Performance Modeling of Multihop Network Subject to Uniform and Nonuniform Geometric Traffic

Eric Noel and K. Wendy Tang, *Member, IEEE*

***Abstract*—Performance modeling under nonuniform traffic is a useful tool to validate simulation accuracy and lend insights to realistic implementation of multihop networks. We present memoryless and independence assumptions based performance models capable of tracking nonuniform traffic for an arbitrary multihop network with the deflection and store-and-forward routing strategies. We also include a description of the efficient numerical algorithms, and provide comparisons to simulation. Our models are a generalization of Greenberg–Goodman, Greenberg–Hajek, Hajek–Krishna, and Brassil–Cruz models.**

***Index Terms*—Deflection routing, multihop networks, performance modeling, store-and-forward routing.**

I. Introduction

MULTIHOP networks have found applications as wavelength-division multiplexed (WDM) lightwave networks [43], [38] and as interconnection networks for multicomputers [48]. In the former, multihop networks are used as logical topologies for wavelength assignments of transmitters and receivers at a node; whereas in the latter, multihop networks are used as physical topologies for the interconnection of multiple processors in a parallel computer system. In both cases, the number of neighbors at a node is small, and a typical message must go through a number of hops to reach its destination, hence the name *multihop networks*. Large numbers of multihop networks have been studied, including the Manhattan Street [38], the ShuffleNet [53], BanyanNet [54], Toroidal Mesh, and Diagonal Mesh [55]. In the following we model two routing strategies for multihop networks: *deflection routing* and *store-and-forward routing*.

Because of its simplicity, *deflection routing* or *hot-potato routing* [5] is a popular routing strategy among multihop networks. It is a *bufferless, dynamic* routing algorithm. Basically, messages are sorted according to a *deflection criterion*, such as *age* or *path length*. Those with higher priorities are routed optimally to the shortest path while those with lower priorities are *deflected* to nonoptimal links that will lead to a longer path length. There is no buffer and hence no buffer management at a node. Performance studies indicate that age-priority-based deflection-routing algorithms reduce the maximum delay [23], [55] when compared to other deflection-routing algorithms.

Contrary to deflection routing, with *store-and-forward routing* [52] deflected packets are temporarily stored in buffers, so all packets are optimally routed over the shortest path. The *store-and-forward routing* strategy has been applied to packet networks [52] and interconnection networks for multiprocessors [22].

Greenberg–Goodman [21], [20], Greenberg–Hajek [19], and Krishna–Hajek [32] have developed a performance model for packet arrivals subject to the *independence and memoryless assumptions* for uniform traffic and applied to deflection routing in (respectively) Manhattan Street Networks, Hypercube networks, and ShuffleNet Networks. Brassil–Cruz [8], [9] have extended this model to nonuniform traffic in Manhattan Street Networks. We generalize this deflection model for arbitrary network topologies, with or without buffers, and with an improved computational efficiency. Moreover, we extend this model for the store-and-forward routing strategy.

We focus on performance models which consist of state equations linking a node's input parameters to output parameters. When memoryless and independence assumptions apply, models for symmetric networks and traffic reduce to the analysis of a single node or a single buffer. Otherwise, additional state equations linking neighboring nodes are considered. In either case, the state equations are solved iteratively until convergence is reached. From the input and output parameters, steady-state performance parameters such as delay, throughput, and blocking are derived. Models belonging to this category may be found in [26], [30], [44] for multiprocessor systems, and [20], [8], [9], [21] for lightwave networks.

We generalize Greenberg–Goodman, Greenberg–Hajek, Hajek–Krishna, and Brassil–Cruz independence and memoryless assumptions based models for arbitrary network topologies, with or without buffers, for both deflection and store-and-forward routing strategies, and with an improved computational efficiency.

We concede that our packet arrival model (limited by the independence and memoryless assumptions) does not represent bursty arrivals which is more common in communication networks [17]. Moreover, because simultaneous arrivals in buffers result in bursty traffic, we expect our store-and-forward model to degrade with increasing traffic load, as was noted in [57].

This article is composed of the following sections: Section II, where we present our network model; Section III, where we derive the models for *deflection routing* with and without input buffers (extended version of [46]); Section IV, where we derive the models for *store-and-forward routing* with finite and infinite

Manuscript received June 29, 1999; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor H. J. Chao. This work was supported in part by the National Science Foundation under Grant ECS9626655 and Grant ECS9700313.

E. Noel is with AT&T Network Computing and Services, Middletown, NJ 07748 USA (e-mail: ericn@hello.att.com).

K. W. Tang is with the Department of Electrical and Computer Engineering, State University of New York, Stony Brook, NY 11794 USA (e-mail: wtang@ece.sunysb.edu).

Publisher Item Identifier S 1063-6692(00)10924-0.

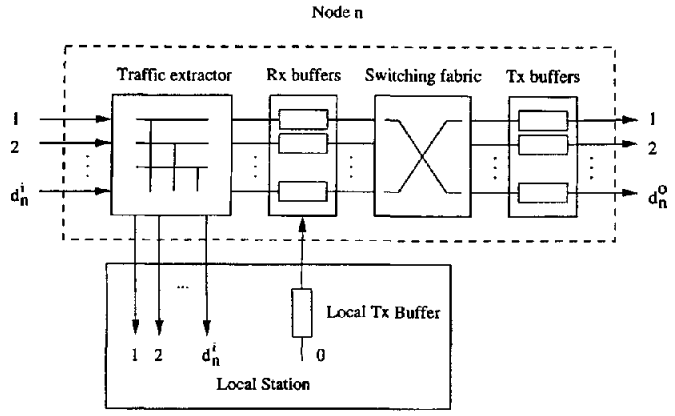


Fig. 1. Node model.

buffers; and Section V, where we compare our models to simulation.

II. Network Model

For each model, a network consists of a set of nodes connected by zero delay links. As shown in Fig. 1, each node consists of a traffic extractor, d_n^i receive buffers (Rx buffers, tagged 1 to d_n^i), a switching fabric, and d_n^o transmit buffers (Tx buffers, tagged 1 to d_n^o). The traffic extractor diverts transit packets which arrive at destination to the local station, so these packets never occupy the receive buffers. The switching fabric maps packets from receive buffers to transmit buffers. Depending on the model, buffer size ranges from 1 to infinity. Time is slotted and synchronized so that all nodes receive and transmit packets simultaneously. Each time slot is decomposed in two phases: a switching phase (packet switched from Rx buffers to Tx buffers), and a transmit phase (packets sent from Tx buffers to Rx buffers).

Attached to each node n is a local station which can accept up to d_n^i packets in the same time slot. Packet generation follows a Bernoulli process defined by the probabilities the local station creates a packet to destination nodes in the next time slot (geometric inter-departure times). Transit packets (packets forwarded by neighboring nodes) have priority over local packets (packets created by the local station). So, a local packet can enter node n Rx buffer only when strictly less than d_n^i transit packets enter node n; otherwise the local packet is blocked for bufferless models, or queued (in the local Tx buffer tagged 0) for the other models.

III. Deflection Model

A. Routing Algorithm

The switching fabric associates every received packet with a set of preferred outgoing links based on the shortest path [13] to the desired destination. The set of preferred outgoing links is empty when all outgoing links result in the same path length. The rule used by the switching fabric to map packets from receive to transmit buffers is age-priority based (the age of a packet corresponds to the number of time slots it has been circulated): first choice is given to the oldest packet with a nonempty set of preferred outgoing links. (*Twin packets*, or packets with equal age, are randomly sorted.) When a packet set of preferred outgoing links overlaps with one or more younger packets sets of preferred outgoing links, the *contention resolution algorithm* described below is invoked. Otherwise, an outgoing link is randomly selected out of the set of preferred outgoing links and the packet is switched to the corresponding transmit buffer. Once all packets with nonempty set of preferred outgoing links have been serviced, packets with an empty set of preferred outgoing links are randomly assigned an outgoing link from the unselected outgoing links.

The *contention resolution algorithm* is applied every time the packet being mapped from receive to transmit buffers (also called the contending packet) has its set of preferred outgoing links overlapping with one or more younger packet's preferred outgoing links. The algorithm consists of first creating a deflection set composed of outgoing links preferred by the contending packet and by the least number of younger packet(s). Then, an outgoing link is randomly selected out of the deflection set and assigned to the contending packet.

B. Steady-State Probabilities

For expository convenience, we have included in Table I the nomenclature of parameters used here. For any node n, we consider packet x, also denoted by $\{d_x, a_x\}$, of destination node d_x and age a_x. To calculate the probability $p^o_{n,l_x}(d_x; a_x+1)$ that packet $\{d_x, a_x\}$ *leaves* node n on link l_x in the next time slot, we first evaluate the probability that a packet destined to node d_x of age a_x exits on link l_x in the next time slot, conditioned by the event that k packets are present in node n receive buffers (represented by E_k). Then, applying the total probability theorem we obtain

$$p^o_{n,l_x}(d_x; a_x+1) = \sum_{k=1}^{d_n^i} \sum_{\mathcal{A}_k, \mathcal{D}_k, \mathcal{L}_k} \Pr\left[\begin{array}{c}\text{Packet } \{d_x, a_x\} \\ \text{exits on link } l_x \\ \text{in next time slot}\end{array} \;\middle|\; E_k\right] p_n(E_k) \tag{1}$$

where $\mathcal{A}_k = \{a_1, a_2, \ldots, a_x, \ldots, a_k\}$ represents all packet age combinations, $\mathcal{D}_k = \{d_1, d_2, \ldots, d_x, \ldots, d_k\}$ represents all packet destination combinations, and $\mathcal{L}_k = \{l_1, l_2, \ldots, l_x, \ldots, l_k\}$ represents all packet incoming link combinations.

