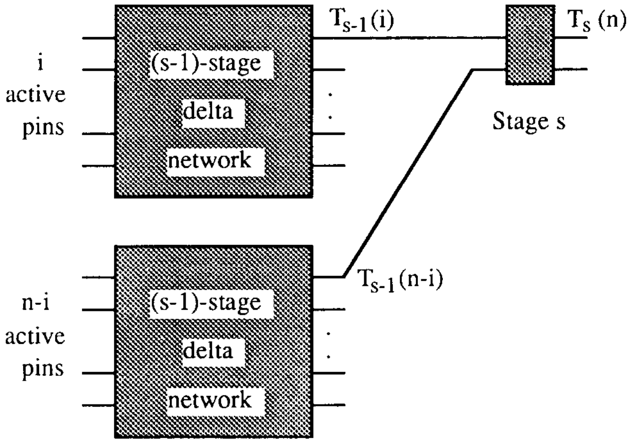

FIG. 5. Recursive structure of an s-stage delta network.

$$\begin{aligned}\Pr(Z=1, I_0+I_1=n) &= \sum_{i=0}^{n} \Pr(Z=1, I_0=i, I_1=n-i)\\ &= \sum_{i=0}^{n} \Pr(I_0=i, I_1=n-i)\Pr(Z=1 \mid I_0=i, I_1=n-i),\\ \Pr(Z=1 \mid I_0+I_1=n) &= \sum_{i=0}^{n} \Pr(I_0=i, I_1=n-i \mid I_0+I_1=n)\Pr(Z=1 \mid I_0=i, I_1=n-i).\end{aligned}$$

The recurrence (18) follows from the following substitutions:

$$\begin{aligned}T_s(n) &= \Pr(Z = 1 \mid I_0 + I_1 = n),\\ Q_s(i \mid n) &= \Pr(I_0 = i,\ I_1 = n - i \mid I_0 + I_1 = n),\\ U(T_{s-1}(i),\ T_{s-1}(n - i)) &= \Pr(Z = 1 \mid I_0 = i,\ I_1 = n - i),\end{aligned}$$

where $U(\pi_0, \pi_1)$ (the crossbar output pin utilization when the crossbar inputs have utilizations π_0 and π_1) is p in Corollary 5.2. □

COROLLARY 5.7. *The expected number of active output pins* (m) *conditional on* n *active inputs to a J-stage network is given by*:

$$\mathrm{E}(m \mid n) = 2^J T_J(n). \tag{20}$$

From the theorem and its corollary, the required conditional throughput function $\mu_n = \mu\mathrm{E}(m \mid n)$ *can be used to solve the birth and death process defined in Section* 3.3.

5.7 SATURATED DELTA NETWORK. The delta network becomes saturated when the queue lengths at each of its inputs grows very large. This occurs as the number of customers in the closed system of DMA servers and delta network approaches infinity.

COROLLARY 5.8. *For a saturated delta-*2 *network,* p_s, *the probability that the top pin in stage s is active is given by the following*:

$$p_s = \frac{2}{s+2} \tag{21}$$

and the expected number of active outputs in a J-stage network is given by:

$$\mathrm{E}(m \mid 2^J) = \frac{2^{J+1}}{J+2}. \tag{22}$$

PROOF. Since all input pins to the network are active the recurrence of Section 5.1 simplifies considerably in the absence of splitting probabilities $Q_s(i \mid n)$ since $Q_s(2^{s-1} \mid 2^s) = 1$ (from (17)).

Consider an s-stage delta network with all 2^s inputs active. The $(s - 1)$-stage subnetworks are also saturated (see Figure 5) and so the recurrence (18) in Theorem 5.6 simplifies to give the following:

$$p_s = \frac{2p_{s-1}}{2 + p_{s-1}} \qquad \text{where} \quad p_s = T_s(2^s).$$

This recurrence has solution (21), and (22) follows from Corollary 5.7. □

For a 2-stage, saturated delta-2 network with $\mu = 1$, we find $p_2 = \frac{1}{2}$ using (21). Thus, the throughput of the saturated network is 2. This compares with a figure of $4.4/(4 + 4 - 1) = 2^{2/7}$ given by eq. (16) for a full crossbar network.

Although the recurrence (18) has only been simplified under these rather restricted conditions, these are of course the conditions of greatest interest, namely, high loading. In the following section, we derive similar recurrence formulas for the case when the output pins are nonuniformly selected, which allows us to investigate the effect of hot-spots on performance.

6. *Conditional Throughput Function for Nonuniform Delta Network*

So far we have assumed that all outputs from the MIN are equally utilized, but the use of memory elements, say, on the output side may not be uniform for all elements. For example, if a shared lock or a frequently accessed part of a data structure were residing in a particular memory element, then requests to that memory address would be more frequent than to other destination addresses. We can characterize this by assigning a routing probability to each memory for all requests. Furthermore, each processing element (and so input pin) may have a different memory access pattern, that is, it may have a favored set of memories that it frequently accesses. However, in this section, we only consider the simplest case in which there is a single hot-spot, namely, the top output pin, and all other memories are equally utilized. With more computational effort, more general hot-spot contention can be modeled by a direct extension of the approach presented here.

The basis of the analysis of a nonuniform delta network is the same as that of a uniform one, namely, using Theorem 3.1 to determine the throughput function for the FES given some effective service rate function μ_n. Hence, we derive a recursive formula to determine μ_n for the case where there is a hot-spot at the top output pin. This entails the use of reachability properties for any task at an arbitrary switch in the network attempting to reach a particular output pin. We first show that paths to output pins of the same class (defined in Section 2.3) have the same steady-state probability of being blocked to an arrival, that is, have the same degree of contention. We can then obtain recurrence formulas for the throughput of each output pin class in much the same way as we did for the uniform case in Theorem 5.6. This result is parameterized in terms of unknown pin mean holding-time ratios, and these are finally determined iteratively using the fact that in the network, an output pin's throughput must be proportional to its selection probability (which is the same for all but the top pin).

