A More Accurate Analytical Model on Blocking Probability of Multicast Networks

Yuanyuan Yang and Jianchao Wang

Abstract—*Multicast* communication is one of the most important collective communication operations and is highly demanded in telecommunication environments and scalable parallel and distributed computing systems. In this paper, we consider the issue of supporting multicast in the widely used three-stage Clos network or $v(m, n, r)$ network. We improve a recently proposed analytical model [7] for the blocking probability of the $v(m, n, r)$ multicast network by introducing more reasonable assumptions based on the properties of multicast communication and the Clos network. We also compare the improved analytical model with the simulation results under three typical routing control strategies. As can be seen, the improved model matches better with the simulation results and further confirms that a $v(m, n, r)$ network with a comparable cost to a permutation network is almost nonblocking for multicast connections.

Index Terms—Blocking probability, multicast communication, performance analysis, routing algorithm.

I. INTRODUCTION

MULTICAST communication is one of the most important collective communication operations and is highly demanded in telecommunication environments and scalable parallel and distributed computing systems. In this paper, we consider the issue of supporting multicast in the well-known three-stage Clos network [1]. Clos-type networks have been extensively studied for both unicast communication and multicast communication in the literature. For this type of network, it has been shown [4], [5] that a nonblocking multicast network requires a higher network cost than a permutation network [1]. Therefore, it is interesting to know the blocking behavior of the multicast network with only a comparable network cost to a permutation network.

Research in modeling the blocking behavior of multicast networks, in contrast to its unicast counterpart, is still in its early stage. Zegura [6] proposed an analytical model for a class of series-parallel multicast networks which include the Clos network. Zegura's model evaluates the blocking probability when adding a destination node to an existing multicast connection. In other words, the blocking probability is calculated from the source node to a single destination node of a multicast connection (i.e., point-to-point blocking probability). While this feature may be suitable for certain types of multicast applications, such as teleconferencing, it may not be realistic for other types of multicast applications, such as video-on-demand and record update in a distributed database. Later, Yang and Wang [7] proposed a different type of analytical model for the blocking probability of the Clos type multicast network. This model is concerned with that whether there exists a multicast tree from the source node to all the destination nodes of a new multicast connection. Clearly, this model is more suitable for other types of multicast applications, such as video-on-demand and record update in a distributed database. In [7], the blocking probability is calculated for the entire multicast tree (i.e., one-to-many blocking probability), which is generally considered as a challenging problem because that the enormous dependencies among individual paths in a multicast tree and/or among different multicast trees make an explicit calculation intractable. The analytical model was also compared with the simulation study under various routing control strategies in [7]. Both the analytical and simulation results show that a network with a comparable cost to a permutation network is almost nonblocking for multicast connections and thus can provide cost-effective support for multicast communication. However, as can be seen, there still exists some gap between the analytical model and the simulation results in [7].

In this paper, we are interested in improving the analytical model proposed in [7]. As will be seen later, we reduce the gap between the analytical model and the simulation results by introducing more reasonable assumptions based on the properties of multicast communication and the Clos network. The improved analytical model matches better with the simulation results and further confirms that a $v(m, n, r)$ network with a comparable cost to a permutation network is almost nonblocking for multicast connections.

The rest of the paper is organized as follows. In Section II, some background knowledge and previous work are discussed. In Section III, a more accurate analytical model for the blocking probability of the multicast networks is presented. Section IV compares the new model with the previous one as well as with the simulation results. Section V concludes the paper.

II. PRELIMINARIES AND PREVIOUS RELATED MULTICAST MODEL

As shown in Fig. 1, a three-stage Clos network or a $v(m, n, r)$ network has r $(n \times m)$ switches in the first stage (the input

Fig. 1. A general schematic of an $N \times N$ $v(m, n, r)$ network, where $N = nr$.

Fig. 2. The subnetwork associated with a multicast connection with fanout f. The dashed lines indicate the idle link subnetwork.

stage), m ($r \times r$) switches in the middle stage, and r ($m \times n$) switches in the third stage (the output stage). As two of the $v(m, n, r)$ network parameters, n and r, are restricted by the number of network input/output ports, the main focus of the study is to determine the minimum value of the network parameter m for a certain type of connecting capability to achieve the minimum network cost.