We compute the conditional probability in (1) by constructing the recursive function $S^x(j|\mathcal{P}_j, \ldots, \mathcal{P}_x, \ldots, \mathcal{P}_k)$ for $j = 1, 2, \ldots, x$. Qualitatively, for $j < x$, the function computes the product of the probabilities that packets older than packet $\{d_x, a_x\}$ (packets indexed j to $x-1$) are not assigned outgoing link l_x, multiplied by the probability packet x is assigned outgoing link l_x.

More precisely, let node n receive buffers be occupied by k packets $\{d_1, a_1\}, \{d_2, a_2\}, \ldots, \{d_x, a_x\}, \ldots, \{d_k, a_k\}$ with respective set of preferred outgoing links $\mathcal{P}_1, \mathcal{P}_2, \ldots, \mathcal{P}_k$ and such that the age of these packets are sorted with $\{d_1, a_1\}$ being the oldest packet, i.e., $a_1 > a_2 > \cdots > a_x > \cdots > a_k$. Also, define for each packet j, D_j as the deflection set (set of outgoing links

TABLE I
NOMENCLATURE OF PARAMETERS USED IN THE DEFLECTION-ROUTING MODEL FOR STEADY-STATE PROBABILITIES

N	Total number of nodes.
$\{d_j, a_j\}$	j^{th} packet in node n receive buffer, of destination node d_j and age a_j.
$p^i_{n,l}(d;a)$	Probability packet $\{d,a\}$ *arrives* to node n on link l in the next time slot. ($p^i_{n,0}(d;a)$ is the probability node n local packet destined to node d has been queued a time slots before entering an Rx buffer in next time slot.)
$p_n(E_k)$	Probability that only $\{d_1,a_1\},\{d_2,a_2\}$, ..., $\{d_k,a_k\}$ entered node n Rx buffers.
$p_n(S_{k_0})$	Probability node n creates k_0 packets in the next time slot. ($k_0 \in \{0,1\}$, $p_n(S_{k_0=1}) = \sum_{d=0}^{N-1} p^i_{n,0}(d;0)$).
$p_{n,0}(B_s)$	Probability that node n has s packets *queued* in its local buffer.
$p^s_{n,0}(d;a;j)$	Probability that packet $\{d,a\}$ is *queued* at position j of size s local buffer in node n. ($p^s_{n,0}(d;a;1) = p^i_{n,0}(d;a)$)
$p^o_{n,l}(d;a+1)$	Probability that packet $\{d,a\}$ *leaves* node n on link l in the next time slot. ($p^o_{n,l}(n;a) = 0$.)
A	Age bound. $p^o_{n,l}(d;a) = p^i_{n,l}(d;a) = 0$ for $a \geq A$.
$\mathcal{A}_k$	All $(k-1)$ sets spanning over $\{0,1,...,A-1\}$. If all receive buffers are occupied, or the test packet $\{d_x,a_x\}$ has age 0, then $\mathcal{A}_k$ spans over $\{1,...,A-1\}$.
$\mathcal{D}_k$	All $(k-1)$ sets spanning over $\{0,1,...,N-1\}$, excluding node n.
$\mathcal{L}_k$	All k-subsets of the d^i_n-set $\{1,2,...,d^i_n\}$.
$\mathcal{P}_j$	Packet $\{d_j,a_j\}$ set of preferred outgoing links.
D_j	Packet $\{d_j,a_j\}$ set of preferred outgoing links not selected by older packets, and preferred by the least number of younger packets. Referred as packet $\{d_j,a_j\}$ deflection set.
C_j	Packet $\{d_j,a_j\}$ set of preferred outgoing links not selected by older packets, and not preferred by younger packets.
$S^x(j\|\mathcal{P}_j,...,\mathcal{P}_k)$	Product of the probabilities packets $\{a_j,d_j\}$, ..., $\{a_k,d_k\}$ do not select outgoing link l_x, times the probability test packet $\{a_x,d_x\}$ selects outgoing link l_x, when $a_j > a_{j+1} ... > a_x > ... > a_k$.

also preferred by younger packets), and C_j as the set of outgoing links preferred by packet j but not any younger packets. Assuming packet arrivals to the same node are independent of one another and of the state of neighboring nodes (independence and memoryless assumptions), for $j = 1, 2, \ldots, x-1$ we define the function $S^x(j|\mathcal{P}_j, \ldots, \mathcal{P}_x, \ldots, \mathcal{P}_k)$ to be equal to

$$\begin{cases} S^x(j+1|\mathcal{P}_{j+1}, \ldots) & \text{if } (\mathcal{P}_j = \varnothing) \\ & \quad \text{or } (C_j \neq \varnothing \wedge l_x \notin C_j), \\ (1-1/|C_j|) & \\ \quad S^x(j+1|\mathcal{P}_{j+1}, \ldots) & \text{if } l_x \in C_j, \\ \sum_{t=1}^{|D_j|-1} (1-1/|D_j|) & \\ \quad S^x(j+1|\mathcal{P}^t_{j+1}, \ldots) & \text{if } C_j = \varnothing \wedge l_x \in D_j, \\ \sum_{t=1}^{|D_j|} 1/|D_j| & \\ \quad S^x(j+1|\mathcal{P}^t_{j+1}, \ldots) & \text{if } C_j = \varnothing \wedge l_x \notin D_j. \end{cases} \tag{2}$$

That is, if packet $\{d_j, a_j\}$ has an empty set of preferred outgoing links (case $\mathcal{P}_j = \varnothing$), we set to one its probability of not being assigned link l_x prior to packet $\{d_x, a_x\}$ being assigned a link (packets with empty set of preferred outgoing links are assigned an outgoing link last). Note that if packet $\{d_x, a_x\}$ also has an empty set of preferred outgoing links, in effect, we defer calculating the probability that packet $\{d_j, a_j\}$ is not assigned link l_x until we calculate the probability that packet $\{d_x, a_x\}$ is assigned link l_x (see below, case $j = x$). When C_j is not empty and does not contain l_x (case $C_j \neq \varnothing \wedge l_x \notin C_j$), the probability packet $\{d_j, a_j\}$ is not assigned link l_x is also one. If C_j contains l_x (case $l_x \in C_j$), the probability packet $\{d_j, a_j\}$ is not assigned l_x is one minus the probability to randomly choose link l_x in C_j. When C_j is empty, we follow the contention resolution algorithm and construct the deflection set D_j. If D_j contains l_x (case $C_j = \varnothing \wedge l_x \in D_j$), the probability packet $\{d_j, a_j\}$ is not assigned l_x is one minus the probability to randomly choose link l_x in D_j. Since this assignment is random and changes the set of preferred outgoing links of one or more younger packets (of indices $>j$), we must sum over all possible ways that packet $\{d_j, a_j\}$ is not assigned link l_x ($|D_j| - 1$ possibilities). Similarly, if D_j does not contain l_x (case $C_j = \varnothing \wedge l_x \notin D_j$), because the random assignment of an outgoing link in D_j changes the set of preferred outgoing links of one or more younger packets, we must sum over all possible ways that packet $\{d_j, a_j\}$ is assigned a link in D_j ($|D_j|$ possibilities). The variable t represents packet $\{d_j, a_j\}$ outgoing link assignment (the tth element in D_j). It is also used to indicate that one or more younger packets have its set of preferred outgoing links changed by packet $\{d_j, a_j\}$ assignment:

$$\mathcal{P}^t_n = \begin{cases} \mathcal{P}_n & \text{if } \mathcal{P}_n \cap \{\text{the } t^{th}\text{element in } D_j\} = \varnothing, \\ \mathcal{P}_n \setminus \{\text{the } t^{th} & \\ \text{element in } D_j\} & \text{otherwise.} \end{cases}$$

When $j = x$, the recursive function stops with the expression $S^x(x|\mathcal{P}_x, \ldots, \mathcal{P}_k)$ given by

$$\begin{cases} 1/|C_x|, & \text{if } l_x \in C_x, \\ 1/|D_x|, & \text{if } C_x = \varnothing \wedge l_x \in D_x, \\ 1/(u - O_x) & \\ \quad \prod_{i=0}^{|O_x|-1} (1 - 1/(u-i)), & \text{if } \mathcal{P}_x = \varnothing, \\ 0, & \text{otherwise,} \end{cases} \tag{3}$$

where u is the number of outgoing links not assigned to packets older than $\{a_x, d_x\}$, and O_x is the number of packets with an empty set of preferred outgoing links and older than $\{a_x, d_x\}$.

In (3), when C_x is nonempty and contains l_x (case $l_x \in C_x$), the probability that l_x is assigned to packet $\{d_x, a_x\}$ is one over the size of set C_x (random assignment). When C_x is empty and the deflection set D_x contains l_x, the probability that l_x is assigned to packet $\{d_x, a_x\}$ is one over the size of set D_x (random assignment). When $\mathcal{P}_x$ is empty (case $\mathcal{P}_x = \varnothing$), the probability link l_x is assigned to packet $\{d_x, a_x\}$ is the probability that the O_x packets with empty set of preferred outgoing links and older than packet $\{d_x, a_x\}$ are not assigned to link l_x, times the probability packet $\{d_x, a_x\}$ is assigned link l_x, randomly selected out of the $u - O_x$ remaining links.

So, assuming packet arrivals to receive buffers are independent of one another, and of the state of neighboring nodes (independence and memoryless assumptions), the conditional probability in (1) is $S^x(1|\mathcal{P}_1, \mathcal{P}_2, \ldots, \mathcal{P}_x, \ldots, \mathcal{P}_k)$. When twin packets are present, to calculate the conditional probability in (1), we calculate $S^x(1|\mathcal{P}_1, \mathcal{P}_2, \ldots, \mathcal{P}_x, \ldots, \mathcal{P}_k)$ for each twin packet permutation, and take the average (twin packets are randomly sorted).

