MULTIHOP NETWORKS: PERFORMANCE MODELING UNDER NON-UNIFORM TRAFFIC PATTERNS

Eric Noel

AT&T Laboratories
Holmdel, NJ 07733
ericn@surya.att.com

K. Wendy Tang

Department of Electrical Engineering
SUNY at Stony Brook, Stony Brook, NY 11794-2350
wtang@sbee.sunysb.edu

ABSTRACT

Performance modeling under non-uniform traffic is a useful tool to validate simulation accuracy and lend insights to realistic implementation of multihop networks. In this article, we present a performance model capable of tracking non-uniform traffic for an arbitrary multihop network. Our model is a an extension of Greenberg-Goodman and Brassil-Cruz models which are limited to Manhattan Street networks. By considering packets with non-null states only, we also provide an improved computational efficiency. Application to estimate the performance of Toroidal Mesh and Diagonal Mesh networks under non-uniform traffic is provided. Comparison of performance parameters derived from the model and from simulation show good agreement. Limitations of our memoryless and independence assumptions based model are addressed.

I. INTRODUCTION

Multihop networks have found applications as wavelength division multiplexed lightwave networks [12] and as interconnection networks for multicomputers [13]. 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 number of multihop networks have been proposed. This include the Manhattan Street [11], the ShuffleNet [15], BanyanNet [16], Toroidal Mesh, and Diagonal Mesh [17].

Because of its simplicity, *deflection routing* or *hot potato routing* [2] 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 non-optimal 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 algorithms reduce the maximum delay [14, 17].

Most literature on multihop networks contains only simulation results to estimate the networks performance. If performance analysis is included, it is often for *uniform* traffic only. While computer simulation is often prone to programming errors, uniform traffic pattern is far from the realistic situation. There is, therefore, a need for a *non-uniform* traffic performance model that can be applied to multihop networks of arbitrary topology. Such a model will validate the simulation results and will lend insights to the capability of the proposed multihop network.

Already, Greenberg-Goodman and Brassil-Cruz [8, 4, 7] have developed a performance model for packet arrivals subject to the independence and memoryless assumptions, for non-uniform traffic, and limited to Manhattan Street networks using deflection routing strategy. In our paper we generalize this model for arbitrary network topologies with an improved computational efficiency. To demonstrate the validity of our model, we apply it to two kinds of non-uniform traffic profile, the *single node accumulation* and *single node broadcast*, for 9x11 Toroidal and Diagonal Mesh networks.

These two non-uniform traffic patterns are representative of multicomputer behavior for a wide range of parallel algorithms such as inner product calculations and relaxation iterations [3]. However, we concede that our packet arrival model (limited by the independence and memoryless assumptions) do not represent bursty arrivals which is more common in communication networks [6]. Performance modeling for bursty arrivals is not a trivial problem. Due to their correlated and time-variant nature (a violation of the independence and memoryless assumptions), probabilistic model of these arrivals will result in an exponential growth in the number of states and is computationally prohibitive even for networks of small size. For this reason and to the best of our knowledge, probabilistic network performance models have not been applied to bursty arrivals in the literature.

Comparison of parameters derived from the model with simulation results showed good agreement. Furthermore, by considering only packets with non-null state, we improved the computational efficiency of Greenberg-Goodman and Brassil-Cruz models. For example, under single node accumulation for both networks, we estimated that our model's time complexity is 26 time better; whereas for single node broadcast, our model's time complexity is 3×10^8 times better.

This article is organized as follows: In Section 2, we present the performance model. Applications of the model to Toroidal and Diagonal Mesh networks are included in Section 3. Section 4 discusses the limitations and the complexity analysis of the model. Finally, in Section 5, we present a conclusion.

II. PERFORMANCE MODEL

A. Network Model

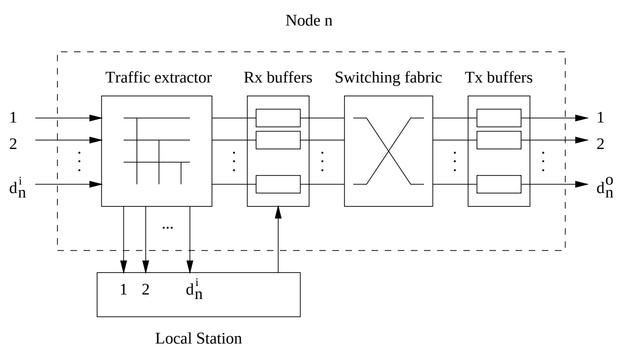

Figure 1: Node model.