6.1 Pin Class Reachability and Blocking Probability. Before analyzing hot-spot contention in delta networks, we first prove some reachability properties for tasks in a network in terms of pin classes, that is, to what classes of network output pins can a task connect from a given class of pin within the network. We then consider the probability that a path to an output pin of a given class is blocked.

Lemma 6.1. *For any s-stage delta-2 network with a partial shuffle topology, only class $k + 1$ $(1 \leq k < s)$ output pins are reachable from class k (and only from class k) output pins in the $(s - 1)$-stage subnetworks, and only the class 0 and class 1 pins are reachable from the class 0 (and only from class 0) pins in the subnetworks.*

Proof. The class 0 pins of the $(s - 1)$-stage subnetworks connect to the top switch in stage s, so only class 0 and class 1 pins of the s-stage network are reachable from the class 0 (and only from class 0) pins of the subnetworks.

Now consider the 2^{k-1} class k pins labeled i in the range $2^{k-1} \leq i \leq 2^k - 1$ $(1 \leq k < s)$ in the $(s - 1)$-stage subnetworks. Since switch i in stage s takes its inputs from output pin i in stage $s - 1$ and has outputs labeled $2i$ and $2i + 1$, only pin j in the range $2(2^{k-1}) \leq j \leq 2(2^k - 1) - 1$, that is, $2^k \leq j \leq 2^{k+1} - 1$ is reachable from (and only from) subnetwork pin of class k. In other words, only class $k + 1$ pins in stage s are reachable from class k (and only from class k) pins in the subnetworks, where $1 \leq k < s$. □

Proposition 6.2. *For any s-stage subnetwork $(0 < s \leq J)$ in the recursive definition of a J-stage delta-2 network with a partial shuffle topology, only network output pins up to and including class $J - s$ are reachable from the class 0 (and only from class 0) output pins of the subnetworks. Likewise, only network output pins of class $J - s + k$ $(0 < k \leq s)$ are reachable from class k (and only from class k) output pins of the subnetworks.*

Proof. The proof is by induction on $t = J - s$ for $s = J, J - 1, \ldots, 1$. For the base case $(t = 0)$, the proposition is trivially true.

For the case $t > 0$, let us assume the proposition holds for $J - s = t - 1$. By Lemma 6.1, only stage s output pins of class 0 and 1 can be reached from class 0 (and only from class 0) pins in the $(s - 1)$-stage subnetworks, so only network output pins up to and including class t are reachable from class 0 (and only from class 0) pins of the subnetworks (by the inductive hypothesis).

Also by Lemma 6.1, only stage s output pins of class $j + 1$ can be reached from class j (and only from class j) pins $(1 \leq j < s)$ in the $(s - 1)$-stage subnetworks, so only network output pins of class $t + j$ are reachable from class j (and only from class j) pins of the subnetworks (by the inductive hypothesis).

Hence, the proposition holds for $J - s = t$ and this completes the proof. □

For each interstage link, there is an associated steady-state probability that an arriving task wishing to use that link will be blocked: this is called the *link blocking probability*. Each path contains a set of links across the network and has a similar blocking probability. Two paths have the same blocking probability if all links between corresponding stages have the same blocking probability. Now suppose that when a task arrives at a particular switching node it selects either output pin with equal probability because the routing probabilities to all possible destination addresses from this node are the same; then the link blocking probability for both output links will be identical. By applying this observation to appropriate nodes in the network, we can show that paths to certain output pins have the same blocking probability despite the presence of a hot-spot at the top destination address. This is formalized by the following proposition.

PROPOSITION 6.3. *For an s-stage delta-2 network with a partial shuffle topology in which all output pins but the topmost one are uniformly utilized, the path blocking probability is the same for all paths to pins of the same class (i.e., paths to 2^k class $k + 1$ pins $(0 \leq k < s)$ have the same blocking probability).*

PROOF. The base case of our inductive proof $(s = 1)$ holds trivially because there is only one pin of class 0 and one of class 1. For the multistage case $(s > 1)$, let us assume that the result holds for an $(s - 1)$-stage delta network. Now since both upper and lower subnetworks are identical by symmetry and customers are uniformly distributed over the network input pins, the path blocking probabilities for partial paths to a pin of class k $(1 \leq k \leq s - 1)$ in both subnetworks are the same (by the inductive hypothesis). By Lemma 6.1, we know that pins of class $k > 0$ in the subnetworks connect only to pins of class $k + 1$ in stage s. Since the hot-spot is not reachable from these stage s pins, their link-blocking probabilities are the same. Hence, path-blocking probabilities are the same for all paths to pins of class $k + 1$ $(1 \leq k \leq s - 1$ or $2 \leq k + 1 \leq s)$.

Finally, we note that when $k = 0$, the proposition is satisfied vacuously, which completes the proof. □

When there is a single hot-spot at the top output pin of a delta network, paths to output pins of the same class experience the same degree of contention. If different classes of network output pins are reachable from a given switch in the network, then paths from the upper switch output may have different blocking probabilities to paths from the lower switch output. Consequently, when the upper and lower switch outputs become active, the mean time taken for them to become inactive, their mean holding time, which we will also call their *release time*, will be different. Hence, modeling the presence of a hot-spot differs from the uniform routing case in two important aspects: the different routing probabilities and the different release times for outputs of a given switch.

6.2 HOT-SPOT TRAFFIC MODEL. The hot-spot traffic model is characterized by the hot-spot routing probability ρ that is the probability that a new arrival selects the top network output pin as its destination.

Each of the other network outputs are chosen with probability $(1 - \rho)/(2^J - 1)$ in a J-stage network.