When the $v(m, n, r)$ network is considered for supporting multicast, it is reasonable to assume that every switch in the network has multicast capability. Since output switches have multicast capability, a *multicast connection* from an input port can be simply expressed in terms of output switches it connects to. The number of output switches in a multicast connection is referred to as the *fanout* of the multicast connection.

Several designs have been proposed for this type of multicast network [2]–[5]. It was shown [4], [5] that the necessary and sufficient condition for a $v(m, n, r)$ network to be *nonblocking* for arbitrary multicast connections under the best available routing control strategy is that the number of middle switches $m = \Theta(n \log r / \log \log r)$. This suggests that there is little room for further improvement on the nonblocking condition for multicast connections. However, as can be seen, there still exists a gap between the number of middle switches required for a nonblocking multicast network and that for a nonblocking permutation network which requires only $m \geq 2n - 1$ [1]. It would be interesting to know the blocking behavior of the multicast network when the nonblocking condition is not satisfied, especially when the number of middle switches is comparable to a permutation network. [7] presented an analytical model on the blocking probability of $v(m, n, r)$ multicast networks. This model suggests that when m gets slightly larger than n, such as $m \geq cn$, where $c (\geq 1)$ is a small constant, the $v(m, n, r)$ multicast network is almost nonblocking for multicast connections. This trend of the blocking behavior revealed by the analytical model was also verified by extensive experimental simulations in [7]. However, there still exists some disparity between the analytical model and the simulation results in [7].

Before we present the improved analytical model, we briefly review the previous analytical model on the blocking probability of $v(m, n, r)$ networks proposed in [7].
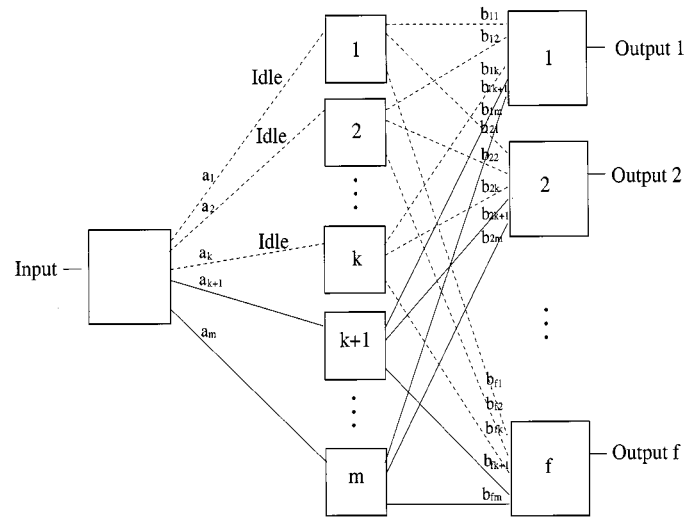
In general, determination of blocking probability in a multistage network (even for permutation networks) is inherently complex and difficult. This is due to the fact that there are many possible paths to consider in a typical large network, and the dependencies among links in the network lead to combinatorial explosion problems. Therefore, some assumptions are necessary to make the calculation possible.

The analytical model on the blocking probability of $v(m, n, r)$ multicast networks proposed in [7] is based on the following assumptions.

1) The events that individual links are busy are independent.
2) The incoming traffic is uniformly distributed over all the input-middle interstage links and the outgoing traffic is also uniformly distributed over all the middle-output interstage links.
3) All interstage links have the same probability of being busy.

Under the above assumptions, let the probability that a typical input (output) port is busy be a. Then the probability that an interstage link is busy is given by $p = an/m$, and the probability that an interstage link is idle is given by $q = 1 - p$.