We model the multihop network as a set of nodes connected by zero delay links. As shown in Figure 1, each node consists of a traffic extractor, d_n^i receive buffers (Rx buffers), a switching fabric, and d_n^o transmit buffers (Tx buffers). 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. All buffers are size one, so there is no queueing of packets. Time is synchronized so that all nodes receive and transmit packets simultaneously.

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-arrival times). The 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 and lost.

The switching fabric assigns to each received packet a set of preferred outgoing links based on the shortest path [5] 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 non empty 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 non empty set of preferred outgoing links have been serviced, packets with 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 packets 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 1 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. Then, applying the total probability theorem we obtain,

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

where $\mathcal{A}_k = \{a_1, a_2, ...a_x, ..., a_k\}$ represents all possible packet age combinations, $\mathcal{D}_k = \{d_1, d_2, ..., d_x, ...,$

N	Total number of nodes in the network under test.
$\{d_j, a_j\}$	j^{th} packet in node n receive buffer, of destination node d_j and age a_j.
$p^i_{n,l}(d;a)$	Probability that 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 that packet $\{d,a\}$ *leaves* node n on link l in the next time slot. $p^o_{n,l}(n;a)$, the probability packet $\{n,a\}$ leaves node n, is always zero.
$p_n(E_k)$	Probability $\{d_1,a_1\},\{d_2,a_2\},...,\{d_k,a_k\}$ entered node n Rx buffers.
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 to 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 $a_j > a_{j-1}... > a_x > ... > a_k$.

Table 1: Nomenclature of parameters used in the model for steady-state probabilities.

$d_k\}$ represents all possible packet destination combinations, and $\mathcal{L}_k = \{l_1, l_2, ..., l_x, ..., l_k\}$ represents all packet incoming link combinations.

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

More precisely, let node n receive buffers be occupied by k packets $\{d_1,a_1\}$, ...,$\{d_x,a_x\}$, ...,$\{d_k,a_k\}$ of respective set of preferred outgoing links $\mathcal{P}_1$, $\mathcal{P}_2$, ..., $\mathcal{P}_k$ and such that the age of these packets are sorted with $\{d_1,a_1\}$ being the oldest packet, i.e. $a_1 > a_2 > ... > a_x > ... > a_k$. Also, define for each packet j, D_j as the deflection set (set of outgoing links 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 $j = 1, 2..., x-1$ we define the function $S^x(j|\mathcal{P}_j, ..., \mathcal{P}_x, ..., \mathcal{P}_k)$ by:

- $S^x(j|\mathcal{P}_j, ...) = S^x(j+1|\mathcal{P}_{j+1}, ...)$ if $(\mathcal{P}_j = \emptyset)$ or $(C_j \neq \emptyset \wedge l_x \notin C_j)$: If packet $\{d_j,a_j\}$ has an empty set of preferred outgoing links or does not prefer link l_x, we set to one its probability of being assigned link l_x, prior packet $\{d_x,a_x\}$ is assigned an outgoing link.
- $S^x(j|\mathcal{P}_j, ...) = (1 - 1/|C_j|)S^x(j+1|\mathcal{P}_{j+1}, ...)$ if $l_x \in C_j$: If C_j contains l_x, 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.
- $S^x(j|\mathcal{P}_j, ...) = \sum_{t=1}^{|D_j|-1}(1-1/|D_j|)S^x(j+1|\mathcal{P}^t_{j+1}, ...)$ if $C_j = \emptyset \wedge l_x \in D_j$: If D_j contains l_x, there are $|D_j|-1$ possible ways that packet $\{d_j,a_j\}$ is assigned an outgoing link and affects the set of preferred outgoing links of one or more younger packets.
- $S^x(j|\mathcal{P}_j, ...) = \sum_{t=1}^{|D_j|} 1/|D_j| S^x(j+1|\mathcal{P}^t_{j+1}, ...)$ if $C_j = \emptyset \wedge l_x \notin D_j$: If D_j does not contain l_x, there are $|D_j|$ possible ways that packet $\{d_j,a_j\}$ is assigned an outgoing link and affects the set of preferred outgoing links of one or more younger packets.

When packet $\{d_j,a_j\}$ is assigned the t^{th} link of its deflection set D_j, we remove the assigned link from the set of preferred outgoing links of all younger packets (operation denoted by $\mathcal{P}^t_\alpha$).