To compute the second term of (1), let packets $\{d_1, a_1\}, \ldots, \{d_x, a_x\}, \ldots, \{d_k, a_k\}$ arrive to node n receive buffers from respective input links $\{l_1, l_2, \ldots, l_k\}$. Then, assuming packet arrivals to the same node are independent, the probability that only this combination of k packets enter node n Rx buffers is

$$p_n(E_k) = \prod_{j=1}^{k} p^i_{n,l_j}(d_j;\, a_j) \cdot \prod_{j^*=k+1}^{d^i_n} \left(1 - \sum_{\mathcal{A}^*_k, \mathcal{D}^*_k, \mathcal{L}^*_k} p^i_{n,l^*_{j^*}}\left(d^*_{j^*},\, a^*_{j^*}\right)\right) \cdot \left(1 - \sum_{\mathcal{A}_1, \mathcal{D}_1} p^i_{n,0}(d; a)\right) \tag{4}$$

where $\mathcal{A}^*_k = \{a^*_{k+1}, a^*_{k+2}, \ldots, a^*_{d^i_n}\}$ represents all $(d^i_n - k)$ sets such that $a^*_j \in \{1, \ldots, A-1\}$ for $j \in [k+1, d^i_n]$. $\mathcal{D}^*_k = \{d^*_{k+1}, d^*_{k+2}, \ldots, d^*_{d^i_n}\}$ represents all $(d^i_n - k)$ sets such that $d^*_j \in \{0, 1, \ldots, N-1\} \setminus \{n\}$ for $j \in [k+1, d^i_n]$. $\mathcal{L}^*_k = \{l^*_{k+1}, l^*_{k+2}, \ldots, l^*_{d^i_n}\}$ represents all $(d^i_n - k)$ permutations of $\{1, 2, \ldots, d^i_n\} \setminus \{l_1, l_2, \ldots, l_k\}$. $\mathcal{A}_1 = \{0, 1, \ldots, A-1\}$ represents all possible age values the local packet can be (when there is no input buffer, the only possible value is 0). And $\mathcal{D}_1 = \{0, 1, \ldots, N-1\} \setminus \{n\}$ represents all possible destination values the local packet can have.

The first term of (4) represents the probability k packets enter node n receive buffers. The second term represents the probability the remaining $d^i_n - k$ receive buffers are empty. And the third term represents the probability node n does not generate a local packet. Note that if all receive buffers are full ($k = d^i_n$), the second and third terms are removed (recall transit packets have priority over local packets). Also, if a local packet is received ($\exists j \in [1, k] | l_j = 0$), the third term is removed.

When a finite input buffer is used, we must derive the steady-state probabilities associated with buffered local packets. Only $p^i_{n,0}(d;\, 0)$, the probability that node n local station transmits a new packet, and $p_b(n)$, the probability exactly d^i_n transit packets enter node n are needed.

We define the probability node n creates k_0 packets in the next time slot by

$$p_n(S_{k_0}) = \begin{cases} 1 - \sum_{d \in \mathcal{D}} p^i_{n,0}(d;\, 0), & \text{if } k_0 = 0 \\ \sum_{d \in \mathcal{D}} p^i_{n,0}(d;\, 0), & \text{if } k_0 = 1 \\ 0, & \text{else} \end{cases} \tag{5}$$

where $\mathcal{D} = \{0, 1, \ldots, N-1\} \setminus \{n\}$.

Next, we consider the phase where local packets enter the local buffer. During this phase, the evolution of the local buffer depends upon the local station

$$p_{n,0}(B_s) = \begin{cases} p_{n,0}(B_{s-1})\, p_n(S_{k_0=1}) + p_{n,0}(B_s), & \text{if } s = B_L \\ p_{n,0}(B_{s-1})\, p_n(S_{k_0=1}) + p_{n,0}(B_s) p_n(S_{k_0=0}), & \text{if } B_L > s > 0 \\ p_{n,0}(B_s) p_n(S_{k_0=0}), & \text{if } s = 0 \end{cases} \tag{6}$$

where B_L is the maximum local buffer size.

To account for local packet arrivals, for the phase where local packets enter the local buffer, the queued packets are updated as follows:

$$p^s_{n,0}(d;\, a;\, j) = \begin{cases} U(a = j-1) p_{n,0}(B_{s-1}) p^i_{n,0}(d;\, 0) + p^s_{n,0}(d;\, a;\, j) + p^{s-1}_{n,0}(d;\, a;\, j) p_n(S_{k_0=1}), & \text{if } s = B_L \\ U(a = j-1) p_{n,0}(B_{s-1}) p^i_{n,0}(d;\, 0) + p^s_{n,0}(d;\, a;\, j) p_n(S_{k_0=0}) + p^{s-1}_{n,0}(d;\, a;\, j) p_n(S_{k_0=1}), & \text{if } s < B_L \end{cases} \tag{7}$$

where $U(\text{true}) = 1$ and $U(\text{false}) = 0$. Note, $p^s_{n,0}(d;\, a;\, j) = 0$ if $j > s$.

So the probability for the local Tx buffer to send a local packet to node n Rx buffers is

$$p^i_{n,0}(d;\, a) = \sum_{s=1}^{B_L} p^s_{n,0}(d;\, a;\, 1). \tag{8}$$

During the phase where local packets depart from the local buffers, we know that packets can leave the local buffer only if

$<d_n^i$ transit packets are received. So, for the finite buffer case, the local buffer evolves as follows:

$$p_{n,0}(B_s) = \begin{cases} p_{n,0}(B_s)p_b(n), & \text{if } s = B_L \\ p_{n,0}(B_s)p_b(n) \\ \quad +p_{n,0}(B_{s+1})(1-p_b(n)), & \text{if } B_L > s > 0 \\ p_{n,0}(B_s) + p_{n,0}(B_{s+1})(1-p_b(n)), & \text{if } s = 0. \end{cases} \tag{9}$$

And the queued packets become

$$p^s_{n,0}(d;\, a;\, j) = \begin{cases} p^s_{n,0}(d;\, a;\, j)p_b(n), & \text{if } s = B_L \\ p^s_{n,0}(d;\, a;\, j)p_b(n) \\ \quad +p^{s+1}_{n,0}(d;\, a;\, j)(1-p_b(n)), & \text{if } s < B_L. \end{cases} \tag{10}$$

To model infinite input buffers we set B_L to a "large" value.

C. Model Implementation

Our model can accommodate arbitrary network architectures and traffic patterns. Its inputs are the *network connectivity matrix*, *traffic pattern*, *preferred outgoing links matrix*, the *specified accuracy*, and the *input buffer length*.

As mentioned, evaluation of the steady-state probabilities is iterative. At each iteration, using (1), we compute the output probabilities for every node. Then, we set the input probabilities to the respective output probabilities of the connected nodes (node n is connected to neighboring node $\tilde{n}$ via output link l or input link $\tilde{l}$) while incrementing packet age. Once the change in link utilizations from one iteration to the next is less than the specified accuracy ϵ, the iteration stops and we declare convergence to be reached. Furthermore, to reduce the total number of operations, we only consider packets with nonnull state ($p^i_{n,l}(d;\, a) > 0$). Hence, (1) summation over all possible packet age, destination and incoming link is reduced to a summation over all combinations of incoming packets with nonnull states. This is achieved by incorporating event-driven simulation techniques to our model implementation: an event queue to schedule packet departures, and link-lists to keep track of packets with nonnull states.

To describe the steps followed in our model implementation, we consider a node with nonempty receive buffers at iteration t. Then, using (1), we compute the probability (p^o) for each received packet to enter every neighboring node in the next iteration. Next, we send the received packets to every neighboring node for which p^o is nonzero. In other words, our model allows for a packet with multiple preferred outgoing links to be forwarded to more than one neighboring node at the next iteration (contending packets permitting). Consequently, more than one packet may enter the same receive buffer during the same time slot, and each such packet represents a possible outcome. When using (1) to compute the output probabilities, we limit the set of combinations ($\mathcal{A}_k$, $\mathcal{L}_k$, and $\mathcal{D}_k$) to the packets in the receive buffers. To apply these steps to all nodes, we use an event queue, equivalent to the one used for scheduling packet departures in event-driven simulations. Lastly, to achieve additional speed-up, we always use the most recently updated receive buffers.

IV. Store-and-Forward Model

A. Routing Algorithm

As with the deflection model, the switching fabric associates every received packet with a set of preferred outgoing links based on the shortest path [13] to the desired destination. The rule used by the switching fabric to map packets from receive to transmit buffers is also age-priority based. Packets are sorted by decreasing age order (twin packets are randomly sorted), starting with the oldest packet, an outgoing link is randomly selected out of its set of preferred outgoing links and the packet is switched to the corresponding transmit buffer. If the selected transmit buffer is full (finite buffer model), the packet is blocked and lost.

B. Steady-State Probabilities

For expository convenience, we have included in Table II the nomenclature of parameters used here. In the following, we do not repeat the derivation of the steady-state probabilities associated with buffered local packets [$p^i_{n,0}(d;\, a)$ and $p_{n,0}(B_s)$] since it has been done in the deflection model section (Section III-B).