Recall that a multicast connection is represented by the output switches it connects to. Given a multicast connection request with fanout f ($1 \leq f \leq r$), let $P_B(f)$ denote the probability that this connection request cannot be satisfied, that is, the blocking probability for this connection request. Fig. 2 depicts the subnetwork associated with this multicast connection, in which one input switch is linked to m middle switches, and each of the m middle switches is linked to f output switches. Denote the interstage links between the input switch and m middle switches as $a_1, a_2, \ldots, a_m$ (also referred to as input-middle interstage links), and the interstage links between the m middle switches and the kth ($1 \leq k \leq f$) output switches as $b_{k1}, b_{k2}, \ldots, b_{km}$ (also referred to as middle-output interstage links). All paths realizing the multicast connection in the network form a multicast tree rooted at the input.

Let ε be the event that the connection request with fanout f cannot be realized in the subnetwork in Fig. 2. Notice that any

interstage link is either busy or idle. Denote the event that the link a_i is busy as $\overline{a_i}$ and the event that the link a_i is idle as $\overline{\overline{a_i}}$ for $1 \leq i \leq m$. Let σ represent the state of the input-middle interstage links $a_1, a_2, \ldots, a_m$, $P(\varepsilon \mid \sigma)$ be the conditional blocking probability in this state, and $P(\sigma)$ be the probability of being in state σ. Then the *blocking probability for a multicast connection with fanout f* is given by

$$
\begin{aligned}
P_B(f) &= P(\varepsilon) \\
&= \sum_\sigma P(\sigma) P(\varepsilon \mid \sigma) \\
&= \sum_{k=0}^m \binom{m}{k} q^k p^{m-k} P(\varepsilon \mid \overline{a_1}, \ldots, \overline{a_k}, a_{k+1}, \ldots, a_m).
\end{aligned}
$$

$$(1)$$

The following lemma was used in [7] to calculate the blocking probability.

Lemma 1: Assume that the interstage links $a_1, a_2, \ldots, a_k$ in the subnetwork in Fig. 2 all are idle. A multicast connection from an input of the input switch to the f distinct output switches cannot be realized if and only if there exists an output switch whose first k inputs (i.e., middle-output interstage links) are busy.

It was shown in [7] that under the link independent assumption

$$
P(\varepsilon \mid \overline{a_1}, \ldots, \overline{a_k}, a_{k+1}, \ldots, a_m) = 1 - (1 - p^k)^f. \quad (2)
$$

By combining (1) and (2), we obtain the blocking probability for a multicast connection with fanout f

$$
P_B(f) = \sum_{k=0}^m \binom{m}{k} q^k p^{m-k} [1 - (1 - p^k)^f]. \quad (3)
$$

Now, suppose the fanout is uniformly distributed over 1 to r. The "average" value of the blocking probability over all possible fanouts can be written as

$$
\begin{aligned}
P_B &= \frac{1}{r} \sum_{f=1}^r P_B(f) \\
&= \frac{1}{r} \sum_{f=1}^r \sum_{k=0}^m \binom{m}{k} q^k p^{m-k} [1 - (1 - p^k)^f]
\end{aligned}
$$

$$(4)$$

where P_B is referred as the *blocking probability of the $v(m, n, r)$ multicast network*.

III. AN IMPROVED ANALYTICAL MODEL

Although the analytical model in [7] shows the same trend as the simulation for the blocking behavior of the network, and it matches well with the simulation when $m > cn$ for $c > 1$, there is still some gap in blocking probabilities between the analytical model and simulation results, especially when m is very close to n. In this section, we improve the analytical model from two aspects as discussed in the following subsections.

A. Improvement Based on the Limitation of Output Switches

First we can make some correction to the blocking probability (3) based on the limitation of output switches. As only

one-to-many or one-to-one connections are considered and there are n outputs on each output switch, if an output switch is chosen as one of destinations in a multicast connection, this output switch must have at least one idle output and have at most $n - 1$ busy inputs. From Lemma 1 and Fig. 2, in the case of $k > n - 1$, there must exist some idle input on each of f output switches. Thus the conditional blocking probability in this case becomes 0, and it can be modified to

$$
\begin{aligned}
&P(\varepsilon \mid \overline{a_1}, \ldots, \overline{a_k}, a_{k+1}, \ldots, a_m) \\
&\quad = \begin{cases} 1 - (1 - p^k)^f, & \text{if } 1 \leq k \leq n-1 \\ 0, & \text{if } k \geq n. \end{cases}
\end{aligned}
$$

$$(5)$$

Then the blocking probability for a multicast connection with fanout f in (3) should be corrected to

$$
P_B(f) = \sum_{k=0}^{n-1} \binom{m}{k} q^k p^{m-k} [1 - (1 - p^k)^f]. \quad (6)
$$