The last term of the recursive function corresponds to $j = x$:

- $S^x(x|\mathcal{P}_x, ..., \mathcal{P}_k) = 1/|C_x|$ if $l_x \in C_x$: When C_x is non empty and contains l_x, the probability that l_x is assigned to packet $\{d_x, a_x\}$ is one over the size of set C_x.

- $S^x(x|\mathcal{P}_x, ..., \mathcal{P}_k) = 1/|D_x|$ if $C_x = \emptyset \wedge l_x \in D_x$: 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.

- $S^x(x|\mathcal{P}_x, ..., \mathcal{P}_k) = 1/(u - O_x) \prod_{i=0}^{|O_x|-1}(1 - 1/(u - i))$ if $\mathcal{P}_x = \emptyset$: When $\mathcal{P}_x$ is empty, 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. (u is the number of remaining outgoing links.)

So, assuming packet arrival 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 $S^x(1 \mid \mathcal{P}_1, \mathcal{P}_2, ..., \mathcal{P}_x, ..., \mathcal{P}_k)$. When twin packets are present, to calculate the conditional probability in (1), we calculate $S^x(1 \mid \mathcal{P}_1, \mathcal{P}_2, ..., \mathcal{P}_x, ..., \mathcal{P}_k)$ for each twin packet permutation, and take the average (twin packets are randomly sorted).

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

$$\Pr\begin{bmatrix} k \text{ packets} \\ \text{in receive} \\ \text{buffer} \end{bmatrix} = \prod_{j=1}^{k} p^i_{n,l_j}(d_j, a_j)$$
$$\prod_{j^*=k+1}^{d^i_n}\left(1 - \sum_{\mathcal{A}^*_k, \mathcal{D}^*_k, \mathcal{L}^*_k} p^i_{n,l^*_{j^*}}(d^*_{j^*}, a^*_{j^*})\right)$$
$$\left(1 - \sum_{d=0, d\neq n}^{N-1} p^i_{n,0}(d;0)\right),$$

where $\mathcal{A}^*_k = \{a^*_{k+1}, a^*_{k+2}, ..., a^*_{d^i_n}\}$ represents all $(d^i_n - k)$ sets such that $a^*_j \in \{1, ..., A-1\}$ for $j \in [k+1, d^i_n]$. $\mathcal{D}^*_k = \{d^*_{k+1}, d^*_{k+2}, ..., d^*_{d^i_n}\}$ represents all $(d^i_n - k)$ sets such that $d^*_j \in \{0, 1, ..., N-1\} \setminus \{n\}$ for $j \in [k+1, d^i_n]$. And $\mathcal{L}^*_k = \{l^*_{k+1}, l^*_{k+2}, ..., l^*_{d^i_n}\}$ represents all $(d^i_n - k)$ permutations of $\{1, 2, ..., d^i_n\} \setminus \{l_1, l_2, ..., l_k\}$.

That is, the first term 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 created ($\exists j \in [1, k] | a_j = 0$), the third term is removed.

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*, and the *specified accuracy*.

To describe the steps followed in our model implementation, we consider a node with non-empty receive buffers at iteration t. Then, using (1), we compute the probability (p^o) for each received packet (of non-zero state probability) to enter every neighboring node in the next iteration. Next, we send the received packets to every neighboring node for which p^o is non zero. 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.

We illustrate this procedure with the example represented in Figure 2 for a 3x3 Unidirectional Toroidal Mesh network (for clarity, we do not use the most recently updated receive buffers). In Figure 2, packets entering the same receive buffer are shown on the same row, and packets entering different receive buffers are shown on different rows. Starting with an empty network (column labeled "Iteration 1" in Figure 2), both nodes 0 and 3 send a packet destined to node 4 with probability one. Since there are two routes equally distant from node 0 to node 4, the packet sent by node 0 is multiplied into two equally probable packets, one to neighboring node 1 and the other to neighboring node 3. Similarly, because there is only one shortest route from node 3 to node 4, the packet sent by node 3 is forwarded to neighboring node 5 with probability one. The remaining iterations should be interpreted similarly.

D. Convergence

It is easy to see that solving for the network output probabilities is equivalent to finding the fixed point of a multi-dimensional function which maps a set of output probabilities from one iteration to the next (combine (1) with the mapping of input probabilities to output probabilities). Conditions for the existence of a fixed point may be found in [9]. However, whether these conditions applied to our multi-dimensional function remains to be verified.