To calculate the output probabilities $p^o_{n,l}$'s, let k packets arrive to node n at the beginning of a time slot. Define these k packets by $\{d_1,\, a_1\}$, $\{d_2,\, a_2\}$, $\ldots$, $\{d_k,\, a_k\}$ and assume their respective input links to be $\{l_1,\, l_2,\, \ldots,\, l_k\}$. Assuming packet arrivals to the same node are independent, the probability this combination of k packets enter node n Rx buffers is [as in (4)]

$$p_n(E_k) = \prod_{j=1}^{k} p^i_{n,l_j}(d_j,\, a_j) \cdot \prod_{j^*=k+1}^{d_n^i} \left(1 - \sum_{\mathcal{A}_k^*,\, \mathcal{D}_k^*,\, \mathcal{L}_k^*} p^i_{n,l^*_{j^*}}\left(d^*_{j^*},\, a^*_{j^*}\right)\right) \cdot \left(1 - \sum_{\mathcal{A}_1,\, \mathcal{D}_1} p^i_{n,0}(d;\, a)\right) \tag{11}$$

where $\mathcal{A}_k^* = \{a^*_{k+1},\, a^*_{k+2},\, \ldots,\, a^*_{d_n^i}\}$ represents all $(d_n^i - k)$ sets such that $a_j^* \in \{1,\, \ldots,\, A-1\}$ for $j \in [k+1,\, d_n^i]$. $\mathcal{D}_k^* = \{d^*_{k+1},\, d^*_{k+2},\, \ldots,\, d^*_{d_n^i}\}$ represents all $(d_n^i - k)$ sets such that $d_j^* \in \{0,\, 1,\, \ldots,\, N-1\} \setminus \{n\}$ for $j \in [k+1,\, d_n^i]$. $\mathcal{L}_k^* = \{l^*_{k+1},\, l^*_{k+2},\, \ldots,\, l^*_{d_n^i}\}$ represents all $(d_n^i - k)$ permutations of $\{1,\, 2,\, \ldots,\, d_n^i\} \backslash \{l_1,\, l_2,\, \ldots,\, l_k\}$. $\mathcal{A}_1 = \{0,\, 1,\, \ldots,\, A-1\}$ represents all possible age values the local packet can be (when there is no input buffer, the only possible value is 0). And $\mathcal{D}_1 = \{0,\, 1,\, \ldots,\, N-1\} \backslash \{n\}$ represents all possible destination values the local packet can have.

Next, assuming selection of an outgoing link from the set of preferred outgoing links is random and independent of Tx buffer states, we calculate the probability given E_k that exactly k_x packets, taken from the k packets in the receive buffer and

TABLE II
NOMENCLATURE OF PARAMETERS USED IN THE STORE-AND-FORWARD MODEL FOR STEADY-STATE PROBABILITIES

N	Total number of nodes.
$\{d_j, a_j\}$	j^{th} packet in node n receive buffer, of destination node d_j and age a_j.
$p^i_{n,l}(d;a)$	Probability packet $\{d,a\}$ *arrives* to node n on link l in the next time slot. ($p^i_{n,0}(d;0)$ is the probability node n creates a local packet destined to node d.)
$p^o_{n,l}(d;a+1)$	Probability packet $\{d,a\}$ *leaves* node n on link l in the next time slot. ($p^o_{n,l}(n;a)=0$.)
$p_n(E_k)$	Probability that only $\{d_1,a_1\},\{d_2,a_2\}$, ..., $\{d_k,a_k\}$ entered node n Rx buffers.
$p_n(S_{k_j})$	Probability k_j packets are switched to node n Tx buffer j.
$p_{n,l}(B_s)$	Probability node n Tx buffer l has s queued packets.
$p^s_{n,l}(d;a;j)$	Probability packet $\{d;a\}$ is *queued* at position j of size s buffer l in node n
A	Age bound. $p^o_{n,l}(d;a)=p^i_{n,l}(d;a)=0$ for $a \geq A$.
$\mathcal{A}_k$	All $(k-1)$ sets spanning over $\{0,1,...,A-1\}$. If all receive buffers are occupied, or the test packet $\{d_x,a_x\}$ has age 0, then $\mathcal{A}_k$ spans over $\{1,...,A-1\}$.
$\mathcal{D}_k$	All $(k-1)$ sets spanning over $\{0,1,...,N-1\}$, excluding node n.
$\mathcal{L}_k$	All k-subsets of the d^i_n-set $\{1,2,...,d^i_n\}$.
$\mathcal{P}_j$	Packet $\{d_j,a_j\}$ set of preferred outgoing links.

combined in decreasing age order, are switched to output buffer $l_x(\sum_{j=1}^{d^o_n} k_j = k)$:

$$p_n(S_{k_x}|E_k) = \prod_{i=1}^{k_x} \frac{1}{|\mathcal{P}_i|} \prod_{j=1}^{k-k_x} \left(1 - \frac{1}{|\mathcal{P}'_j|}\right) \tag{12}$$

where $\mathcal{P}_i$, $i \in [1, k_x]$ represents the sets of outgoing links preferred by the k_x packets, $\mathcal{P}'_i$, $i \in [1, k-k_x]$ represents the sets of outgoing links preferred by the $k-k_x$ packets, and

$$\frac{1}{|\mathcal{P}_i|} = \begin{cases} 1/|\mathcal{P}_i| & \text{if } \mathcal{P}_i \cap \{l_x\} \neq \emptyset, \\ 0 & \text{else.} \end{cases}$$

To calculate the probability k_x packets are switched to Tx buffer x, we apply the total probability theorem

$$p_n(S_{k_x}) = \sum_{k=0}^{d^i_n} \sum_{E_k} p_n(S_{k_x}|E_k)\, p_n(E_k) \tag{13}$$

where the summation over E_k represents all possible size k sets of incoming packets.

Next, we calculate the probability output buffer $l_x > 0$ is of length s after switching packets from node n Rx buffers to node n Tx buffers:

$$p_{n,l_x}(B_s) = \begin{cases} \displaystyle\sum_{k_x=0}^{d^i_n} \sum_{u=B_T-k_x}^{B_T} p_{n,l_x}(B_u) p_n(S_{k_x}), & \text{if } s = B_T \\ \displaystyle\sum_{k_x=0}^{\min(s,d^i_n)} p_{n,l_x}(B_{s-k_x})\, p_n(S_{k_x}), & \text{else} \end{cases} \tag{14}$$

where B_T is the transmit buffer maximum size. Note that we calculate this probability by decreasing s order.

After switching packets from node n Rx buffers to node n Tx buffers, probabilities of packets arriving or queued in Tx buffer x are updated as follows ($l_x > 0$):

$$p^s_{n,l_x}(d;\, a;\, j) = \begin{cases} \displaystyle\sum_{k_x=0}^{d^i_n} \sum_{u=B_T-k_x}^{j-1} p_{n,l_x}(B_u) \sum_{f(u,j)} p_n(S_{k_x}|E_k)\, p(E_k) \\ \quad + \displaystyle\sum_{k_x=0}^{d^i_n} \sum_{u=B_T-k_x}^{B_T} p^u_{n,l_x}(d;\, a;\, j), & \text{if } s = B_T \\ p_n(S_{k_x}) \displaystyle\sum_{k_x=s-j+1}^{\min(s,d^i_n)} p_{n,l_x}(B_{s-k_x}) \sum_{f(s-k_x,j)} p_n(S_{k_x}|E_k)\, p_n(E_k) \\ \quad + \displaystyle\sum_{k_x=0}^{d^i_n} p^{s-k_x}_{n,l_x}(d;\, a;\, j) p_n(S_{k_x}), & \text{else} \end{cases} \tag{15}$$

where the summation over $f(u,j)$ represents the summation over all E_k (combination of incoming packets) and S_{k_x} (subset of incoming packets switched to l_x) that contains packets of age $a-u$, of destination d, and of position $j-u$ in the ordered set S_{k_x} (decreasing age order). Note that $p^u_{n,l_x}(d;\, a;\, j) = 0$ if $j > u$.

Finally, the probability for a packet to exit node n Tx buffer after switching packets from Rx buffers to Tx buffers is

$$p^o_{n,l_x}(d;\, a) = \sum_{s=0}^{B} p^s_{n,l_x}(d;\, a;\, 1). \tag{16}$$

After packet departure from Tx buffers, we remove level 1 of Tx buffers, so the buffer and queued packet probabilities evolve as follows:

$$\begin{aligned} &p_{n,l}(B_s) = p_{n,l}(B_0) + p_{n,l}(B_1), && \text{if } s = 0 \\ &p_{n,l}(B_{s-1}) = p_{n,l}(B_s), && \text{if } B_T > s > 0 \\ &p_{n,l}(B_s) = 0, && \text{if } s = B_T \\ &p^{s-1}_{n,l_x}(d;\, a;\, j-1) = p^s_{n,l_x}(d;\, a;\, j), && \text{if } j > 1. \end{aligned} \tag{17}$$

C. Model Implementation

The implementation of the store-and-forward routing model is very similar to the one for the deflection model. The same inputs are used, multiplication of packets faced with more than one preferred outgoing link also occurs, and an event queue is used to schedule departures of packets with nonnull states.

As with the deflection model, we use an iterative procedure to solve for the output probabilities and the buffer state probabilities. Consider a node with receive buffers of nonzero state probabilities $[p_{n,l}(B_s) \neq 0]$ at iteration t. In the first phase of our implementation, we extract all possible combinations E_k of packets located in the receive buffers and the bottom-most local Tx buffer $[p_n(E_k) \neq 0$ and $k \leq d_n^i]$. Then, for each such combination, we calculate all k_x-subsets of E_k (noted S_{k_x}) that can be switched to outgoing link k_x $[p_n(S_{k_x}|E_k) \neq 0]$. Each packet within S_{k_x} is assigned the state probability $p_n(S_{k_x}|E_k)p_n(E_k)$. After sorting the packets in S_{k_x} by descending age order, we combine them with Tx buffer x states (buffers of size ranging from 0 to B, where each size represents a possible outcome) and update the buffer and packet probabilities following (14) and (15). While repeating this procedure for each combination E_k, we update $p_b(n)$. At the end of iteration t, we forward all packets in the bottom-most level of the Tx buffers to their neighboring nodes, and we remove the bottom-most level of all Tx buffers (so that output probabilities become input probabilities of neighboring nodes).

Because departure of packets from the local Tx buffers depends on transit packets, we handle the local Tx buffers differently. At the beginning of iteration t, before constructing E_k's, we update the local Tx buffer to account for the local station following (7) and (6). And when a new packet is created $(p_n(S_{k_0=1}) > 0)$, we append it to the local Tx buffer states (buffers of size ranging from 0 to B, where each size represents a possible outcome). Finally, the last operation during iteration t consists of updating the local Tx buffer to account for local packet departure (bottom-most level is removed) and local packet queueing (the age of all buffered packets is incremented by one) as shown in (10) and (9).

