Blocking Probability of Bicast Connections in a Rearrangeable Clos Network

Somdip Datta and Hisashi Kobayashi
Dept. of Electrical Engineering
Princeton University
Princeton, NJ 08544
Email: datta@princeton.edu, hisashi@princeton.edu

***Abstract*— Clos networks, which are ubiquitous in large capacity switches, have been widely studied for unicast connection requests. But most of these results do not hold when the switch has to support multicast connections. Our interest is specifically in bicast connections which are required for setting up backup protected circuits in optical backbone networks. In this work we have studied the non-blocking properties of *rearrangeable* Clos switches. Furthermore, we have developed expressions to estimate the blocking probability when the non-blocking conditions are not met. We have also run simulations to determine the blocking probability which correspond well with the estimates. We have considered two ways of realizing a bicast connection - by splitting the circuit at the outer stage or at the middle stage.**

I. Introduction

The multi-stage Clos network is the most widely used architecture to build circuit switches of large size using smaller switching blocks. Figure 1 shows the general schematic of a three-stage Clos network with nr input/output ports, divided into r blocks of n ports each. There are m middle stage blocks of size $r \times r$ each. Each middle-stage block is connected to each and every input-stage and output-stage block by exactly one link. The critical issue in the design of a Clos switch is to eliminate or reduce *blocking*. Blocking is said to have occurred when a path can't be found to connect an idle input port to an idle output port in the switch.

If the switch is *rearrangeable*, i.e. existing connections are allowed to be rearranged to accommodate a new connection request, there will not be any blocking if $m \geq n$ (Slepian-Duguid Theorem, 1959). On the other hand, a *non-rearrangeable* Clos switch is *strictly non-blocking* when $m \geq 2n - 2$ (Clos, 1953, [1]). When the non-blocking property is not satisfied, the analysis of the blocking probability of a Clos switch is practically intractable. However, several approximations exist in the literature, of varying degree of accuracy and simplicity [2], [3], [4], [7].

The results for strictly non-blocking criteria and the blocking probability however ceases to be valid when there are one-to-many or many-to-one connections in the switch. One-to-many connections are required, for example, in computer communication networks where data might be required to be simultaneously broadcast to multiple ports. The principal motivation for our work is an optical backbone network with protected circuits (Figure 2). Backbone service providers, for reasons of security and reliability, provide a primary as well as a backup path for most circuits so that fiber cuts and component failures do not lead to disruption of service for their clients. The source node for the circuit-pair broadcasts data from one input port into two output ports, while the destination node selects the data from one of two input ports to go into one output port. Thus each switch is subjected to a mixture of unicast and bicast connection requests. Furthermore, bicast connections can be realized by implementing broadcast/select connections in the middle stage blocks (Figure 3b), or alternately, they can be realized by implementing broadcast/select in the input and output stage blocks (Figure 3a). In this paper we have discussed the properties of both kinds of realizations, though elaborated more on the latter.

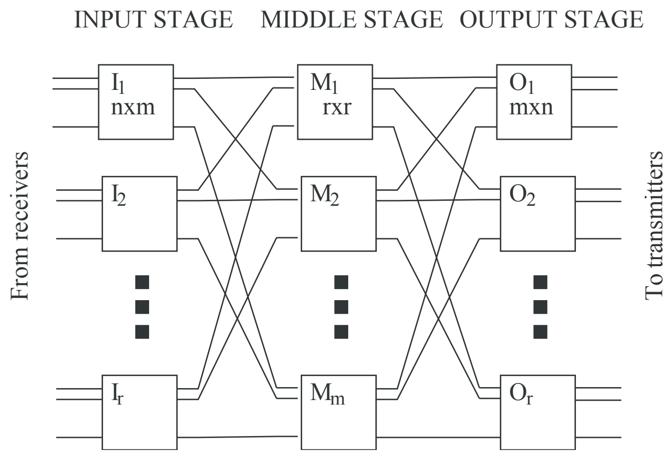

Fig. 1. Schematic of a three-stage Clos switch with nr input/output ports.

Table I gives the conditions for a Clos switch to be strictly non-blocking in the presence of bicast connections, for both unicast and bicast requests, assuming sufficiently high r.