Qualitatively, convergence is achieved once our iterative procedure reaches steady state. Proof that there always exist a unique steady state remains to

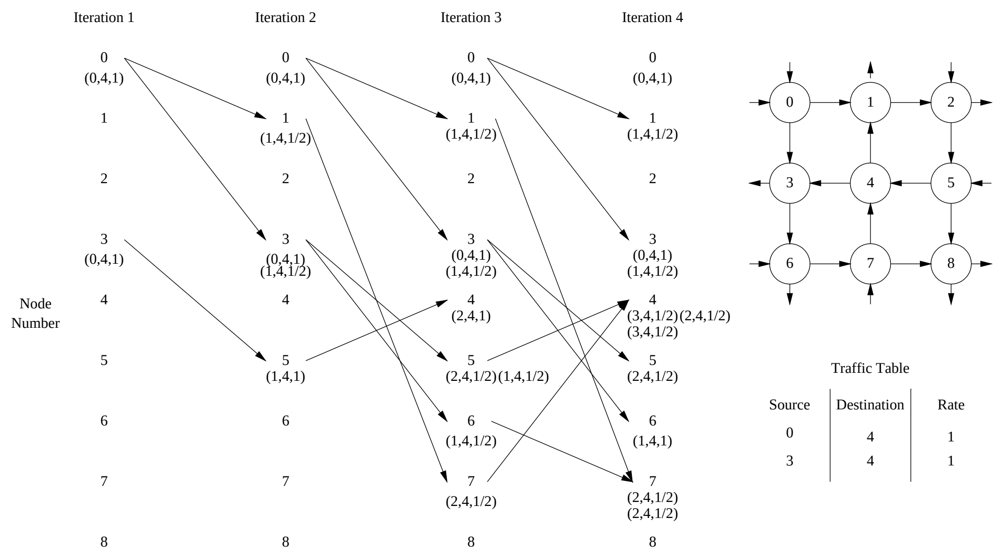

Figure 2: Illustration with a 3x3 Unidirectional Toroidal Mesh network. (Each packet is represented by a triplet (a, d, p), where a: age, d: destination, p: input probability.)

be found. Experience with the model shows that a steady state is always reached within a number of iterations approximately equal to maximum packet age.

E. Performance Parameters

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 $h(a)$ (the probability that a packet arrives to its destination node in a hops), the mean delay μ, 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:

$$
\begin{aligned}
p_b &= 1 - \frac{\sum_n \sum_l \sum_a p^i_{n,l}(n;a)}{\sum_n \sum_d p^i_{n,0}(d;0)} && (2)\\
h(a) &= \frac{\sum_n \sum_l p^i_{n,l}(n,a)}{\sum_n \sum_l \sum_a p^i_{n,l}(n,a)} && (3)\\
\mu &= \textstyle\sum_a a h(a) && (4)\\
U_n^o(l) &= \textstyle\sum_d \sum_a p^o_{n,l}(d;a) && (5)\\
U_n^i(l) &= \textstyle\sum_d \sum_a p^i_{n,l}(d;a) && (6)\\
R_n^o &= \textstyle\sum_l U_n^o(l) && (7)\\
R_n^i &= \textstyle\sum_l U_n^i(l) && (8)
\end{aligned}
$$

III. APPLICATION

A. Simulation

To validate the performance model, we constructed an event-driven simulator from which we derived the same performance parameters derived from the model. The simulator replicated all aspects of our network model. For each traffic and network type, we made 10 independent replications of 150,000 departures each (statistics associated with the first 50,000 departures were disregarded). The resulting performance parameters were averaged over the 10 replications and a 95% confidence interval was constructed by assuming the normalized error to be t-distributed [10]. We set our model convergence error to 10^{-6} and the age bound $A = 20$.

B. Multihop Networks Under Test

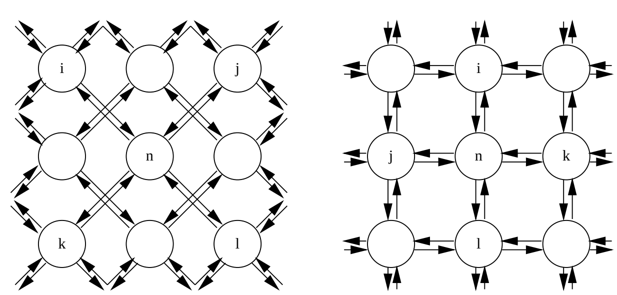

Figure 3: Node n connected neighbors for a Diagonal Mesh network (left) and a Toroidal Mesh network (right).