V. Comparison to Simulation

In this section, we compare our models to simulation for a few traffic patterns and network topologies. We begin by introducing the simulation model (Section V), the performance metrics (Section V-B), and the multihop networks under test (Section V-C). Then, in the remaining sections, we make the comparison for uniform traffic (Section V-D), single node accumulation traffic (Section V-E), and random traffic (Section V-F).

A. Simulation Model

We developed an event-driven simulator that implements our network model (presented in Section II), with either the deflection-routing algorithm (presented in Section III) or the store-and-forward routing algorithm (presented in Section IV). As with the models, we used geometric distributions to generate interdeparture times.

For each traffic and network type, the simulation results were estimated from ten independent runs of 200 000 departures each (statistics associated with the first 50 000 departures were disregarded). The resulting performance parameters were averaged over the ten replications and a 95% confidence interval (interval around the sample mean that captures 95% of the samples) was constructed by assuming the normalized error to be t-distributed [33].

B. Performance Metrics

From the steady-state probabilities, we derived the following performance metrics: the blocking probability p_b (the probability that a packet fails to arrive at its destination), the delay distribution $h_a(a)$ (the probability that a packet arrives to its destination node in a hops), the mean delay μ_a, the buffer queue length distribution $h_s(s)$, the mean buffer queue length $\mu_s(n, l)$, the outgoing link utilization $U_n^o(l)$ (the probability that a packet *exits* node n on link l in the next time slot), the incoming link utilization $U_n^i(l)$ (the probability that a packet *enters* node n on link l in the next time slot), the outgoing packet rate R_n^o (the number of packets *exiting* node n in the next time slot), and the incoming packet rate R_n^i (the number of packets *entering* node n in the next time slot). They are summarized as follows:

$$p_b = 1 - \frac{\sum_n \sum_l \sum_a p_{n,l}^i(n;\, a)}{\sum_n \sum_d p_{n,0}^i(d;\, 0)} \tag{18}$$

$$h_a(a) = \frac{\sum_n \sum_l p_{n,l}^i(n,\, a)}{\sum_n \sum_l \sum_a p_{n,l}^i(n,\, a)} \tag{19}$$

$$\mu_a = \sum_a a h_s(a) \tag{20}$$

$$h_s(n,\, l,\, s) = p_{n,l}(B_s) \tag{21}$$

$$\mu_s(n,\, l) = \sum_s s h_s(s) \tag{22}$$

$$U_n^o(l) = \sum_d \sum_a p_{n,l}^o(d; a) \tag{23}$$

$$U_n^i(l) = \sum_d \sum_a p_{n,l}^i(d; a) \tag{24}$$

$$R_n^o = \sum_l U_n^o(l) \tag{25}$$

$$R_n^i = \sum_l U_n^i(l). \tag{26}$$

To compare model and simulation for all parameters but delay and queue length distributions, we used the 95% confidence interval. For the delay and queue length distributions, we calculated the relative error $\mathcal{E}$ defined by

$$\mathcal{E}(h) = 100 \sum_{i,\hat{h}(i)\neq 0} \frac{|h(i) - \hat{h}(i)|}{n\hat{h}(i)}$$

where $\hat{h}$ is the distribution derived from simulation (histogram averaged over each simulation replication), and n is the number of terms in the summation.

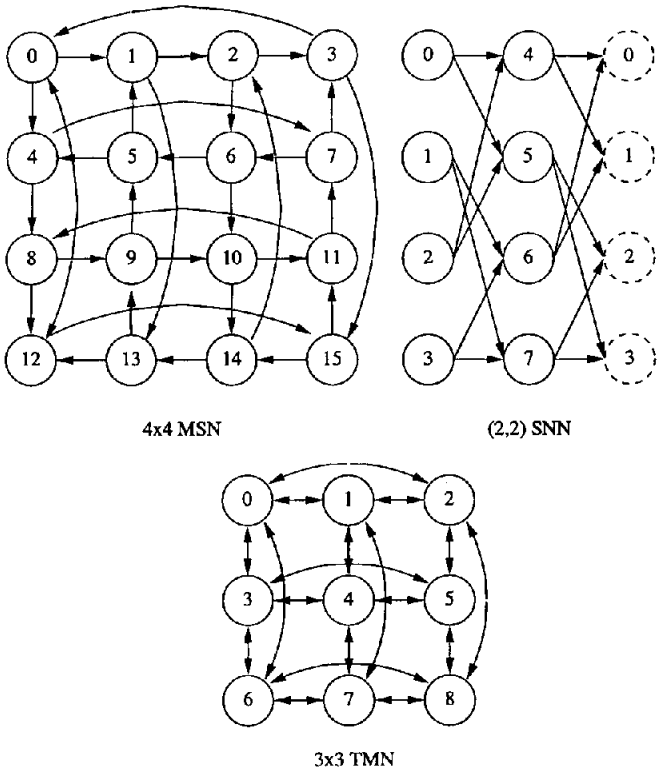


Fig. 2. Example of a 4 × 4 Manhattan Street Network, a 3 × 3 Toroidal Mesh Network, and a (2, 2) ShuffleNet Network.

C. Networks Under Test

To compare our models to simulation, we use the Manhattan Street Network [38], the Toroidal Mesh Network [7], and the ShuffleNet Network [1] (see Fig. 2).

The Manhattan Street Network is a degree 2 directed mesh connected network, with its links resembling the one-way streets and avenues of Manhattan (even number of rows and columns). The Toroidal Mesh Network is a degree 4 network, similar to the Manhattan Street Network except that all its links are bidirectional. And the (p, k) ShuffleNet Network is a degree p unidirectional cylindrically connected Omega network.

The diameters (longest distance between two nodes) of an R rows by C columns ($R \times C$) Manhattan Street Network (D_m), $R \times C$ Toroidal Mesh Network (D_t), and $p^k \times k(p, k)$ ShuffleNet Network (D_s) are ([11], [55], [2], respectively)

$$D_m = \begin{cases} C/2 + R/2 + 1, & \text{if } R \bmod 4 = 0 \wedge \\ & \quad C \bmod 4 = 0 \\ C/2 + R/2, & \text{else} \end{cases}$$
$$D_t = (C-1)/2 + (R-1)/2$$
$$D_s = 2C - 1.$$

Note that because our models implementations are independent of network topologies, our naming convention of nodes does not necessarily take advantage of topology symmetries. However, our definitions for adjacent nodes are equivalent to the ones described in [38] for the Manhattan Street Network, [7] for the Toroidal Mesh Network, and [1] for the ShuffleNet Network.

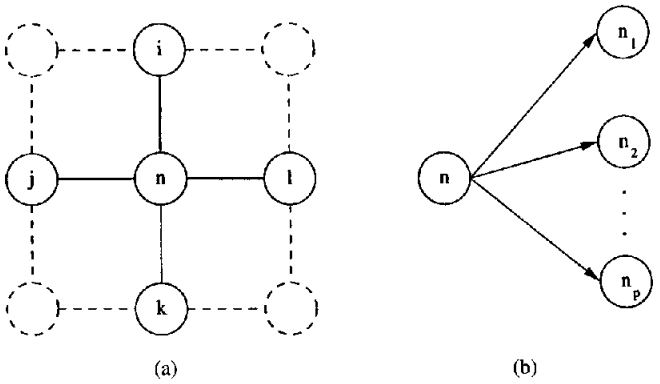


Fig. 3. Definition of connected neighbors. (a) Manhattan Street Network and Toroidal Mesh Network. (b) ShuffleNet Network.

For both the Manhattan Street Network and the Toroidal Mesh Network, neighboring nodes are defined by (see Fig. 3)

$$n = rC + c$$
$$i = (r - 1 \bmod R)C + c$$
$$j = rC + (c - 1 \bmod C)$$
$$k = (r + 1 \bmod R)C + c$$
$$l = rC + (c + 1 \bmod C).$$

In addition, for the Manhattan Street Network, the direction of connections is defined by

$$\begin{aligned} n &\to i \quad \text{if } c \bmod 2 = 1 \\ n &\to k \quad \text{if } c \bmod 2 = 0 \\ n &\to j \quad \text{if } r \bmod 2 = 1 \\ n &\to l \quad \text{if } r \bmod 2 = 0. \end{aligned}$$

Whereas for the (p, k) ShuffleNet Network, adjacent nodes are defined by (see Fig. 3)

$$n = cR + r$$
$$n_i = \begin{cases} (i + p(r \bmod p^{k-1}) \\ \quad +(c+1)R) \bmod R, & \text{if } c = C - 1 \\ i + p(r \bmod p^{k-1}) + (c+1)R, & \text{else.} \end{cases}$$

D. Uniform Traffic

With uniform traffic, all nodes within the network transmit packets to all other nodes with the same probability. And when applied to symmetric networks, uniform traffic allows us to reduce our models to the analysis of steady-state probabilities for a single node (say node 0). As shown in Fig. 4, this is done with the relabeling operator $R_n(d)$ [38] which maps packets exiting node 0 and entering node n at iteration t (identified in terms of age a, destination d, and output probability p) into packets entering node 0 from node $R_n(0)$ at iteration $t + 1$ [of age incremented by one and destination transformed by the relabeling operator $R_n(d)$].