In our analysis of hot-spots, we only need to consider the nodes in the decode tree of the top network input pin because all input pins are identical. Firstly, consider an arrival to the topmost switching node in stage s of a J-stage network. We can determine the probability that the task selects the upper switch output pin by summing the routing probabilities to the network pins that are reachable from the upper pin. It has been shown in Proposition 6.2 that, from the upper output, network pin number 0 is reachable as are all pins up to and including class t, and from the lower output, only pins of class $t + 1$ are reachable, where $t = J - s$. Since there are 2^{k-1} pins of class k, the switch routing probability $\omega(s)$ for the top switch in stage s is given by:

$$\omega(s) = \frac{\rho + \sum_{k=1}^{t} 2^{k-1}q}{\rho + \sum_{k=1}^{t+1} 2^{k-1}q} = \frac{\rho + (2^t - 1)q}{\rho + (2^t - 1)q}, \tag{23}$$

where $q = (1 - \rho)/(2^J - 1)$.

The expression for $\omega(s)$ is applicable to the top row of switches across the network. For the other switches in the decode tree of the top network input, the switch output pins are of identical class and so, by Proposition 6.2, only network output pins of identical class are reachable. Hence, the routing probabilities to the switch outputs are the same and so the value of ρ is a half.

6.3 NONUNIFORM 2-BY-2 CROSSBAR. Since all network inputs are treated identically, we only need to consider a single representative pin (say the topmost input). We base our recursive analysis, to determine the conditional throughput function, on the decode tree of the top input pin. Hence, we require knowledge of the properties of the crossbars in this decode tree only. In particular, the topmost switches in each stage of this decode tree will be nonuniform whereas other switches will be uniform because their output pins have the same class; the hot-spot is not reachable from these switches. As with the uniform routing case, we wish to determine the probability that a particular switch output is active given the utilization of the inputs to the switch. However, in this case, not only do we have to consider the nonuniform routing, but also the fact that when the upper and lower switch outputs become active, their release times are different. In fact, in our analysis of the crossbar, we only need to know the *ratio* of the release times for the upper and lower outputs.

6.3.1 *Definitions.* The following definitions for the nonuniform crossbar are analogous or extensions to those for the uniform crossbar considered in Section 5.1.

—The equilibrium probabilities π_0, π_1 are abbreviations of $\pi(1)$, $\pi(*\ 1)$, respectively (as in Section 5.1).

—The release time ratio r is the ratio of the MHT of the lower output and the MHT of the upper output.

—The blocking probability b_{ij} for input pin i and output pin j ($i, j \in \{0, 1\}$) is the steady-state probability that, at the instant a task arrives on input pin i, the output pin it requires (pin j) is already held by a task currently holding the other input pin.

6.3.2 *A Nonuniform Crossbar in Equilibrium.*

LEMMA 6.4. *For a nonuniform* 2×2 *crossbar with output pins* 0 *and* 1 *that have*

—selection probability ρ *and* $(1 - \rho)$

—MHT m *and* rm, *where* r *is the release time ratio*

—MRHT d_0 *and* d_1

the probability that the upper output is active is given by:

$$p = \rho m\left[\frac{\pi_0}{\rho(m + b_{00}d_0) + (1 - \rho)(rm + b_{01}d_1)} + \frac{\pi_1}{\rho(m + b_{10}d_0) + (1 - \rho)(rm + b_{11}d_1)}\right]$$

and the probability that the lower output is active is given by:

$$q = \frac{(1-\rho)rp}{\rho},$$

where π_i *is the utilization of input pin* i $(i = 0, 1)$.

PROOF. Let the arrival rate on input pin i be λ_i $(i = 0, 1)$. Four applications of Little's result then yield

(i) For input pin 0: $\pi_0 = \lambda_0\,[\rho(m + b_{00}d_0) + (1-\rho)(rm + b_{01}d_1)]$
(ii) For input pin 1: $\pi_1 = \lambda_1\,[\rho(m + b_{10}d_0) + (1-\rho)(rm + b_{11}d_1)]$
(iii) For output pin 0: $p = \rho(\lambda_0 + \lambda_1)m$
(iv) For output pin 1: $q = (1-\rho)(\lambda_0 + \lambda_1)rm$

The result follows by eliminating λ_0 and λ_1. □

COROLLARY 6.5. *Assuming that the nonuniform crossbar has the AOP and that the MRHT = MHT for both output pins,*

$$p = \rho(\rho + (1-\rho)r)\left[\frac{\pi_0}{G(\pi_1)} + \frac{\pi_1}{G(\pi_0)}\right] \quad \textit{and} \quad q = \frac{(1-\rho)rp}{\rho},$$

where $G(\pi_i) = (1-\pi_i)(\rho^2 + (1-\rho)^2r^2) + 2\rho(1-\rho)r$.

PROOF. Conditional on only the lower input being active, let $P(\uparrow)$ be the probability that the task is holding the upper output and let $P(\downarrow)$ be the probability that it is holding the lower output. Applying Little's result to the crossbar at times when only its lower input is active is equivalent to using Lemma 6.4 with $\pi_0 = 0$ and $\pi_1 = 1$ (so that $b_{10} = b_{11} = 0$). This gives:

$$p' = \frac{\rho}{\rho + (1-\rho)r} \quad \text{and} \quad q' = \frac{(1-\rho)r}{\rho + (1-\rho)r},$$

where p' and q' are the utilizations of the upper and lower output pins when only the lower input pin is active. Also we have:

$$p' = \frac{\pi(0\ 1\ 1\ 0)}{\pi(0\ 1)} \quad \text{and} \quad q' = \frac{\pi(0\ 1\ 0\ 1)}{\pi(0\ 1)}$$

and so

$$P(\uparrow) = p' = \frac{\rho}{\rho + (1-\rho)r} \quad \text{and} \quad P(\downarrow) = q' = \frac{(1-\rho)r}{\rho + (1-\rho)r}.$$

Given that the AOP holds and that one input pin is active (by symmetry), a new arrival on the other input sees the active input connected to the upper output with probability $P(\uparrow)$ and to the lower output with probability $P(\downarrow)$. Thus, we have the following blocking probabilities:

$$b_{00} = \pi_1 P(\uparrow) \quad \text{and} \quad b_{01} = \pi_1 P(\downarrow),$$
$$b_{10} = \pi_0 P(\uparrow) \quad \text{and} \quad b_{11} = \pi_0 P(\downarrow).$$

The result follows by substituting these blocking probabilities and $d_0 = m$, $d_1 = rm$ in Lemma 6.4. □

Let $U_k(\pi_0, \pi_1, \rho, r)$ be the utilization of output pin k of a crossbar with input utilizations π_0 and π_1; upper pin routing probability ρ; and release time ratio r. Thus $U_0(\pi_0, \pi_1, \rho, r) = p$ and $U_1(\pi_0, \pi_1, \rho, r) = q$ in Corollary 6.5.