The two networks under test were a 9 rows by 11 columns (9x11) Toroidal Mesh [1] and a 9x11 Diagonal

[17] Mesh subject to two non-uniform traffic patterns: single node accumulation traffic (all nodes transmit to a single node) and single node broadcast (a single node transmits to all nodes). Such traffic patterns are representative of multicomputers behavior for a wide range of parallel algorithms [3].

Both networks are bi-directional degree four multihop networks. We only consider odd numbered Diagonal Mesh networks, as even numbered networks do not create a fully connected graph. The diameter of an R rows by C columns (RxC) Diagonal Mesh network (D_d) and an RxC Toroidal Mesh network (D_t) are [17]: (Assuming, without loss of generality $C \geq R$.)

$$D_d = \begin{cases} \max(R, \frac{C-1}{2}) & \text{if } C > R \\ R-1 & \text{if } C = R, \end{cases}$$

$$D_t = \frac{C-1}{2} + \frac{R-1}{2}.$$

For a R rows and C columns network, we number nodes from 0 to $N-1$ ($N = RC$) left to right, top to bottom.

C. 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 [3]. The networks tested were 9x11, and the node accumulation was node 49.

In Figure 4, we show the outgoing and incoming packet rate derived from our model (Equations 7 and 8). In Figure 5, we compare the model and the simulation delay histograms (Equation 3). In Table 2, we compare the model and the simulation mean delay, blocking probability and outgoing link utilization (Equations 4, 2, 5). We found good agreement between the model and simulation for the mean delay (μ), and outgoing link utilization. For the blocking probability (p_b), our model predicts a value 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.

Moreover, we found little delay performance differences between the two networks under test. This can be explained by noticing that the diameter of the Diagonal Mesh and Toroidal Mesh networks are equal for 9x11 ($D_t = D_d = 9$, see Section 3.1). Delay differences in favor of the Diagonal Mesh network would become more significant for $C > R + 2$ [17].

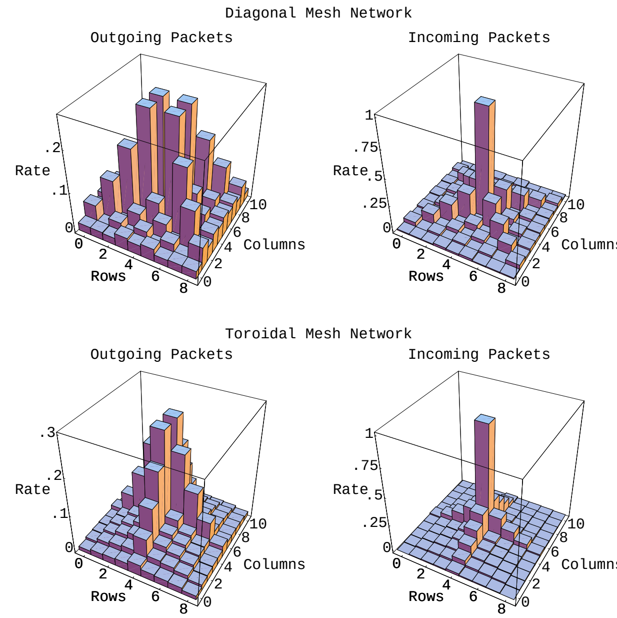

Figure 4: Outgoing and incoming packet rate for a 9x11 Diagonal Mesh network and a 9x11 Toroidal Mesh network subject to single node accumulation traffic.

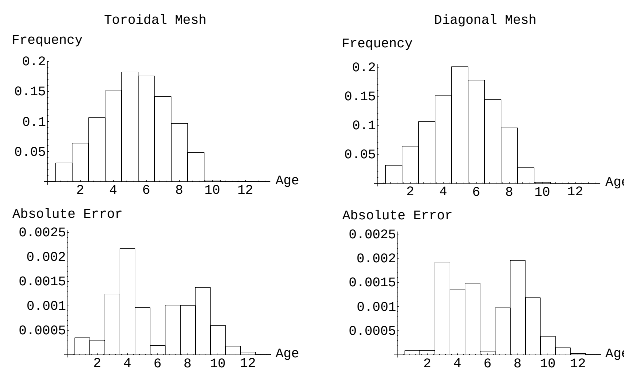

Figure 5: 9x11 Toroidal Network and 9x11 Diagonal Network delay histograms and absolute error histograms for single node accumulation traffic.