Using the coordinate systems defined in Fig. 4, let $d = (r_d, c_d)$ and $n = (r_n, c_n)$ in coordinate system $(0, r, c)$. The operator $R_n(d)$ maps the coordinates of d from $(0, r, c)$ to (n, r', c') as follows:

$$\begin{cases} r'_d = (r_d - r_n)\cos(\theta) + (c_d - c_n)\sin(\theta) \bmod R \\ c'_d = (r_n - r_d)\sin(\theta) + (c_d - c_n)\cos(\theta) \bmod C. \end{cases} \quad (27)$$

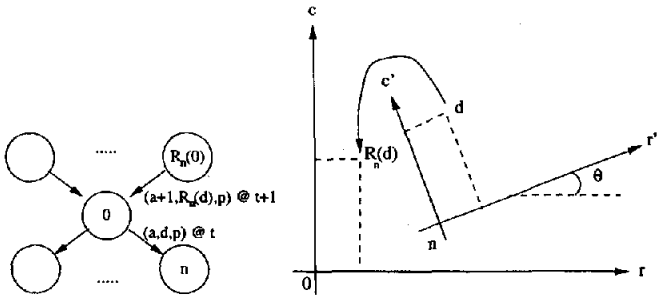


Fig. 4. Application of the relabeling operator.

TABLE III
COMPARISON BETWEEN THE MODELS AND SIMULATION (NUMBERS IN PARENTHESIS WITH 95% CONFIDENCE INTERVAL) FOR A 8 × 8 MANHATTAN STREET NETWORK SUBJECT TO THE UNIFORM TRAFFIC PATTERN. (p_b: BLOCKING PROBABILITY, μ_a: MEAN DELAY, LOCAL μ_s: MEAN QUEUE LENGTH OF THE LOCAL TX BUFFER, TRANSIT μ_s: MEAN QUEUE LENGTH OF THE TRANSIT TX BUFFERS, U_0^o: OUTGOING LINK UTILIZATION)

	8x8 MSN		
	Deflection		Store-and-Forward
	$B_L = 0$, $B_T = 0$	$B_L = 3$, $B_T = 0$	$B_L = 3$, $B_T = \infty$
p_b	0.020 (0.020±0.001)	$< 10^{-6}$ (0)	$< 10^{-6}$ (0)
μ_a	5.585 (5.590±0.007)	5.621 (5.630±0.004)	5.312 (5.327±0.002)
Local μ_s	NA	0.064 (0.065±0.001)	0.064 (0.064±0.001)
Transit μ_s	NA	NA	0.167 (0.167±0.002)
U_0^o	0.172 (0.175±0.003)	0.176 (0.176±0.002)	0.158 (0.160±0.002)

For the Toroidal Mesh Network $\theta = 0$, and for the Manhattan Street Network, θ is defined as follows:

$$\theta = \begin{cases} 0 & \text{if } r_n \bmod 2 = 0 \wedge c_n \bmod 2 = 0 \\ \pi/2 & \text{if } r_n \bmod 2 = 0 \wedge c_n \bmod 2 = 1 \\ \pi & \text{if } r_n \bmod 2 = 1 \wedge c_n \bmod 2 = 1 \\ 3\pi/2 & \text{if } r_n \bmod 2 = 1 \wedge c_n \bmod 2 = 0. \end{cases} \quad (28)$$

The network tested was a 8×8 Manhattan Street Network where each of the local stations transmitted packets with probability 63/1000. In Table III, we compare the models and the simulation blocking probability, mean delays, and outgoing link utilization. In Fig. 5 we compare the models and the simulation delay histograms (relative error of ≈5% for each histogram), and in Fig. 6 we compare the models and the simulation queue length histograms (relative error <1% for the local queue length and ≈12% for the transit queue length). We found good agreement between the model and simulation. Note that for the buffered deflection model and the store-and-forward model, the blocking probability (p_b) calculated with our model is less than the convergence bound of 10^{-6} (or below the model accuracy), which is consistent with our simulation results where packets were never blocked.

As noted in [49], addition of input buffers to deflection routing significantly improves blocking at a negligible penalty

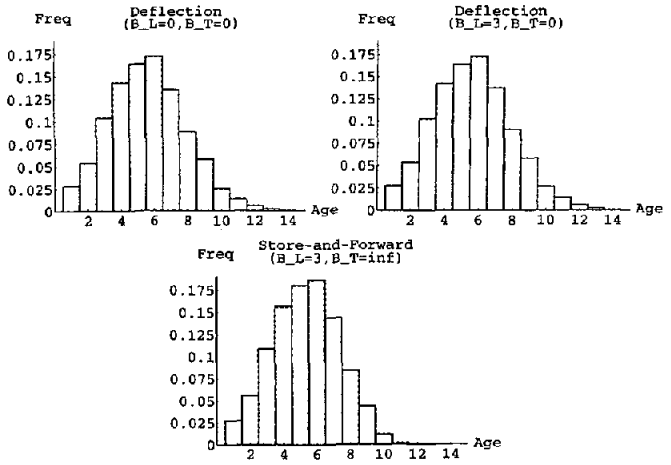


Fig. 5. Delay histograms for a 8 × 8 Manhattan Street Network subject to the uniform traffic pattern (continuous: model, dashed: simulation).

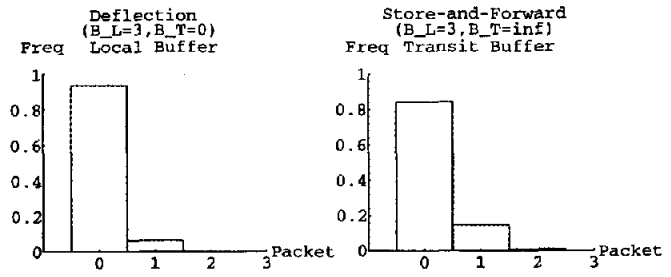


Fig. 6. Queue length histograms for 8 × 8 Manhattan Street Network subject to the uniform traffic pattern (continuous: model, dashed: simulation).

in delay and link utilization. Since with store-and-forward routing, the penalty for conflicting packets is queueing (mean queue length is 0.167) as opposed to deflection (maximum penalty of four hops for the Manhattan Street Network [38]) delay is noticeable lower for the store-and-forward routing.

E. Single Node Accumulation

The single node accumulation traffic pattern corresponds to all but one node transmit to the same node at a rate of $1/(N-1)$. This traffic pattern corresponds to the scenario where all nodes of a multiprocessor system send messages to a single node, as found in applications such as relaxation iterations [6]. The network tested was a 9 × 11 Toroidal Mesh Network, and the node accumulation was node 49.

In Table IV, we compare the models and the simulations blocking probability, mean queue lengths, mean delay, and outgoing link utilization. For the blocking probability, our models predict values less than the convergence bound of 10^{-6} (or below the model accuracy), which is consistent with our simulation results. For the local mean queue length, all queues had the same value so we used node 0. For the transit mean queue length, we considered the queue with largest value (node 50, out going link 2). In Fig. 7, we compare the models and the simulation delay histograms (relative error of ≈6% for each histogram). In Fig. 8, we show the queue length histogram of node 50, outgoing link 3 Tx buffer for the store-and-forward

TABLE IV
COMPARISON BETWEEN THE MODELS AND SIMULATION (NUMBERS IN PARENTHESIS WITH 95% CONFIDENCE INTERVAL) FOR A 9 × 11 TOROIDAL MESH NETWORK SUBJECT TO THE SINGLE NODE ACCUMULATION TRAFFIC PATTERN. (p_b: BLOCKING PROBABILITY, μ_a: MEAN DELAY, LOCAL max μ_s: MAX MEAN QUEUE LENGTH OF LOCAL TX BUFFERS, TRANSIT max μ_s: MAX MEAN QUEUE LENGTH OF TRANSIT TX BUFFERS, max U^o: MAX OUTGOING LINK UTILIZATION.)

	9x11 TMN		
	Deflection		Store-and-Forward
	$B_L = 0$, $B_T = 0$	$B_L = 3$, $B_T = 0$	$B_L = 3$, $B_T = \infty$
p_b	$< 10^{-6}$ (0)	$< 10^{-6}$ (0)	$< 10^{-6}$ (0)
μ_a	5.277 (5.300±0.007)	5.277 (5.298±0.004)	5.163 (5.169±0.006)
Local μ_s	NA	0.010 (0.010±0.001)	0.010 (0.010±0.001)
Transit μ_s	NA	NA	0.305 (0.310±0.001)
U_0^o	0.278 (0.277±0.001)	0.278 (0.278±0.001)	0.281 (0.282±0.001)

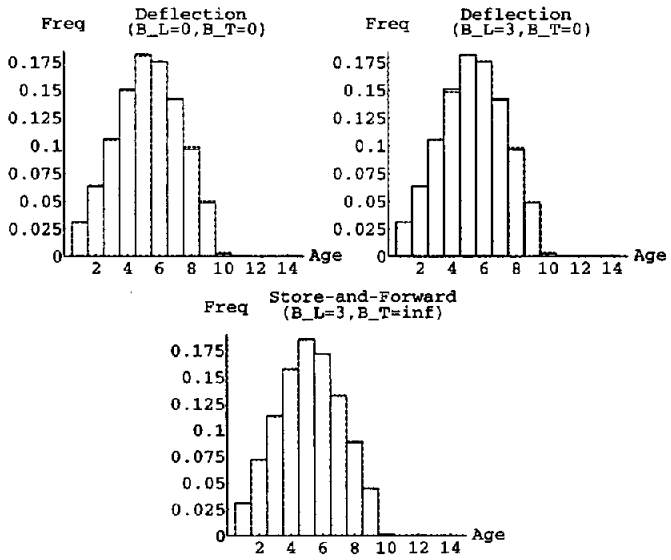


Fig. 7. Delay histograms for a 9 × 11 Toroidal Mesh Network subject to the single node accumulation traffic pattern (continuous: model, dashed: simulation).

model (relative error of ≈10%). In Fig. 9, we show the incoming and outgoing packet rates for each model. We found good agreement between the model and simulation.

Since no blocking was experienced, the effects of adding input buffers did not result in any noticeable performance improvements. Moreover, we did not see noticeable differences between deflection routing and store-and-forward routing. This is because with deflection routing the deflection frequency was low (maximum deflection probability ≈0.02, with a maximum penalty of two hops) and with store-and-forward routing, the deflection penalty was also low (maximum mean queue length ≈0.3).