B. Improvement Based on the Busy Interstage Link Ratio

Consider the number of busy input-middle interstage links and the number busy middle-output interstage links at any time. As only one-to-many or one-to-one connections are considered, in general we should have fewer busy input-middle interstage links than busy middle-output interstage links. Thus assumption 3 in the last section may not hold. Let the probability that a middle-output interstage link is busy and idle be

$$
p_b = \frac{an}{m}, \quad q_b = 1 - p_b \quad (7)
$$

respectively, as before, and the probability that an input-middle interstage link is busy and idle be corrected to

$$
p_a = \alpha p_b \quad \text{and} \quad q_a = 1 - p_a \quad (8)
$$

respectively, where $\alpha \leq 1$. Then the blocking probability for a multicast connection with fanout f in (6) can be rewritten as

$$
P_B(f) = \sum_{k=0}^{n-1} \binom{m}{k} q_a^k p_a^{m-k} \left[1 - \left(1 - p_b^k \right)^f \right]. \quad (9)
$$

Hence, the overall blocking probability of $v(m, n, r)$ *multicast network* in the modified model can be expressed as

$$
P_B = \frac{1}{r} \sum_{f=1}^r \sum_{k=0}^{n-1} \binom{m}{k} q_a^k p_a^{m-k} \left[1 - \left(1 - p_b^k \right)^f \right]. \quad (10)
$$

Now it remains to determine the value of α which actually represents the average ratio of the number of busy input-middle interstage links to the number of busy middle-output interstage links.

Let's first consider the multicast connection in Fig. 2. For the given input switch, k middle switches and f output switches, there are many ways to realize the multicast connection from the input switch to the f output switches via up to $\min(k, f)$ middle switches. Clearly, this multicast connection contributes f busy links among all middle-output interstage links. In order to find the average number of busy input-middle interstage links

contributed by such multicast connection, we consider all possible multicast connections to the f output switches, and assume that each such a multicast connection is chosen equally likely. In general, a multicast connection can be fanouted through j middle switches to the f output switches, where $1 \le j \le \min(k, f)$. Thus, this multicast connection contributes j busy links among all input-middle interstage links. We now calculate how many ways such a multicast connection uses j busy input-middle interstage links in the subnetwork shown in Fig. 2. We have $\binom{k}{j}$ ways to choose j middle switches, and then $S(f, j) \cdot j!$ ways to partition the f output switches to j disjoint sets so that each of the j middle switches is routed to a different set of output switches, where $S(f, j)$ is the Stirling number of the second kind [8]. Denote $\binom{k}{j} \cdot j!$ as $(k)_j$, which equals $k(k-1) \cdots (k-j+1)$. Hence there are a total of $(k)_j S(f, j)$ ways. We can obtain the following lemma.

Lemma 2: Under the assumption that each multicast connection from the input switch to the f output switches via up to $\min(k, f)$ middle switches (in Fig. 2) is chosen equally likely, the average number of busy input-middle links contributed by such a multicast connection is

$$k \left[1 - \left(1 - \frac{1}{k} \right)^f \right] \tag{11}$$

and the average ratio of the number of busy input-middle interstage links to the number of busy middle-output interstage links is

$$\frac{k}{f} \left[1 - \left(1 - \frac{1}{k} \right)^f \right]. \tag{12}$$

Proof: Clearly, the total number of ways to realize a multicast connection from the input switch to the f output switches is

$$\sum_{j=1}^{\min(k,f)} (k)_j S(f, j) \tag{13}$$

and the average number of busy input-middle interstage links is

$$\frac{\sum_{j=1}^{\min(k,f)} (k)_j S(f, j) j}{\sum_{j=1}^{\min(k,f)} (k)_j S(f, j)}. \tag{14}$$