Realization	Outer stage splitting	Middle Stage splitting
Unicast	$m > 4n - 4$	$m > 2n - 2$
Bicast	$m > 4n - 3$	$m > 3n - 3$

TABLE I

Stricly non-blocking criteria for a Clos switch in the presence of bicast connections

We can see that in the presence of bicast connections, the

0-7803-8219-6/04/$17.00 ©2004 IEEE

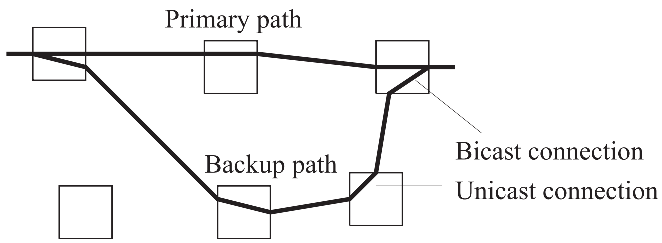

Fig. 2. Each switch in a backbone network is implemented as a Clos network. The originating and terminating switches of a protected circuit requires a bicast (one-to-two or two-to-one) connection between its ports.

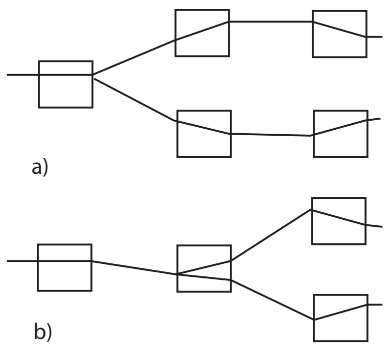

Fig. 3. (a) Realizing a bicast connection in a Clos switch with outer stage splitting and (b) middle stage splitting.

strictly non-blocking condition leads to a much costlier design, hence many switch vendors are willing to drop this condition as long as the blocking probability is still low. If the switch is *rearrangeable*, i.e. existing connections are allowed to be rearranged to accommodate a new connection request, the non-blocking condition is less costly ($m \geq 2n$). In our previous work [10], [11] we have studied the blocking probability of *non-rearrangeable* Clos switches in the presence of bicast connections when the strictly non-blocking conditions are not met. In this paper we will analyze *rearrangeable* Clos networks with bicast connections and determine when they are non-blocking (section II), how to route bicast requests (section III) and what is their blocking probability when the non-blocking conditions are not met (section IV). In section V we have described our simulation model and we have analyzed the results in section VI. We conclude in section VII.

II. Non-blocking criteria

In this section we will describe the non-blocking criteria for rearrangeable Clos switches in the presence of bicast connections. Note that the two ways of realizing a bicast connection affect blocking differently (most notably, the number of occupied input to middle stage links is higher for the outer stage split case because of the possible one to two connections at the input/output stages).

A. Middle stage splitting

For the middle stage split case it is difficult to give a sufficient condition tighter than $m \geq 3n$. We can however state the following necessary condition.

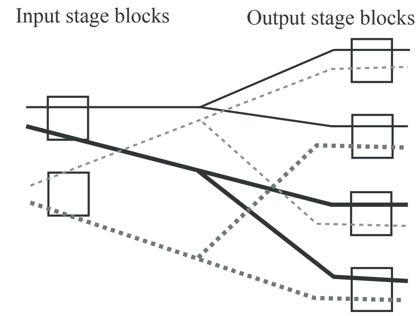

Fig. 4. A set of four bicast connection requests, no two of which can share the same middle block.

Theorem: A rearrangeable Clos switch in the presence of bicast connections realized by splitting at the middle stage, is non-blocking only if $m \geq 2n$.

Proof: Consider the set of four bicast connection requests as in Figure 4, involving a subset of input/output blocks in the network. Any two of the requests have either an input block or an output block in common. Therefore, all four of them must use a different middle block. No more than two input/output ports have been used in any block in this configuration. Thus we can replicate this set of connection requests $n/2$ times and have a set of $2n$ demands such that no two of them can use the same middle block. Thus this configuration represents a set of valid connection requests that must be routed using at least $2n$ middle blocks with or without rearrangement.