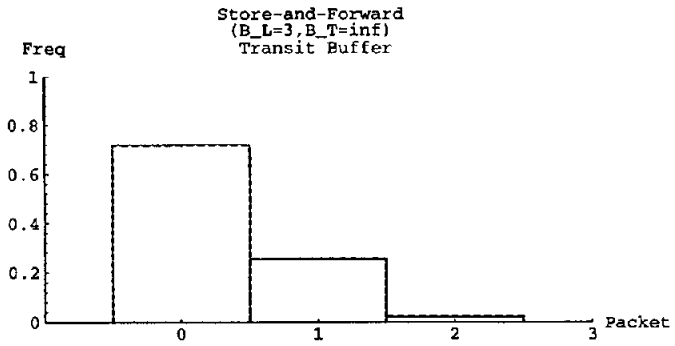


Fig. 8. Queue length histograms for a 9 × 11 Toroidal Mesh Network subject to the single-node accumulation traffic pattern (continuous: model, dashed: simulation).

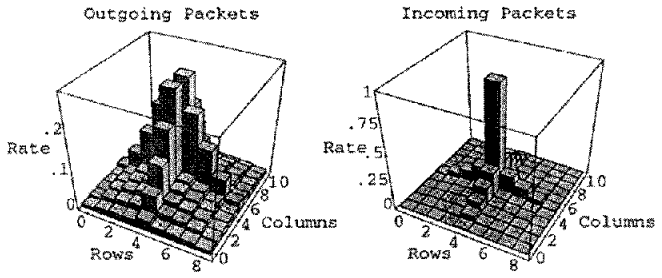


Fig. 9. Outgoing and incoming packet rate for a 9 × 11 Toroidal Mesh Network subject to single-node accumulation traffic pattern (same results for deflection and store-and-forward routing).

F. Random Traffic

We use a random traffic pattern to confirm our models can be used for arbitrary traffic patterns. The random traffic pattern was created by assuming each active source sends a packet with probability 1/2, and each source and destination pair to be active with probability 12/10 000. The network tested was a (3, 3) ShuffleNet Network, and our random traffic pattern resulted with the traffic matrix shown in Table V.

In Table VI, we compare the model and the simulation mean delay, blocking probability, mean queue lengths, and outgoing link utilization. For the maximum local queue lengths we used node 0, and for the maximum transit queue length we used node 48, outgoing link 2. In Fig. 10 we compare the model and the simulation delay histograms (relative error of ≈20% for each histogram). And in Fig. 11 we compare the model and the simulation queue length histograms (relative error of ≈40%). We found the agreement between the model and simulation to be not as good as with the previous traffic patterns.

We found that addition of input buffers to deflection routing improved blocking. Also, we did not find significant differences between deflection routing with input buffers and store-and-forward routing.

VI. CONCLUSION

In this article, we presented two performance models for multihop networks under nonuniform traffic pattern. The models are a generalization of Greenberg–Goodman and Brassil–Cruz models which were designed specifically for Manhattan Street Networks [20], [8]. Our model, on the

TABLE V
TRAFFIC MATRIX FOR THE RANDOM TRAFFIC PATTERN

Source	Destinations	Rate
0	44, 61	1/4
16	64	1/2
22	55	1/2
38	30, 57	1/2
46	0	1/2
48	76	1/2
52	64	1/2
57	72	1/2
61	76	1/2
72	54	1/2
77	67	1/2

TABLE VI
COMPARISON BETWEEN THE MODELS AND SIMULATION (NUMBERS IN PARENTHESIS WITH 95% CONFIDENCE INTERVAL) FOR A (3, 3) SHUFFLENET NETWORK SUBJECT TO THE RANDOM TRAFFIC PATTERN. (p_b: BLOCKING PROBABILITY, μ_a: MEAN DELAY, LOCAL $\max \mu_s$: MAX MEAN QUEUE LENGTH OF LOCAL TX BUFFERS, TRANSIT $\max \mu_s$: MAX MEAN QUEUE LENGTH OF TRANSIT TX BUFFERS, $\max U^o$: MAX OUTGOING LINK UTILIZATION)

	(3,3) SNN		
	Deflection		Store-and-Forward
	$B_L = 0$, $B_T = 0$	$B_L = 3$, $B_T = 0$	$B_L = 3$, $B_T = \infty$
p_b	$2\ 10^{-4}$ ($2\ 10^{-4} \pm 4\ 10^{-5}$)	$< 10^{-6}$ (0)	$< 10^{-6}$ (0)
μ_a	3.901 (3.847±0.003)	3.900 (3.849±0.003)	3.957 (3.970±0.008)
Local μ_s	NA	0.500 (0.502±0.003)	0.500 0.502±0.003)
Transit μ_s	NA	NA	1.500 (1.691±0.032)
U_0^o	0.669 (0.667±0.003)	0.668 (0.668±0.002)	0.833 (0.834±0.004)

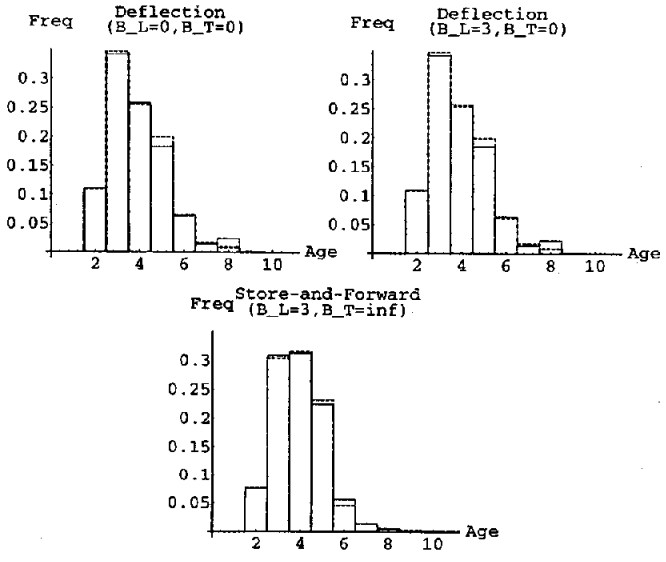


Fig. 10. Delay histograms for a (3, 3) ShuffleNet Network subject to the random traffic pattern (continuous: model, dashed: simulation).

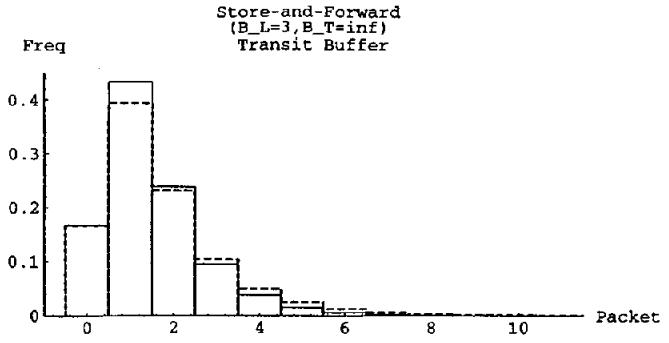


Fig. 11. Queue length histograms for a (3, 3) ShuffleNet Network subject to the random traffic pattern (continuous: model, dashed: simulation).

other hand, can be applied to an arbitrary network topology of arbitrary degree. Furthermore, by considering packets with nonnull states only, our model is computationally more efficient than Greenberg–Goodman and Brassil–Cruz direct implementations.

As an application, we compared our models against simulation for a 8 × 8 Manhattan Street Network subject to uniform traffic, of a 9 × 11 Toroidal Network subject to single node accumulation traffic, and of a (3, 3) ShuffleNet Network subject to random traffic. We found the model provides good agreement with simulation.

By incorporating event-driven simulation methodology and considering packets with nonnull states only, our model implementations have an improved time efficiency. For example, with the 8 × 8 Manhattan Street Network, our model provides several orders of magnitude run time improvement over the Greenberg–Goodman and Brassil–Cruz implementations.

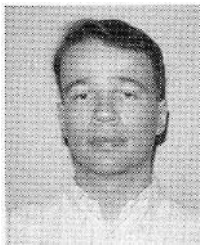
ACKNOWLEDGMENT

The authors would like to thank the anonymous referees for their helpful comments and suggestions.

REFERENCES

[1] A. S. Acampora and M. J. Karol, "An overview of lightwave packet networks," *IEEE Network*, vol. 3, pp. 29–41, Jan. 1989.
[2] A. S. Acampora, M. J. Karol, and M. G. Hluchyj, "Terabit lightwave networks: The multihop approach," *AT&T Tech. J.*, vol. 66, no. 4, pp. 21–34, Nov. 1987.
[3] A. S. Acampora and S. I. A. Shah, "Multihop lightwave networks: A comparison of store-and-forward and hot-potato routing," *IEEE Trans. Commun.*, vol. 40, pp. 1082–1090, June 1992.
[4] W. C. Athas and C. L. Seitz, "Multicomputers: Message-passing concurrent computers," *IEEE Comput. Mag.*, vol. 21, pp. 9–24, Aug. 1988.
[5] P. Baran, "On distributed computing networks," *IEEE Trans. Commun. Syst.*, vol. CS-12, pp. 1–9, Mar. 1964.
[6] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation Numerical Methods*. Englewood Cliffs, NJ: Prentice Hall, 1989.
[7] F. Borgonovo and E. Cadorin, "HR^4-net: Hierarchical random-routing reliable and reconfigurable network for metropolitan area," in *Proc. IEEE Infocom'87*, pp. 320–326.
[8] J. Brassil and R. Cruz, "Nonuniform traffic in the Manhattan street network," in *Proc. 1991 IEEE Int. Conf. Communications*, 1991, pp. 1647–1651.
[9] J. Brassil and R. Cruz, "Nonuniform traffic in the Manhattan street network," *Bell Labs. Tech. Memo.*, pp. BL0 113 820–930 204–58TM, 1993.
[10] W. Bux, "A reliable token-ring system for local area communication," in *National Telecommunications Conf.*, New Orleans, LA, 1981, pp. A.2.2.1–A.2.2.6.