To obtain (11), we need the following properties of the Stirling number of the second kind [8]:

$$S(f, j) = S(f - 1, j - 1) + j S(f - 1, j) \tag{15}$$

and

$$\sum_{j=1}^{k} (k)_j S(f, j) = \sum_{j=1}^{f} (k)_j S(f, j) = k^f. \tag{16}$$

From (16), we have

$$\sum_{j=1}^{\min(k,f)} (k)_j S(f, j) = k^f.$$

The combinatorial meaning of the above equation is that for any one of the f output switches, it can be reached from any one of k middle switches (i.e., k ways). Therefore, the total number of ways to realize the multicast connection is k^f. Now by using both (15) and (16) and noticing that $S(f, f) = 1$ and $S(f, 0) = 0$, we can obtain the following.

Case 1) $f \le k$:

$$\sum_{j=1}^{\min(k,f)} (k)_j S(f, j) j$$

$$= \sum_{j=1}^{f} (k)_j S(f, j) j$$

$$= \sum_{j=1}^{f} (k)_j [S(f + 1, j) - S(f, j - 1)]$$

$$= \left[\sum_{j=1}^{f+1} (k)_j S(f + 1, j) - (k)_{f+1} S(f + 1, f + 1) \right]$$

$$\quad - \left[\sum_{j=1}^{f} (k)_{j+1} S(f, j) - (k)_{f+1} S(f, f) \right]$$

$$= k^{f+1} - k \sum_{j=1}^{f} (k-1)_j S(f, j)$$

$$= k^{f+1} - k(k-1)^f. \tag{17}$$

Case 2) $f > k$:

$$\sum_{j=1}^{\min(k,f)} (k)_j S(f, j) j$$

$$= \sum_{j=1}^{k} (k)_j S(f, j) j$$

$$= \sum_{j=1}^{k} (k)_j [S(f + 1, j) - S(f, j - 1)]$$

$$= \sum_{j=1}^{k} (k)_j S(f + 1, j) - \sum_{j=1}^{k} (k)_{j+1} S(f, j)$$

$$= k^{f+1} - k \sum_{j=1}^{k-1} (k-1)_j S(f, j)$$

$$= k^{f+1} - k(k-1)^f. \tag{18}$$

Therefore, from (17) and (18), (14) becomes

$$\frac{k[k^f - (k-1)^f]}{k^f} = k \left[1 - \left(1 - \frac{1}{k} \right)^f \right]$$

which is exactly (11). Finally, since the number of busy middle-output interstage links contributed by such a multicast connection is always f, we have (12) holds. □
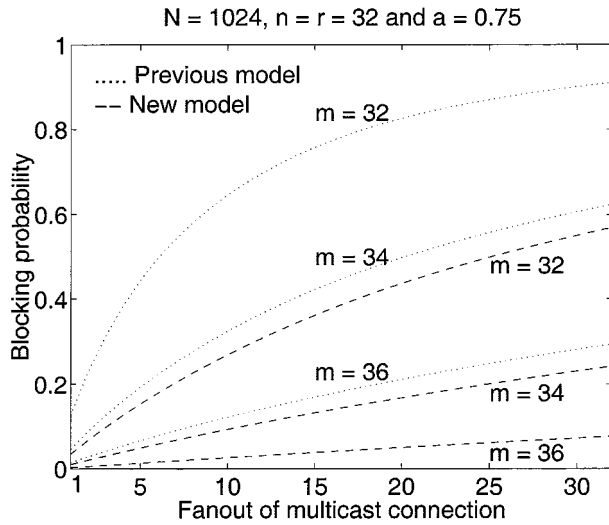
Fig. 3. Comparison of blocking probabilities in different models for a $v(m, 32, 32)$ network with fanouts between 1 and 32.