B. Outer stage splitting

It is easy to see that $m \geq 2n$ is a necessary condition as every input port might have to be connected to two middle blocks. Similarly, $m \geq 2n$ is also a sufficient condition as the bicast connection requests can be represented as two unicast requests in a Clos switch with $2n$ input ports per block. In the rest of this paper we have assumed that the bicast connections have been realized by splitting at the outer stages, leaving the middle stage split case for future study.

III. Routing

The routing algorithm for a request to connect input port i to output ports j_1 and j_2 is as follows. As all connections are bidirectional, the process also involves connecting input ports j_1 and j_2 to output port i.

1) First consider the path from input i to output j_1. If there is a middle switch accessible from both, use it and go to step 5.
2) Find a middle switch accessible from input block i, call it a. If none is found, the path is blocked. Abort.
3) Find a middle switch accessible from output block j_1, call it b. If none, the path is blocked. Abort.
4) Use the looping algorithm [6] to rearrange the connections and use middle block a to connect i and j_1.
5) Repeat above procedure for the paths from input i to output j_2; input j_1 to output i; input j_2 to output i. The request fails if any of these paths is blocked.

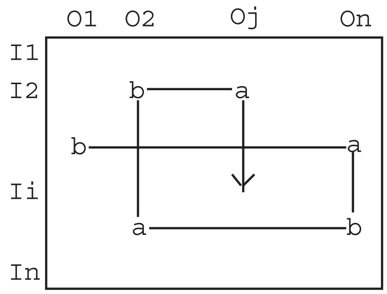

Fig. 5. Looping algorithm to connect a request from Ii to Oj. Row i contains b but not a and column j contains a but not b. The new request can be accommodated by rearranging connections along the path shown.

Looping algorithm A convenient representation to describe this algorithm is the *Paull matrix* P (Figure 5) [6]. The rows of P correspond to the input blocks, the columns to the output blocks. Cell (i, j) contains the set of middle blocks carrying a connection from input block i to output block j. As two connections from the same input block or same output block can't share a middle block, no middle block can appear more than once in a row or a column.

A new request (i, j) can be accommodated without rearrangement if a middle block a can be found which appears neither in row i nor in column j. Suppose no such a can be found. Input block i has at most $n-1$ other connections and so does output block j. If $m \geq n$, there exists a middle block a accessible from input block i, and a middle block b accessible from output block j. In other words, there is a pair a, b such that row i does not contain a and column j does not contain b. Additionally, column j contains a, otherwise we could have used a without rearrangement.

Starting from the a in column j, we search the $abab$... path such that every a following a b lies in the same column and every b following an a lies in the same row. The path stops when no more such a or b can be found. Swap a and b in the path. Column j now does not contain an a. Use a for the new request. This algorithm finds a rearrangement in at most $2n$ swaps.

IV. Blocking probability

In this section we analyze the blocking probability in a rearrangeable Clos switch in the presence of bicast requests realized by splitting at the outer stages. Note that using the looping algorithm we can successfully connect any connection request as long as at least one middle block is accessible from every input/output port involved. Therefore, assuming i,j_1,j_2 to be distinct, a bidirectional connection request from input i to output j_1 and j_2 can be routed as long as at least one middle block is accessible from the input/output blocks j_1, j_2 and at least two middle blocks are accessible from input/output 1. The blocking probability at full load (P_B) is given by

$$P_B = 1 - (1 - p_m - p_{m-1})^2 (1 - p_m)^4 \qquad (1)$$

where p_m is the probability that exactly m middle blocks are not accessible from an input/output block when all $n-1$ of the input/output ports in that block are busy. The distribution of p_m can be estimated as follows.

For each bidirectional bicast connection, three input ports are used, one of which connects to two middle stage blocks, while the others connect to one each . For each unicast demand two input ports are used both of which connect to one middle stage blocks each. Thus the fraction of input ports which connect to two middle stage blocks (instead of one) is given by,

$$\alpha = \frac{b \cdot 1 + (1-b) \cdot 0}{b \cdot 3 + (1-b) \cdot 2} = \frac{b}{2+b} \qquad (2)$$

where, where b is the fraction of total traffic that is bicast.