6.4 Recursive Analysis of a Delta Network with a Hot-Spot. In this subsection, we derive the throughput of a delta network with nonuniform routing, conditional on the number of active inputs to the network. Although the approach presented here is applicable to full nonuniform routing in which all output pins are selected with different probabilities, for simplicity, we only consider the problem of a single hot-spot that is selected by a newly arriving customer with probability ρ, and all other output pins are uniformly utilized.

Suppose the top pin is the hot-spot and so $\rho > 1/b$ for a b-way MIN. Then, all nodes in the decode tree of the top output pin will be busier than in the uniformly utilized case, so we refer to it as the hot decode tree. The degree of overlap with another output pin's decode tree will determine the extent of the extra contention caused by the hot-spot. For example, customers destined for pin number one, adjacent to the hot pin, suffer the greatest contention because pin one's decode tree nodes are identical to the hot-pin's decode tree nodes. However, the customers destinated for the bottom output pin suffer less hot-spot contention because its decode tree nodes only overlap the hot decode tree at the leaves, that is, the input pins of the first stage. Consequently, in equilibrium, a greater proportion of the customers will build up for pin number one than for the bottom pin. It is easy to see, by an argument analogous to the one used to prove Proposition 6.3, that certain output pins have the same build up of customers and that these pins can be grouped together to form the classes defined earlier in Section 2.3.

Given the **N** and **Z** defined in Section 5.1, we now define the following probability distributions:

$$T_s^{(0)}(n) = \Pr(Z_0 = 1 \mid \#(\mathbf{N}) = n)$$

and for $k > 0$,

$$T_s^{(k)}(n) = \Pr(Z_{2^{k-1}} = 1 \mid \#(\mathbf{N}) = n).$$

$T_s^{(k)}(n)$ is the probability that an output pin of class k is active in an s-stage network with n active inputs and typically we use pin number 2^{k-1} as the representative pin for all class k pins.

In this recursive analysis, we consider the decode tree of the top input pin of the network. For the base case ($s = 1$) of the recursion, the interconnection is a 2-way crossbar with switch-routing probability for the top switch output given by $\omega(1)$. For the s-stage case ($s > 1$), the upper and lower ($s - 1$)-stage subnetworks are separate so that we can determine utilizations of the inputs to the rightmost crossbars by using the result obtained for the smaller network in the recursion. This gives the recurrence formulas in the following theorem.

Theorem 6.6. *Suppose there are n_J active inputs to a J-stage delta network with a partial shuffle topology. Assuming that every crossbar in the network has the AOP and outputs with MRHT = MHT, the utilization of a class k output pin in stage s is given by*:

For $1 < s \leq J$, $k \in \{0, 1\}$, $1 \leq n \leq n_J$,

$$T_s^{(k)}(n) = \sum_{i=max(0,n-2^{s-1})}^{min(n,2^{s-1})} Q_s(i \mid n) U_k(T_{s-1}^{(0)}(i),\ T_{s-1}^{(0)}(n-i),\ \omega(s),\ r(s, n_J)). \quad (24)$$

For $1 < s \leq J$, $1 < k \leq s$, $1 \leq n \leq n_J$,

$$T_s^{(k)}(n) = \sum_{i=max(0,n-2^{s-1})}^{min(n,2^{s-1})} Q_s(i \mid n) U_0\left(T_{s-1}^{(k-1)}(i),\ T_{s-1}^{(k-1)}(n-i),\ \frac{1}{2},\ 1\right) \quad (25)$$

and for $k \in \{0, 1\}$, we have:

$$\begin{aligned} T_1^{(k)}(0) &= 0, \\ T_1^{(k)}(1) &= U_k(0, 1, \omega(1), r(1, n_J)), \\ T_1^{(k)}(2) &= U_k(1, 1, \omega(1), r(1, n_J)), \end{aligned} \tag{26}$$

where $\omega(s)$ is the switch-routing probability and $r(s, n_J)$ is the release-time ratio for the top switch in stage s.

PROOF. The proof is analogous to that of Theorem 5.6. In addition, it uses Lemma 6.1 for (24) and (25), that is, a pin of class k in stage s is reachable from a pin of class $k - 1$ only. □

COROLLARY 6.7. *The expected number of active output pins (m) conditional on n_J active inputs to a J-stage network is given by:*

$$\mathrm{E}(m \mid n_J) = T_J^{(0)}(n_J) + (2^J - 1)T_J^{(1)}(n_J). \tag{27}$$

From the theorem and its corollary with $n_J = 1, 2, \ldots, 2^J$, the required conditional throughput function $\mu_n = \mu \mathrm{E}(m \mid n)$ can be used to solve the birth and death process defined in Section 3.3. This gives $T(N)$, the throughput of the closed system for a given population N. However, the release time ratios $r(s, n_J)$ are unknown in Theorem 6.6. It now remains to obtain these ratios and this is the subject of the next section.