D. Single Node Broadcast

The single node broadcast traffic pattern corresponds to one node transmits to all other nodes at a relative rate of $1/(N-1)$. This traffic pattern corresponds to the case where all nodes of a multiprocessor system receive messages from a single node, as found in applications such as inner product calculations [3].

The networks tested were 9x11, and the broadcasting node was node 49. Outgoing and incoming packet rate are symmetrical to the one for the single node accumulation traffic pattern shown in Figure 4. Model

Single Node Accumulation					
Network	μ		p_b		Maximum Outgoing Link Utilization Error
	Model	Simulation	Model	Simulation	
Diagonal Mesh	5.191	5.212±0.004	$\leq 10^{-6}$	0	0.002
Toroidal Mesh	5.277	5.297±0.005	$\leq 10^{-6}$	0	0.001

Single Node Broadcast					
Network	μ		p_b		Maximum Outgoing Link Utilization Error
	Model	Simulation	Model	Simulation	
Diagonal Mesh	4.927	4.919±0.007	$\leq 10^{-6}$	0	0.002
Toroidal Mesh	5.001	5.000±0.005	$\leq 10^{-6}$	0	0.002

Table 2: Comparison between the model and simulation under single node accumulation traffic and single node broadcast traffic for mean delay (μ), blocking (p_b), and maximum outgoing link utilization error.

and simulation histograms are equivalent to the ones for the single node accumulation traffic pattern shown in Figure 5. In Table 2, we compare the model and the simulation mean delay, blocking probability and outgoing link utilization. We found good agreement between the model and simulation for the mean delay (μ), and outgoing link utilization. For the blocking probability (p_b), our model predicts a value 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.

Again, for such small networks, there are not much differences in the average delay between the two networks.

IV. MODEL COMPLEXITY AND LIMITATIONS

A. Complexity

We use the *time complexity* to quantify our model complexity. We define it as the number of terms generated by the summations in (1) for every packet, and node at each iteration. This is equivalent to the number of loops used in the model implementation to compute the output probabilities of every node, and for each iteration.

The time complexity of our model is a function of traffic characteristics. Consequently, we can only provide an upper bound for the time complexity, derived from the *direct implementation* (implementation which does not exclude packets with null states).

The time complexity upper bound corresponds to the number of possible combinations to have k packets ($k \in [1, d_n^i]$) in each node receive buffer of age ranging from 0 to $A-1$ ($(A-1)^k$ combinations), of destination ranging over the D destination nodes (D^k combinations), and entering in each node from any of the d^i input links (k choose d^i combinations). That is,

$$N \sum_{k=1}^{d_n^i} \binom{d_n^i}{k} (A-1)^k D^k = N(DA)^{d^i},$$

where $d^i = d_n^i$ for all $n \in [0, N-1]$.

To quantify the computational efficiency of our implementation, we compare time complexity of our implementation to the one for the direct implementation for the networks and traffic profiles described in Section III. The time complexity of our implementation is calculated by computing the number of ways to combine the packets with non-null state into the receive buffers of each node, during the last iteration (worst case). The time complexity of the direct implementation is given in the next table.

	Time Complexity	
	Single Node Accumulation	Single Node Broadcast
Direct Implementation	15.8x10^6	1.5x10^{15}
Section C Implementation	0.6x10^6	4.5x10^6

For a 9x11 Diagonal Mesh or Toroidal Mesh network, our implementation time complexity is approximatively 26 times better than the one for direct implementation when subject to single node accumulation traffic, and is approximatively 3x10^8 times better for single node broadcast traffic. Even though in the direct implementation, no operations are performed every time a combination of incoming packets has one or more packet with null state, the penalty in terms of extra loops is significant.

B. Limitations

Using a packet arrival model which complies with the memoryless and independence assumptions allowed us to derive tractable performance expressions. Even