[11] T. Y. Chung and D. P. Agrawal, "On network characterization of and optimal broadcasting in the Manhattan street network," in *Proc. IEEE Infocom'90*, pp. 465–471.
[12] W. J. Dally and C. L. Seitz, "Deadlock-Fress message routing in multiprocessor interconnection networks," *IEEE Trans. Comput.*, vol. C-36, pp. 547–553, May 1987.
[13] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, pp. 269–271, 1959.
[14] V. H. Mac Donald, "Advanced mobile phone service: The cellular concept," *Bell Syst. Tech. J.*, vol. 58, no. 1, pp. 15–40, Jan. 1979.
[15] Z. C. Fluhr and P. T. Porter, "Advanced mobile phone service: Control architecture," *Bell Syst. Tech. J.*, vol. 58, pp. 43–69, Jan. 1979.
[16] E. Foo and T. Robertazzi, "Packet trains in the Manhattan street network using deflection routing," in *2nd Int. Conf. Computer Communications and Networks*, June 1993, pp. 1–5.
[17] V. S. Frost and B. Melamed, "Traffic modeling for telecommunications networks," *Commun. ACM*, pp. 70–81, Mar. 1994.
[18] P. Le Gall, "Traffic modeling in packet switched networks for single links," *Annales des Télécommun.*, vol. 49, no. 3–4, pp. 111–126, 1994.
[19] A. Greenberg and B. Hajek, "Deflection routing in hypercube networks," *IEEE Trans. Commun.*, vol. 40, pp. 1070–1081, June 1992.
[20] A. G. Greenberg and J. Goodman, "Sharp approximation of adaptive routing in mesh networks," in *Teletraffic Analysis and Computer Performance Evaluation*, H. C. Tijms, O. J. Boxma, and J. W. Cohen, Eds. Amsterdam, The Netherlands: North Holland, 1986.
[21] ——, "Sharp approximate models of deflection routing in mesh networks," *IEEE Trans. Commun.*, vol. 41, pp. 210–223, June 1992.
[22] K. D. Gunther, "Prevention of deadlocks in packet-switched data transport systems," *IEEE Trans. Commun.*, vol. 29, pp. 512–524, Apr. 1981.
[23] H.-Y. Huang, T. Robertazzi, and A. A. Lazar, "A comparison of information based deflection strategies," *Comput. Networks ISDN Syst.*, vol. 27, pp. 1388–1407, 1995.
[24] V. I. Istăţescu, *Fixed Point Theory, An Introduction*. Amsterdam, The Netherlands: Reidel, 1981.
[25] R. Jain and S. A. Routhier, "Packet trains—measurements and a new model for computer network traffic," *IEEE J. Select. Areas Commun.*, vol. 4, pp. 986–995, Sept. 1986.
[26] Y.-C. Jenq, "Performance analysis of a packet switch based on single-buffered banyan network," *IEEE J. Select. Areas Commun.*, vol. 1, pp. 1014–1021, Dec. 1983.
[27] F. Kamoun and M. M. Ali, "Queueing analysis of ATM tandem queues with correlated arrivals," in *Proc. IEEE Infocom'95*, pp. 706–716.
[28] P. Kermani and L. Kleinrock, "Virtual cut-through: A new computer communication switching technique," *Comput. Network*, vol. 3, pp. 267–286, 1979.
[29] H. S. Kim and A. Leon-Garcia, "A self-routing multistage switching network for broadband ISDN," *IEEE J. Select. Areas Commun.*, vol. 8, pp. 459–466, Apr. 1990.
[30] ——, "Performance of buffered banyan networks under nonuniform traffic patterns," *IEEE Trans. Commun.*, vol. 38, pp. 648–658, May 1990.
[31] L. Kleinrock, *Queueing Systems Volume 1: Theory*. New York: Wiley, 1975.
[32] A. Krishna and B. Hajek, "Performance of shuffle-like switching networks with deflection," in *Proc. IEEE Infocom'90*, pp. 473–480.
[33] S. S. Lavenberg, *Computer Performance Modeling Handbook*. New York: Academic, 1983.
[34] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson, "On the self-similar nature of Ethernet traffic (Extended version)," *IEEE/ACM Trans. Networking*, vol. 2, pp. 1–15, Mar. 1994.
[35] B. Li and A. Ganz, "Virtual topologies for WDM star LANs—The regular structures approach," in *Proc. IEEE Infocom'92*, May, pp. 2134–2143.
[36] S.-Q. Li, "Performance of a nonblocking space-division packet switch with correlated input traffic," in *IEEE Globecom'89*, pp. 1754–1763.
[37] S. Madhavapeddy, K. Basu, and A. Roberts, "Adaptive paging algorithm for cellular systems," in *45th Vehicular Technology Conf.*, 1995, pp. 976–980.
[38] N. F. Maxemchuck, "The Manhattan street network," in *IEEE Globecom'85*, New Orleans, LA, Sept. 1985, pp. 255–261.
[39] N. F. Maxemchuck, "Comparison of deflection and store-and-forward techniques in the Manhattan street network and shuffle-exchange networks," in *IEEE Infocom'89*, Apr., pp. 800–809.
[40] R. M. Metcalfe and D. R. Boggs, "Ethernet: Distributed packet switching for local computer networks," *Commun. ACM*, vol. 19, no. 7, pp. 395–404, July 1976.
[41] J. A. Morrison, "A combinatorial lemma and its application to concentrating trees of discrete-time queues," *Bell Syst. Tech. J.*, vol. 57, no. 5, pp. 1645–1652, May-June 1978.
[42] J. A. Morrison, "Two discrete-time queues in tandem," *IEEE Trans. Commun.*, vol. 27, pp. 563–573, Mar. 1979.
[43] B. Mukherjee, "WDM-based local lightwave networks—Part II: Multihop systems," *IEEE Network*, pp. 20–32, July 1992.
[44] Y. Munn and H. Y. Youn, "Performance analysis of finite buffered multistage interconnection networks," *IEEE Trans. Comput.*, vol. 43, pp. 153–162, Feb. 1994.
[45] R. M. Newman, Z. L. Budrikis, and J. L. Hullet, "The QPSX man," *IEEE Commun. Mag.*, vol. 26, pp. 20–28, Apr. 1988.
[46] E. Noel and K. W. Tang, "Multihop networks: Performance modeling under nonuniform traffic patterns," in *Proc. IEEE Int. Performance, Computing and Communications Conf.*, 1997, Best paper award, pp. 563–571.
[47] I. Norros, "On the use of fractional Brownian motion in the theory of connectionless networks," *IEEE J. Select. Areas Commun.*, vol. 13, pp. 953–962, Aug. 1995.
[48] D. A. Reed, "The performance of multicomputer interconnection networks," *IEEE Comput.*, pp. 63–73, June 1987.
[49] T. G. Robertazzi and H.-Y. Huang, "Performance evaluation of the Manhattan street network with input buffers," in *Proc. 1992 IEEE Int. Conf. Communications*, 1992, pp. 202–206.
[50] F. E. Ross, "An overview of FDDI: The fiber distributed interface," *IEEE J. Select. Areas Commun.*, vol. 7, pp. 1043–1051, Sept. 1989.
[51] T. Sakamoto, E. Kamagata, and M. Serizawa, "Local registration and paging for in-building personal multi-media communication systems," in *46th Vehicular Technology Conf.*, 1996, pp. 1878–11 882.
[52] M. Schwartz, *Telecommunication Networks: Protocols, Modeling, and Analysis*. Reading, MA: Addison-Wesley, 1987.
[53] H. S. Stone, "Parallel processing with the perfect shuffle," *IEEE Trans. Comput.*, vol. c-20, pp. 153–161, Feb. 1971.
[54] K. W. Tang, "BanyanNet: A bidirectional equivalent of shufflenet," *J. Lightwave Technol.*, vol. 12, pp. 2023–2031, Nov. 1994.
[55] K. W. Tang and S. A. Padubidri, "Diagonal and toroidal mesh networks," *IEEE Trans. Comput.*, vol. 43, pp. 815–826, July 1994.
[56] S. W. Turner, "Performance analysis of multiprocessor interconnection networks using a burst-traffic model," Ph.D. dissertation, State Univ. Illinois, Urbana-Champaign, IL, 1995.
[57] A. M. Viterbi, "Approximate analysis of time-synchronous packet networks," *IEEE J. Select. Areas Commun.*, vol. 4, pp. 879–890, Sept. 1986.
[58] W. Willinger, M. S. Taqqu, R. Sherman, and D. V. Wilson, "Self-similarity through high-variability: Statistical analysis of Ethernet LAN traffic at the source level," *IEEE/ACM Trans. Networking*, vol. 5, pp. 71–86, Feb. 1997.



Eric Noel received the B.S. and M.S. degrees in electrical engineering from the State University of New York, Stony Brook, where he is currently working toward the Ph.D. degree.

He is a Senior Technical Staff Member with AT&T Network Computing and Services, Middletown, NJ, where he is responsible for performance analysis of various AT&T services and network elements. His research interests include performance modeling of data networks and computer systems.

K. Wendy Tang (M'91) received the B.S., M.S., and Ph.D. degrees in electrical engineering from the University of Rochester, Rochester, NY.

She is currently an Associate Professor in the Department of Electrical and Computer Engineering, State University of New York, Stony Brook. Her teaching and research interests include interconnection networks and neural networks.

She was a finalist of the 1994 Eta Kappa Nu Outstanding Young Electrical Engineer Award and a recipient of the 1998 IEEE Region 1 Award, 1998 IEEE Regional Activity Board Achievement Award, and the IEEE Millennium Award in 1999.