We now discuss the ratio shown in (12). We observe that for any $k, f \geq 1$, the ratio

$$\frac{k}{f}\left[1 - \left(1 - \frac{1}{k}\right)^f\right] \leq \frac{k}{f}\left[1 - \left(1 - \frac{1}{k} \cdot f\right)\right] = 1.$$

In general, it is difficult to determine the actual ratio of the busy input-middle interstage links to the busy middle-output interstage links. In this paper, we make a reasonable assumption that the ratio is obtained by randomly picking up a multicast connection with fanout f which uses up to f middle switches among m middle switches. By applying Lemma 2 and noticing that f is chosen equally likely among $1, 2, \ldots, r$, we can let the α in (8) take the average value of (12) for $k = m$ and for all possible values of f. That is

$$\alpha = \frac{1}{r}\sum_{f=1}^{r}\frac{m}{f}\left[1 - \left(1 - \frac{1}{m}\right)^f\right]. \tag{19}$$

In Fig. 3, we compare the blocking probabilities of the new model in (9) and the previous model in (3) for a $v(m, 32, 32)$ network with different fanouts. We can see that the blocking probabilities under the new model are smaller than those under the previous model.

IV. COMPARISON BETWEEN ANALYTICAL MODELS AND SIMULATION RESULTS

In this section, we compare the analytical models with the simulation results. Before showing the comparison, we briefly describe the routing algorithm, routing control strategies, and simulation model used in the simulation. First of all, we introduce the following terminologies on the states of the $v(m, n, r)$ network.

For any input port $i \in \{1, 2 \ldots, nr\}$, the set of middle switches with currently unused links to the input switch associated with input port i is referred to as the *available middle*

switches of input i. Let $M_j, j \in \{1, 2, \ldots, m\}$, denote the currently occupied outputs of middle switch j and we refer it to as the *destination set* of middle switch j.

Given a $v(m, n, r)$ multicast network with destination sets $M_1, M_2, \ldots, M_m$ and a new connection request from input port i, I_i (I_i is defined as the output switches to be connected from input port i in the multicast connection), the main function of a routing algorithm is to choose a set of middle switches which can satisfy the connection request. It was shown [5] that a connection request I_i can be satisfied by using some x ($x \geq 1$) middle switches, say, $j_1, j_2, \ldots, j_x$, from among the available middle switches of a $v(m, n, r)$ network if and only if $I_i \cap (\bigcap_{k=1}^{x} M_{j_k}) = \phi$.

The following generic algorithm is used for routing in a $v(m, n, r)$ multicast network.

Algorithm:

Step 1) If no available middle switches for the current connection request, then exit without making the connection, otherwise go to Step 2.

Step 2) Choose a nonfull middle switch (i.e., a middle switch with at least one idle output link) among the available middle switches for the connection request according to some control strategy. If no such middle switch exists, then exit without making the connection.

Step 3) Realize as large as possible portion of the connection request in the middle switch chosen in Step 2.

Step 4) Update the connection request by discarding the portion that is satisfied by the middle switch chosen in Step 2.

Step 5) If the connection request is nonempty, go to Step 1.

We employ the following three routing control strategies in Step 2 of the above multicast routing algorithm in the simulation for choosing middle switches from the available middle switches in a $v(m, n, r)$ multicast network.

1) *Smallest Relative Cardinality Strategy:* Choose a middle switch whose destination set has the smallest cardinality with respect to the connection request.

2) *Largest Relative Cardinality Strategy:* Choose a middle switch whose destination set has the largest cardinality with respect to the connection request.

3) *Random Strategy:* Choose a middle switch at random.