Therefore the number of input to middle links occupied at full load (x) is described by the truncated binomial distribution p_x,

$$p_x = \bar{p_x} \cdot \left[\sum_{x=n-1}^{m} \bar{p_x} \right]^{-1} \qquad (3)$$

where,

$$\bar{p_x} = \binom{n-1}{x-n+1} \left[\frac{b}{2+b} \right]^{x-n+1} \left[\frac{2}{2+b} \right]^{2n-2-x} \qquad (4)$$

where $n - 1 \leq x \leq min(m, 2n-2)$.

V. Simulation model

We created a software simulation model as well, with a bernoulli call arrival and exponential call holding process. Bicast connections accounted for b fraction of total connection requests. An important element of the simulation is to select a middle stage block, such that the links required to setup the given unicast/bicast connection are free. If no such block(s) can be found, the request is unsuccessful. Otherwise, we have to choose between possibly many available blocks. If the choice is random, we call it *Random Selection*. If we deliberately select the most highly loaded block, we call it *Packed Selection*. Two practical and easy to implement search methods that closely resemble these selection criteria are as follows:

Fixed Sequential Search in the order M_1, M_2, ... and select the first block that has the required links free. This method is a close approximation of *Packed Selection* because it tends to pack traffic more into the middle stage blocks with lower index numbers.

Rotated Sequential Search in a similar order, but start after the block selected for the last connection and wrap at the end of the list. This method is a close approximation of *Random Selection*.

We have simulated the process at near full load ($a \approx 0.99$) by setting the call hold timer much higher than the call arrival timer. We have used $n = 32$, $r = 32$ and varied m in the range $[45, 55]$. We obtained the blocking probability for unicast and bicast connections for $b = 0.5$ and $b = 1.0$, using both *Fixed Sequential* and *Rotated Sequential* Selection. The next section summarizes the results.

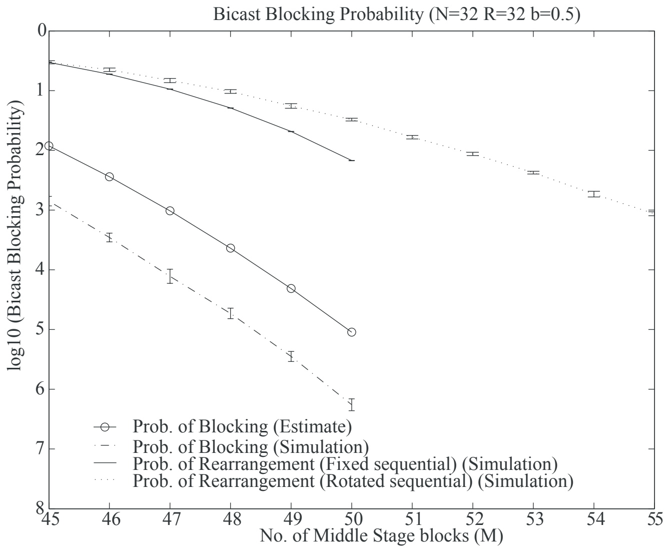

Fig. 6. Blocking probability of bicast connection requests at full load with $n = 32$, $r = 32$, $b = 0.5$ and *outer stage splitting*.

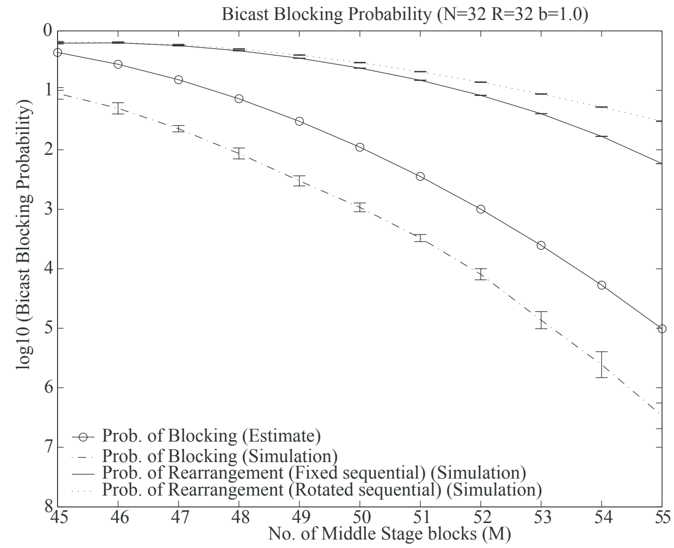

Fig. 7. Blocking probability of bicast connection requests at full load with $n = 32$, $r = 32$, $b = 1.0$ and *outer stage splitting*.