6.5 FIXED-POINT EQUATION FOR RELEASE TIME RATIOS. First we define the following notation:

- Suppose there are n_J active inputs to a J-stage delta network, so that the release-time ratios $r(s, n_J)$ can be abbreviated to r_s for the topmost crossbar in stage s ($s = 1, 2, \ldots, J$), where $r_s > 0$.
- Let $\mathbf{t}$ describe the utilizations of the network output pins, that is, t_k is the utilization of a class k pin ($k = 0, 1, \ldots, J$). From Theorem 6.6, we have $t_k = T_J^{(k)}(n_J)$.
- Let ρ ($0 \le \rho < 1$) be the hot-spot routing probability that describes the network traffic and ρ_k be the routing probability to a class k output pin, so that

$$\rho_0 = \rho \qquad \text{and} \qquad \rho_k = \frac{1 - \rho}{2^J - 1} \qquad k = 1, 2, \ldots, J.$$

- Let $\omega(s)$, which we abbreviate to ω_s, be the corresponding top switch-routing probability function, where $0 \le \omega_s < 1$ ($s = 1, 2, \ldots, J$).
- Let $\mathbf{r} = (r_1, r_2, \ldots, r_J)$, $\mathbf{t} = (t_0, t_1, \ldots, t_J)$, $\boldsymbol{\rho} = (\rho_0, \rho_1, \ldots, \rho_J)$ and $\boldsymbol{\omega} = (\omega_1, \omega_2, \ldots, \omega_J)$.

We now derive a fixed-point equation for the pair $(\boldsymbol{\omega}, \mathbf{r})$ using the functions A, B, and C, defined below. Each function takes a pair of vectors and returns a pair of vectors in which the second component $(\mathbf{r})$ is the same in both pairs. However, in function A, the $\mathbf{r}$ influences the value of the first vector in the result.

- First, from Theorem 6.6, the utilization vector is a function of the switch-routing probabilities and the release-time ratios. Suppose F is this function and define $(\mathbf{t}, \mathbf{r}) = A(\boldsymbol{\omega}, \mathbf{r}) = (F(\boldsymbol{\omega}, \mathbf{r}), \mathbf{r})$.
- Now $\mathbf{t}$ induces a set of network-routing probabilities since the proportion of the total throughput flowing out of a pin in equilibrium must be equal to its selection

probability, that is,

$$\rho_k = \frac{t_k}{t_0 + \sum_{j=1}^{J} 2^{j-1} t_j} = [G(\mathbf{t})]_k, \qquad \text{say.}$$

Now let $(\boldsymbol{\rho}, \mathbf{r}) = B(\mathbf{t}, \mathbf{r}) = (G(\mathbf{t}), \mathbf{r})$.

—The switch-routing probability is a function of the network-routing probability, and given by a more general form of (23):

$$\omega_s = \frac{\rho_0 + \sum_{k=1}^{J-s} 2^{k-1} \rho_k}{\rho_0 + \sum_{k=1}^{J-s+1} 2^{k-1} \rho_k} = [H(\boldsymbol{\rho})]_s, \qquad \text{say.}$$

Now let $(\boldsymbol{\omega}, \mathbf{r}) = C(\boldsymbol{\rho}, \mathbf{r}) = (H(\boldsymbol{\rho}), \mathbf{r})$.

Thus, we have the following fixed-point equation: $(\boldsymbol{\omega}, \mathbf{r}) = C(B(A(\boldsymbol{\omega}, \mathbf{r})))$. Although for all $(\boldsymbol{\omega}, \mathbf{r})$, $C(B(A(\boldsymbol{\omega}, \mathbf{r}))) = (\boldsymbol{\omega}', \mathbf{r})$ for some $\boldsymbol{\omega}'$, the choice of $\mathbf{r}$ determines $\boldsymbol{\omega}'$ and hence the fixed-point pair.

We require fixed-points $(\boldsymbol{\omega}, \mathbf{r})$ of the above equation. Typically, $\boldsymbol{\omega}$ is given by the traffic pattern and a guess is required for $\mathbf{r}$. From the analysis, we know that $r_J = 1$ because complete paths suffer no further contention and DMA transfer rates are the same regardless of the output pin. The proof of uniqueness of $\mathbf{r}$ with respect to the fixed-point equation is not given here, but we conjecture that $\mathbf{r}$ is unique because in the 2-stage network different release-time ratios in the first stage crossbar give output pin utilizations that induce different switch-routing probabilities. This means that both cannot satisfy the fixed-point equation.

6.6 ITERATIVE METHOD. Let $r_s^{(i)}$ be the approximation for r_s after the ith iteration. As an initial guess, we choose the following: $r_s^{(0)} = 1$ for $s = 1, 2, \ldots, J$. This guess is exact when $n_J = 1$ and also for uniform traffic. Furthermore, in all approximations to $\mathbf{r}$, we know that $r_J = 1$.

The induced switch-routing probabilities before the ith iteration are given by the following:

$$\boldsymbol{\omega}^{(i)} = \text{first}(C(B(A(\boldsymbol{\omega}, \mathbf{r}^{(i)})))) \qquad \text{where} \quad \text{first}((p, q)) = p.$$

The difference between the required and the observed switch-routing probabilities can be quantified as follows:

$$d_s^{(i)} = \frac{\omega_s^{(i)} - \omega_s}{\omega_s}.$$

Now if $d_s^{(i)}$ is positive, then the utilization of the upper pin is higher than expected and so the release-time ratio for the top switch in stage s needs to be increased. Thus, we use the following to update the value of $\mathbf{r}$:

$$r_s^{(i+1)} = r_s^{(i)}[1 + Dd_s^{(i)}],$$

where D (typically between 1 and 4) is a damping factor used to reduce the number of iterations. The stopping condition is $|d_s^{(i)}| < \epsilon$, $s = 1, 2, \ldots, J$ for some arbitrarily small ϵ.

7. *Validation and Numerical Results*

7.1 VALIDATION OF ANALYTICAL MODEL. The model is validated against a simulation model of the closed system of parallel DMA servers (Section 3) connected to a circuit-switched delta network in which partial paths are held. Once a complete path is established, it is held for a random-time interval that has a negative exponential-time interval with unit mean.