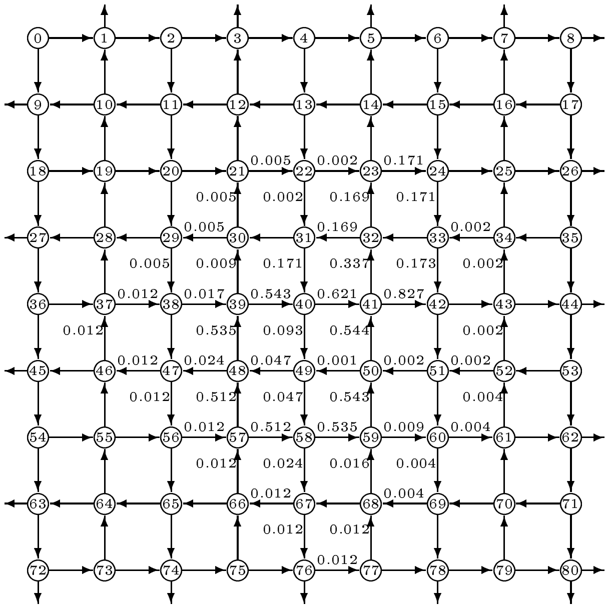

Figure 6: Link utilizations greater than 10^{-4} derived from the model for a 9x9 Unidirectional Toroidal Mesh network subject to a streaming source traffic [4]. (Node 57 sends packets continuously to node 42.)

though our traffic model does not account for long range dependencies, as with bursty traffic [6], insights on performance characteristics of multihop networks subject to non uniform traffic patterns can be gained.

As mentioned, in the development of our model, we assumed packet arrivals to the same node to be independent of one another (independence assumption) and of the state of the neighboring nodes (memoryless assumption). The effects of our independence and memoryless assumptions become clear when relating to the model implementation. Each time a packet has more than one preferred outgoing link, our model allows for the packet to be *multiplied* (one instance to each preferred outgoing link, contending packets permitting). As long as such multiplied packets do not interfere (enter the same node) in a later iteration (that is, as long as the memoryless assumption is not violated), the model remains accurate. However, as soon as such packets interfere with one another, the model accuracy degrades. We illustrate such degradation with Figure 6 for a 9x9 Unidirectional Toroidal Mesh network subject to the streaming source traffic pattern [4]. Upon exiting node 57, packets originated from node 57 are multiplied and meet again in node 41. As a result, the model accounts for deflections which cannot occur, since the source cannot send more than one packet at a time.

Such observation leads us to define the metric $\mathcal{M}$ for evaluating how often the independence and memoryless assumptions are violated.

$$\mathcal{M} \quad = \quad \sum_{n=0}^{N} \sum_{a=0}^{A} \sum_{k=2}^{d_n^i} \Pr \left[\begin{array}{c} k \text{ multiplied packets} \\ \text{of age } a \text{ enter the} \\ \text{same node in the} \\ \text{next time slot} \end{array} \right].$$

In other words, $\mathcal{M}$ represents the frequency that our model allows for multiplied packets to enter the same node in any time slot. So, the smaller $\mathcal{M}$ is, the more accurate is our model.

For the 9x9 Unidirectional Toroidal Mesh network (UTM) subject to the streaming source traffic (Figure 6), the 9x11 Toroidal Mesh network (TM) and the 9x11 Diagonal Mesh network (DM) subject to single node accumulation and single node broadcast traffic (Section 3), we constructed the following table:

Network	Traffic Pattern	$\mathcal{M}$	Max. Outgoing Link Utilization Error
UTM	Streaming Source	0.250	0.336
TM	Single Node Accumulation	≤ 0.073	0.001
DM	Single Node Accumulation	≤ 0.076	0.002
TM	Single Node Broadcast	0.001	0.002
DM	Single Node Broadcast	0.001	0.002

In this table, we have included the different types of networks under the three traffic patterns, the value for the metric $\mathcal{M}$ and the corresponding maximum outgoing link utilization error between the model and simulation.

We can observe that for the streaming source traffic pattern in the 9x9 Manhattan Street network, the $\mathcal{M}$ metric is quite large, $\mathcal{M} = 0.25$ which accounts for the relatively large maximum outgoing link utilization error between simulation and results from the model. However, for the single node accumulation and the single node broadcast traffic patterns, the $\mathcal{M}$ values are quite small for both DM and TM networks which explain why the model and simulation results agree, as evident by the small maximum outgoing link utilization error. Simply stated, this table confirms that the smaller $\mathcal{M}$ is, the more accurate is our model.

Note that in this table, for the single node accumulation traffic profile, we have only provided an upper bound for the metric $\mathcal{M}$. This is because our implementation allows us to calculate $\mathcal{M}$ only when packets from distinct source nodes have distinct destination nodes. In this case, the upper bound on $\mathcal{M}$ is simply the probability of two or more packets of the same age entering the same node in the next time slot.