In the simulation, two types of network traffic are considered: uniform traffic and Poisson traffic. In the uniform traffic model, both the inter-arrival time of connection requests and the holding time of each multicast connection follow the uniform distribution. In the Poisson traffic model, the arrival process of connection requests is a Poisson process and the holding time of each multicast connection follows the exponential distribution. The workload in the simulation is measured by the network utilization, which is defined as the ratio of the total number of busy output ports to the network size N. Finally the blocking probability in the simulation is computed by

$$P_B = \frac{\text{The total number of connection requests blocked}}{\text{The total number of connection requests generated}}.$$
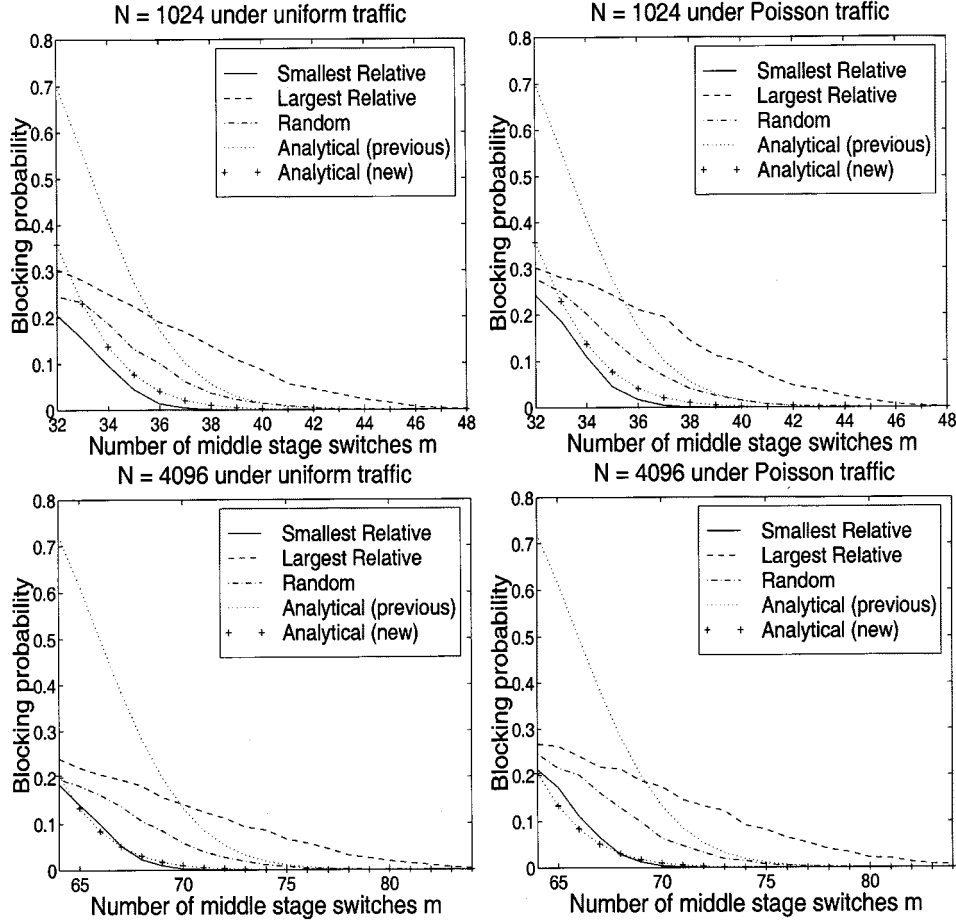
Fig. 4. The blocking probability comparison between the analytical models and the simulation results under three routing control strategies for the $v(m, 32, 32)$ and $v(m, 64, 64)$ multicast networks.

Now, we present the simulation results for two network sizes: 1) $N = 1024, n = r = 32$, and $32 \leq m \leq 48$; 2) $N = 4096, n = r = 64$, and $64 \leq m \leq 84$.

For each network size, control strategy and traffic model, the network is simulated for five runs with different initial network states, and the final results are averaged over these five runs. In each run, 5000 connection requests are handled for network size 1024 with the average network utilization=75%, and 10 000 connection requests are handled for network size 4096 with network utilization=80%. In both cases, 95% confidence interval is achieved.

In Fig. 4, we plot the blocking probability corresponding to $32 \leq m \leq 48$ for network size 1024, and the blocking probability corresponding to $64 \leq m \leq 84$ for network size 4096. The results were obtained under three routing control strategies for uniform traffic and Poisson traffic. We also plot the analytical curves of the previous model (4) and the improved model (10) in Fig. 4. We can see that the curve of the new model matches better with the simulation results, especially in the case of m is very close to n.