Four types of traffic models are considered:

—Saturated network with uniform traffic,
—Saturated network with a hot-spot,
—Network with small population and uniform traffic,
—Network with small population and a hot-spot.

The hot-spot model considered is one in which the hot pin has routing probability twice that of a cool pin.

The throughput results of the validation are tabulated in Tables I, II, III, and IV. As can be seen from the first two tables, the analytical model is very accurate (less than 1% error) when the network is saturated. The relative errors are noticeably higher (up to 2.9%) in the case of the nonsaturated network because of the additional assumptions required in modeling the closed system by the birth and death process (Section 3.3), that is, that all arrangements of customers on a given number of active inputs are equally likely.

The confidence intervals were estimated from the simulation results by using the batch means approach. Each sample run modeled 5000 units of time and five samples were taken, giving a total simulation time of 25,000 units. For large networks, the whole simulation run took over an hour to complete on a SUN 3 workstation. Notice that in the saturated, uniform network with two stages, the simulator's 95% confidence limits do include the exact result of 1.9993 (to four decimal places), but that the analytical model provides the more accurate estimate.

7.2 NUMERICAL RESULTS. The graph of throughput against the number of customers, N, for a closed system compares three types of 16-way interconnections.

—The full 16-way crossbar with uniform routing (Section 4)
—The 4-stage delta-2 network with uniform routing (Section 5)
—The 4-stage delta-2 network with hot-spot routing probability 0.2 (Section 6)

As shown in Figure 6, the full crossbar gives the highest throughput and approaches its asymptote closely only when N is large (greater than 100). The uniform delta-2 network becomes saturated much faster (when N is about 100) because tasks face path conflicts on top of memory conflicts. In the presence of a hot-spot, this network gives even lower throughput and becomes saturated even faster as the hot pin's decode tree saturates causing more path conflicts. The uniform interconnections only show similar throughput when the population is very small (less than 5). This indicates that even at low loads the blocking caused by path conflicts significantly reduces the throughput of a delta network.

Figure 7 shows the effect of hot-spot contention in a 16-way, 4-stage delta-2 network connected to a bank of DMA servers in a closed system with various populations, N. The graph of throughput against the routing probability displays the classic hot-spot effect for populations $N = 8$, 16 and the saturated case in which all input pins are always active. This phenomenon is caused by the additional internal contention placed by the hot-spot on other decode trees. For the saturated case, as ρ increases the throughput reaches its peak when all the output pins are uniformly utilized ($\rho = \frac{1}{16}$), drops sharply as more of the traffic is routed to the hot-spot, and then follows the curve $1/\rho$ very closely, ultimately giving the throughput of a serial link, when $\rho = 1$. In fact, we can easily show that the $1/\rho$ curve provides an upper bound for *all* protocols as follows:

TABLE I. SATURATED, UNIFORM DELTA NETWORK

No. of stages	Analytical model	Simulation model	95% Confidence interval	Relative error (%)
2	2.000	1.992	(1.952, 2.032)	0.40
3	3.200	3.185	(3.143, 3.228)	0.47
4	5.333	5.375	(5.313, 5.437)	−0.78
5	9.143	9.163	(9.101, 9.225)	−0.22
6	16.00	15.97	(15.85, 16.08)	0.19

TABLE II. SATURATED, NONUNIFORM DELTA NETWORK

No. of stages	Hot-spot probability	Analytical model	Simulation model	95% Confidencc interval	Relative error (%)
2	0.400000	1.896	1.892	(1.866, 1.917)	0.21
3	0.222222	3.055	3.057	(3.017, 3.097)	−0.07
4	0.117647	5.174	5.193	(5.115, 5.271)	−0.37
5	0.060606	8.996	8.989	(8.898, 9.079)	0.08
6	0.030769	15.88	15.84	(15.71, 15.97)	0.25

TABLE III. NONSATURATED, UNIFORM DELTA NETWORK

No. of stages	Network population	Analytical model	Simulation model	95% Confidence interval	Relative error (%)
2	4	1.612	1.644	(1.603, 1.685)	−1.9
3	8	2.548	2.543	(2.498, 2.567)	0.20
4	16	4.283	4.227	(4.172, 4,283)	1.3
5	32	7.460	7.248	(7.198, 7.299)	2.9
6	64	13.28	12.98	(12.89, 13.08)	2.3

TABLE IV. NONSATURATED, NONUNIFORM DELTA NETWORK

No. of stages	Network population	Hot-spot probability	Analytical model	Simulation model	95% Confidence interval	Relative error (%)
2	4	0.400000	1.564	1.579	(1.559, 1.598)	−0.95
3	8	0.222222	2.479	2.485	(2.440, 2.531)	−0.24
4	16	0.117647	4.206	4.174	(4.104, 4.244)	0.77
5	32	0.060606	7.385	7.216	(7.139, 7.293)	2.3
6	64	0.030769	13.21	12.88	(12.77, 12.99)	2.6

Suppose the routing probability to network output pin i is ρ_i and the probability that pin i is active (i.e., pin i's throughput when $\mu = 1$) is t_i. Now, since ρ_i of the total throughput, $T(N)$ is routed to output pin i, we have $t_i = \rho_i T(N)$, which implies $T(N) \leq 1/\rho_i$ for all i, that is, $T(N) \leq 1/\max(\rho_i)$. Hence, if ρ is the hot-spot routing probability $T(N) \leq 1/\rho$. When ρ is large, the throughput curves tend towards the curve $1/\rho$, particularly for large N, because the hot output pin is almost always active. These curves relate to a circuit-switched network in which partial paths are held. When partial paths are released and the server retries immediately, throughput increases and so would lie somewhere between the curve $1/\rho$ and the corresponding curve shown in Figure 7; $1/\rho$ is still an upper bound by the same reasoning. In this idealized case, when there is uniform routing ($\rho = \frac{1}{16}$), the throughput is 16μ because all output pins are busy, so the throughput would drop considerably faster if ρ increased slightly, thereby giving a more dramatic hot-spot