VI. Results

Figure 6 and 7 show the simulation results vis-a-vis analytical estimates for the blocking of bicast requests for $b = 0.5$ and $b = 1.0$ respectively. We have also plotted the probability that the connection request requires a rearrangement of existing connections. The latter has two curves corresponding to the two routing strategies (*fixed sequential* and *rotated sequential* search). Note that the blocking probability after rearrangement does not depend on the routing strategy, and therefore there is a single curve for either routing strategy. The curves from the simulations also show the 90% confidence intervals. Some of the conclusions that can be drawn from these figures are:

1) The blocking probability can be reduced to less than 10_{-6} with 50-55 middle stage blocks which is about 80-85% of the requirement for a non-blocking switch.
2) The analytical estimate is consistently about an order of magnitude higher than the observed results.
3) Rearrangement probability observed with fixed sequential selection is lower than the probability observed with rotated sequential method. This is consistent with the results [9], [10] without rearrangement.
4) The blocking probability is more sensitive to bicast traffic (b) than the rearrangement probability.

VII. Conclusion and Future Work

We have tried to understand the blocking phenomena in rearrangeable Clos switches when bicast connections are present in the switch. Our study includes the non-blocking criteria, analytical estimation of blocking probability when the non-blocking conditions are not met and simulation of the latter using an event queue simulator.

We intend to continue working in the following areas,

1) Rearrangement algorithms in the presence of bicast connections with middle stage splitting.
2) Blocking probability analysis for rearrangeable switches using middle stage splitting.

References

[1] C.Clos, "A Study of Nonblocking Switching Networks," *Bell System Tech. J.*, Vol.32, pp. 406-424, November 1953.

[2] C.Y.Lee, "Analysis of Switching Networks," *The Bell System Tech. J.*, Vol.34, No.6, pp. 1287-1315, November 1955.

[3] M.Karnaugh, "Loss of Point-to-Point Traffic in Three-Stage Curcuit Switches," *IBM J. Res. Develop.*, pp.214-216, May, 1974.

[4] M.Schwartz, *Telecommunication Networks: Protocols, Modeling & Analysis*, pp.539-552, Reading, MA, Addison-Wesley, 1987.

[5] F.K.Hwang and S.C.Liaw, "On Nonblocking Multicast Three-stage Clos Networks," *IEEE/ACM Transactions on Networking*, Vol.8, No.4, pp.535-539, August 2000.

[6] F.K.Hwang, *The Mathematical Theory of Nonblocking Switching Networks*, Series on Applied Mathematics Vol.11, World Scientific Publishing Co. Pte. Ltd., 1998.

[7] Y.Yang, "An Analytical Model on Network Blocking Probability," *IEEE Communication Letters*, Vol.1, No.5, pp. 143-145, September 1997.

[8] Y.Yang, "Supporting Multicast Communications in Clos-Type Switching Networks," *Switching Networks: recent Advances*, pp.143-178, Kluwer Academic Publishers, 2001.

[9] Y.Yang and J.Wang, "On Blocking Probability of Multicast Networks," *IEEE Trans. Communications*, Vol.46, No.7, pp. 957-968, July 1998.

[10] S.Datta, J.Labourdette and H.Kobayashi, "Blocking in a Clos Swtich in the Presence of Bicast Connections with Outer Stage Splitting," *Proc. of 37th Annual Conf. on Information Sciences and Systems*, Baltimore, MD, March 2003.

[11] S.Datta, J.Labourdette and H.Kobayashi, "Blocking Probability of Bicast Connections in a Clos Network," *Proc. of IEEE Globecom*, San Francisco, CA, December 2003.