Finally, notice that, although the analytical models and the simulation results were obtained under quite different assumptions, they reveal the same trend in the blocking behavior of the $v(m, n, r)$ multicast network: when m gets slightly larger than n, the network becomes almost nonblocking.

V. CONCLUSION

In this paper, we have presented an improved analytical model for the blocking probability of $v(m, n, r)$ multicast networks based on more reasonable assumptions. The new model is compared with the previous model and the simulation results. The improved model matches better with the simulation results and further confirms that a $v(m, n, r)$ network with a comparable number of middle switches to a permutation network is almost nonblocking for multicast connections and can provide cost-effective support for multicast communication.

REFERENCES

[1] C. Clos, "A study of nonblocking switching networks," *Bell Syst. Tech. J.*, vol. 32, pp. 406–424, 1953.
[2] G. M. Masson and B. W. Jordan, "Generalized multi-stage connection networks," *Networks*, vol. 2, pp. 191–209, 1972.
[3] F. K. Hwang and A. Jajszczyk, "On nonblocking multiconnection networks," *IEEE Trans. Commun.*, vol. COM-34, pp. 1038–1041, 1986.
[4] Y. Yang and G. M. Masson, "Nonblocking broadcast switching networks," *IEEE Trans. Comput.*, vol. 40, pp. 1005–1015, Sept. 1991.
[5] ——, "The necessary conditions for Clos-type nonblocking multicast networks," *IEEE Trans. Comput.*, vol. 48, pp. 1214–1227, Nov. 1999.
[6] E. W. Zegura, "Evaluating blocking probability in generalized connectors," *IEEE/ACM Trans. Networking*, vol. 3, pp. 387–398, Aug. 1995.
[7] Y. Yang and J. Wang, "On blocking probability of multicast networks," *IEEE Trans. Commun.*, vol. 46, pp. 957–968, July 1998.
[8] K. P. Bogart, *Introductory Combinatorics*, 2nd ed.   Orlando, FL: Harcourt Brace Jovanovich, 1990.

Yuanyuan Yang (SM'98) received the B.Eng. and M.S. degrees in computer science and engineering from Tsinghua University, Beijing, China, and the M.S.E. and Ph.D. degrees in computer science from Johns Hopkins University, Baltimore, MD.

She is currently an Associate Professor of Computer Engineering, with a Joint Appointment in Computer Science, at the State University of New York (SUNY) at Stony Brook. Before joining SUNY Stony Brook, she was a Faculty Member with the Department of Computer Science, University of Vermont, Burlington, from 1992 to 1999 (Associate Professor from 1998 to 1999). Her research interests include parallel and distributed computing and systems, high-speed networks, optical networks, high-performance computer architecture, and fault-tolerant computing. She has published extensively in major journals and refereed conference proceedings related to these research areas. She holds two U.S. patents in the area of multicast communication networks, with five more patents pending. Her research has been supported by the U.S. National Science Foundation and Army Research Office.

Dr. Yang has served on the program/organizing committees of a number of international conferences. She is an Associate Editor for IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS. She is a member of the Association for Computing Machinery, IEEE Computer Society, and IEEE Communications Society. For more information, please visit her web page at http://www.ece.sunysb.edu/~yang.

Jianchao Wang received the B.Eng. degree in computer engineering from Tsinghua University, Beijing, China, and the M.S. and Ph.D. degrees in computer science from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China.

He is currently a Principal Member of Technical Staff at GTE Laboratories, Inc., Waltham, MA. Before joining GTE Labs, he was with the Institute of Computing Technology, Chinese Academy of Sciences, Johns Hopkins University, and Legent Corporation, Marlboro, MA. He has five U.S. patents granted or pending. His research interests include IP telephony, databases, programming languages, computer communication networks, computer algorithms, and fault-tolerant computing.

Dr. Wang is a member of the IEEE Computer Society and the Association for Computing Machinery. He has received a number of excellence/achievement awards from GTE and Legent.