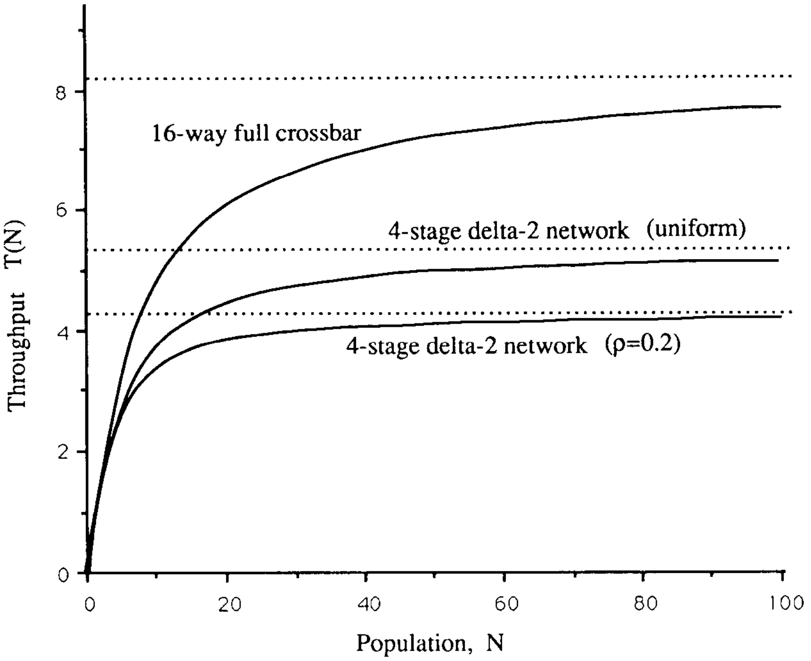

FIG. 6. Comparison of performance of 16-way networks.

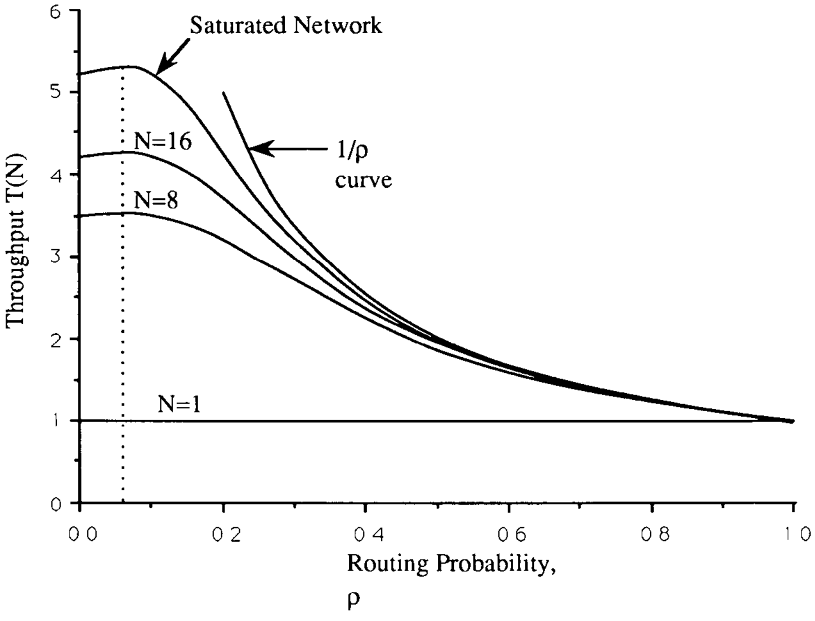

FIG. 7. Graph of throughput vs. hot-spot routing probability (μ = 100).

effect. Similarly, packet-switched networks would give an even more drastic reduction in throughput for a slight increase in traffic to a particular output.

For all of these protocols, in larger networks and with large *N*, the hot-spot effect is more pronounced because the throughput is greater in the uniformly utilized case and the throughput curve closely follows the $1/\rho$ curve as ρ increases past about 0.2. However, it may be unlikely that very high hot-spot routing probabilities are encountered in practice, especially when there are a large number of output pins.

8. *Conclusion*

The work described in this paper extends the flow-equivalent server aggregation technique to incorporate passive resources that represent switching networks of various types. In this way, the modeling of large-scale parallel computer architectures is greatly simplified, and we have presented results predicting the throughput of circuit-switched networks in which partial paths are held by blocked transactions during path building. We can therefore see the degradation in performance suffered by a MIN compared with the equivalent full crossbar offering the same connectivity (if it were possible to fabricate such a device). We also derived for the first time by analytical methods, we believe, results showing the effect of "hot-spots" in asynchronous circuit-switched networks, where one destination address is more frequently selected than the others and causes performance to suffer, ultimately giving the throughput achieved by a single serial link. Previously, this effect has only been shown by simulation, for example, [25].

The circuit-switching communication protocol considered is an important one, being the simplest to fabricate and the one to be used in a new generation of optical switches [15], for example. However, there are a number of other protocols that should be considered. The simplest of these is infinitely buffered packet-switching that has already been studied as we noted in the Introduction, but many hybrids are also possible; perhaps the simplest being packet-switching with limited buffering and hence blocking. In addition, a model should be developed in which switching times are not neglected.

Certainly, for contemporary electronic crossbars and even quite small message lengths, switching time is negligible compared with data transmission time. But for the optical devices referred to the converse holds for the present—unless whole files of data are normally transmitted as single messages. In such a model, the path-building process cannot be assumed instantaneous.

The work presented in this paper includes the modeling of hot-spots parameterized by the probability that an arriving task selects the destination address of the hot-spot. One relatively simple extension to the model presented here is to allow full nonuniform routing so that the effect of more than one hot-spot and the effect of proximity of hot-spots can be studied quantitatively. A further extension would be to model systems in which certain output addresses are favored, depending on the input pin on which a task arrives. Here, of course, the arrival process would no longer be the same at all inputs.