V. CONCLUSION

In this article, we presented a performance model for multihop networks under non-uniform traffic pattern. The model is a generalization of Greenberg-Goodman and Brassil-Cruz models which were designed specifically for Manhattan Street networks [8, 4, 7]. Our model, on the other hand, can be applied to an arbitrary network topology of arbitrary degree. Furthermore, by considering packets with non-null states only, our model is computationally more efficient than Greenberg-Goodman and Brassil-Cruz direct implementations.

As an application, we applied the model for performance evaluation of 9x11 Toroidal and Diagonal Mesh networks subject to (i) single node accumulation and (ii) single node broadcast traffic patterns. These two traffic profiles are chosen not only because of their relative tractability but also because they represent a wide range of problems in multicomputer networks. We found the model provides good agreement with simulation. Not surprisingly, for these small networks of same diameter, there are no significant difference in the average delay between the two networks.

Finally, we also discussed various issues related to the model implementation, including complexity, limitations and convergence. By incorporating event-driven simulation and considering packets with non-null states only, our model is more time efficient. For example, with the 9x11 Diagonal and Toroidal Mesh networks, our model provides several order of magnitude improvement over the Greenberg-Goodman and Brassil-Cruz implementations in time complexity.

In terms of model limitations, both our model and Greenberg-Goodman and Brassil-Cruz models made the *independence and memoryless assumptions* which imply packet arrivals to the same node are independent of one another and of the state of neighboring nodes. We briefly discussed when such an assumption is not valid and defined a metric to assess the model accuracy.

VI. REFERENCES

[1] W. C. Athas, C. L. Seitz, *Multicomputers: Message-Passing Concurrent Computers.* IEEE Computer Magazine, August 1988, pp. 9-24.

[2] P. Baran, *On distributed computing networks*, IEEE Transactions on Communications Systems. Mar 1964, pp. 1-9.

[3] D. P. Bertsekas, J. N. Tsitsiklis, *Parallel and Distributed Computation Numerical Methods.* Englewood Cliffs, NJ: Prentice Hall, 1989.

[4] J. Brassil, R. Cruz. *Nonuniform Traffic in the Manhattan Street Network.* Proceedings of the 1991 IEEE International Conference on Communications, pp.1647-1651.

[5] E. W. Dijkstra, *A Note on Two Problems in Connexion with Graphs.* Numerische Mathematik 1, pp.269-271, 1959.

[6] V. S. Frost, B. Melamed, *Traffic Modeling For Telecommunications Networks.* IEEE Communications Magazine, pp. 70-81, March 1994.

[7] A. G. Greenberg, J. Goodman, *Sharp Approximate Models of Deflection Routing in Mesh Networks.* IEEE Transactions on Communications, vol. 41, no. 1, pp. 210-223, June 1992.

[8] A. G. Greenberg, J. Goodman, *Sharp Approximation of Adaptive Routing in Mesh Networks.* Teletraffic Analysis and Computer Performance Evaluation, O. J. Boxma, J. W. Cohen, H. C. Tijms (Editors), Elsevier Science Publishers B.V. (North Holland), 1986.

[9] V. I. Istăţescu, *Fixed Point Theory An Introduction.* D. Reidel Publishing Company, 1981.

[10] S. S. Lavenberg, *Computer Performance Modeling Handbook.* Academic Press, 1983.

[11] N. F. Maxemchuck, *The Manhattan Street Network.* IEEE Globecom'85, New-Orleans, September 1985, pp. 255-261.

[12] B. Mukherjee, *WDM-Based Local Lightwave Networks Part II: Multihop Systems.* IEEE Network, July 1992, pp. 20-32.

[13] D. A. Reed, *The Performance of Multicomputer Interconnection Networks.* IEEE Computer June 1987, pp. 63-73.

[14] H.-Y. Huang and T. Robertazzi and A. A. Lazar, *A Comparison of Information Based Deflection Strategies.* Computer Networks and ISDN Systems, 1995,vol. 27, pp. 1388-1407.

[15] H. S. Stone, *Parallel Processing with the Perfect Shuffle.* IEEE Transactions on Computers, vol. c-20, no. 2, pp. 153-161, February 1971.

[16] K. W. Tang, *BanyanNet: A Bidirectional Equivalent of ShuffleNet.* Journal of Lightwave Technology, vol. 12, no. 11, November 1994, pp. 2023-2031.

[17] K. W. Tang, S. A. Padubidri, *Diagonal and Toroidal Mesh Networks.* IEEE Transactions on Computers, vol 43, no. 7, July 1994, pp. 815-826.