Finally, our analysis should also be adapted to represent circuit switching in which transactions that are blocked during path-building *release* their partial paths, effectively being "lost" and having to retry after some (randomly distributed) delay. As indicated in Section 7, for this protocol, the hot-spot effect on throughput appears to be more dramatic than when partial paths are held, especially for fast retry rates. Thus, it is important to model this retry protocol and a pilot study that uses fixed-point methods may be found in [7]. This analysis would then be close to the classical modeling of telephone networks. We would also expect packet-switched MINs to show a more pronounced hot-spot effect because when there is no hot-spot they achieve higher throughput. This would then allow comparison of the effectiveness of protocols in a given MIN with respect to the extent of hot-spot contention.

REFERENCES

(Note: Reference [28] is not cited in the text.)

1. AKYILDIZ, I. F. General closed queueing networks with blocking. In *Proceedings of the 12th Annual International Symposium on Computer Performance Modelling* (Brussels, Belgium, Dec.). North-Holland, Amsterdam, The Netherlands, 1987, pp. 283–303.
2. BASKETT, F., CHANDY, K. M., MUNTZ, R. R., AND PALACIOS, F. G. Open, closed and mixed networks of queues with different classes of customers. *J. ACM 22*, 2 (1975).
3. BHUYAN, L. N., AND AGRAWAL, D. P. Design and performance of generalised interconnection networks. *IEEE Trans. Comput. C-32*, 12 (1983), 1081–1090.
4. CHANDY, K. M., HERZOG, U., AND WOO, L. Parametric analysis of queueing networks. *IBM J. Res. Develop.* (Jan. 1975), 36–49.
5. CINLAR, E. *Introduction to Stochastic Processes.* Prentice-Hall, Englewood Cliffs, N.J., 1975.
6. COPE, G. A. The modelling of circuit-switched multi-stage interconnection networks. MSc dissertation. Imperial College, London, England, 1987.
7. COURTOIS, P. J. *Decomposability.* Academic Press, Orlando, Fla., 1977.
8. DARLINGTON, J., AND REEVE, M. J. ALICE: A multiprocessor reduction machine for the parallel evaluation of applicative languages. In *ACM/MIT Conference on Functional Programming Languages and Computer Architecture,* 1981.
9. GARG, U., AND HUANG, Y. P. Decomposing banyan networks for performance analysis. *IEEE Trans. Comput. C-37,* 3 (1988), 371–376.
10. GOKE, L. R., AND LIPOVSKI, G. L. Banyan networks for partitioning multiprocessor systems. In *Proceedings of the 1st Annual Symposium on Computer Architecture.* 1973, pp. 21–28.
11. HARRISON, P. G. Queueing models of the large-scale parallel computer architectures. In *ORSA/TIMS Workshop on Queueing Networks and Their Applications.* New Jersey, Jan. 1987.
12. HARRISON, P. G., AND FIELD, A. J. Performance modelling of parallel computer architectures. In *Proceedings of Performance '86 and ACM Sigmetrics 1986* (Raleigh, N.C., May 27–30). ACM, New York, 1986, pp. 18–27.
13. HARRISON, P. G., AND REEVE, M. J. The parallel graph reduction machine, ALICE. In *Proceedings of the Workshop on Graph Reduction* (Santa Fe, N.M., Sept.). 1986, also published in *Lecture Notes in Computer Science,* No. 279, Springer-Verlag, New York, 1987.
14. JUMP, J. R., AND DIAS, D. M. Analysis and simulation of buffered delta networks. *IEEE Trans. Comput. C-29,* 9 (Sept. 1980), 791–801.
15. DTI JOERS (Joint Opto-Electronic Research Scheme) proposal, Issue 2. Digital Optics: Application to Computing and Communications, March 1986.
16. KELLY, F. P. Blocking probabilities in large circuit-switched networks. *Adv. Appl. Prob. 18* (1986), 473–505.
17. KNUTH, D. *Fundamental Algorithms,* vol. 1. Addison-Wesley, Reading, Mass., 1968.
18. KUEHN, P. J. Approximate analysis of general queueing networks by decomposition. *IEEE Trans. Comm. COM-27,* 1 (1979).
19. KUMAR, M., AND PFISTER, G. F. The onset of hot spot contention. In *Proceedings of the 1986 International Conference on Parallel Processing* (Aug.). 1986, pp. 28–34.
20. LITTLE, J. D. C. A proof of the queueing formula $L = \lambda W$. *Op. Res. 9,* 3, 383–387.
21. LIU, Y. S. The delta network performance and the hot spot traffic. Ultracomputer Project Technical Report. Courant Institute, New York University, New York, 1987.
22. MITRA, D., AND CIESLAK, R. Randomized parallel communications on an extension of the omega network. *J. ACM 34,* 4 (Oct. 1987), 802–824.
23. PATEL, J. H. Performance of processor-memory interconnections for multiprocessors. *IEEE Trans. Comput.* (Oct. 1981), 771–780.
24. PATEL, N. M. *Models of circuit-switched interconnection networks.* PhD dissertation, Imperial College, University of London, London, England, August 1989.
25. PFISTER, G. F., AND NORTON, V. A. Hot spot contention and combining in multistage interconnection networks. *IEEE Trans. Comput. C-34,* 10 (Oct. 1985), 943–948.
26. POMBORTSIS, A., AND HALATSIS, C. Performance of crossbar interconnection networks in presence of "Hot Spots". *Electr. Lett. 24,* 3 (1988), 182–184.

27. SEVCIK, K. C., AND MITRANI, I. The distribution of queuing network states at input and output instants. *J. ACM 28*, 2 (Apr. 1981), 358–371.

28. SIEGEL, H. J. *Interconnection Networks for Large-Scale Parallel Processing: Theory and Case Studies.* Lexington Books, 1985.

RECEIVED JULY 1988; REVISED FEBRUARY AND OCTOBER 1989; ACCEPTED NOVEMBER 1989