

## COLLECTIVE COMMUNICATION

- Collective communication involves global data movement and global control among a group of processes in a parallel/distributed computing system.
- Frequently used collective communication operations include broadcast (one-to-all), multicast (one-to-many), all-to-all, gather, scatter, reduction and scan.
- Many scientific applications exhibit the need of such communication patterns. E.g. matrix multiplication, LU-factorization, matrix transposition and fast Fourier transform (FFT).
- Efficient support for collective communication can significantly reduce the communication latency and simplify the programming of parallel computers.

- **Basic types of collective communication**

- **One-to-all communication:** one process is identified as the sender and all processes in the group are receivers.

- \* **Broadcast:** the same message is sent to all receivers.

- \* **Scatter (personalized broadcast):** the sender sends different messages to different receivers.

- **All-to-one communication:** all processes in a process group are senders and one process is identified as the sole receiver.

- \* **Reduce:** different message from different senders are combined together to form a single message for the receiver.

**Combining operator:** addition, multiplication, maximum, minimum, and logical OR, AND, exclusive OR operators.

- \* **Gather:** different messages from different senders are concatenated together for the receiver.
- **All-to-all communication:** every process in a process group sends a message to all other processes in the group.
  - \* **All-to-all broadcast:** every process sends the same message to all other processes.
  - \* **All-to-all personalized exchange (all scatter):** every process sends a distinct message to every other process.
- **An application of basic collective operations: Barrier synchronization.**
  - Many numerical problems can be solved using iterative algorithms that successively compute better approximations to an answer, terminating when either the final answer has been computed or the final answer

has converged. These algorithms normally require all the iterative processes to be synchronized at the end of each iteration.

```
do not_converged →  
    code to implement process  $i$   
    barrier (wait for all  $n$  processes to complete)  
od
```

- Barrier represents a barrier synchronization point which waits for all  $n$  processes to complete. This type of synchronization is called *barrier synchronization* because the delay point at the end of each iteration represents a barrier that all processes have to arrive at before any of them is allowed to pass.

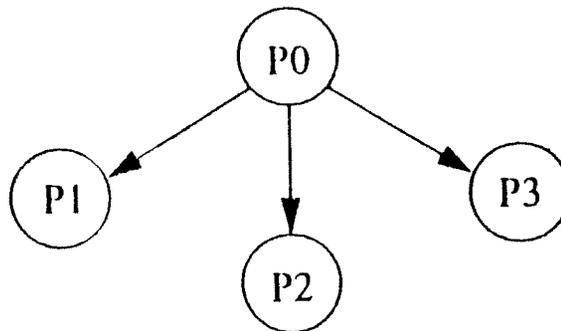
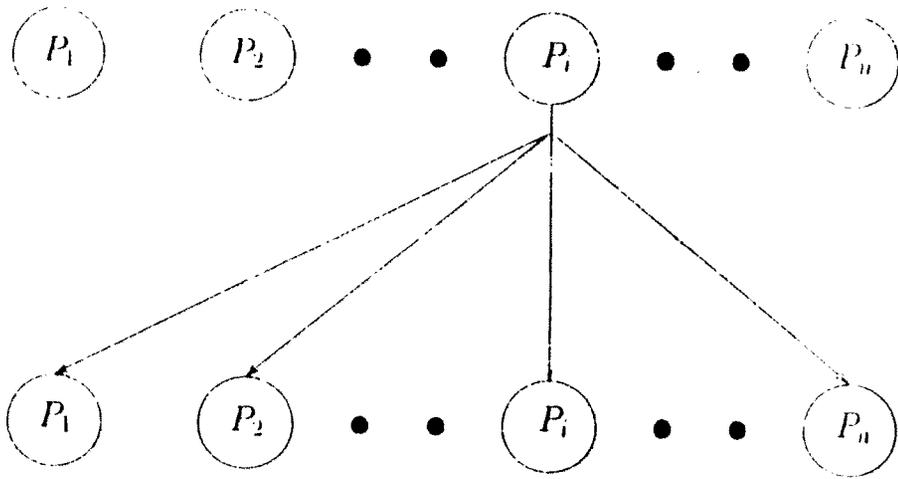
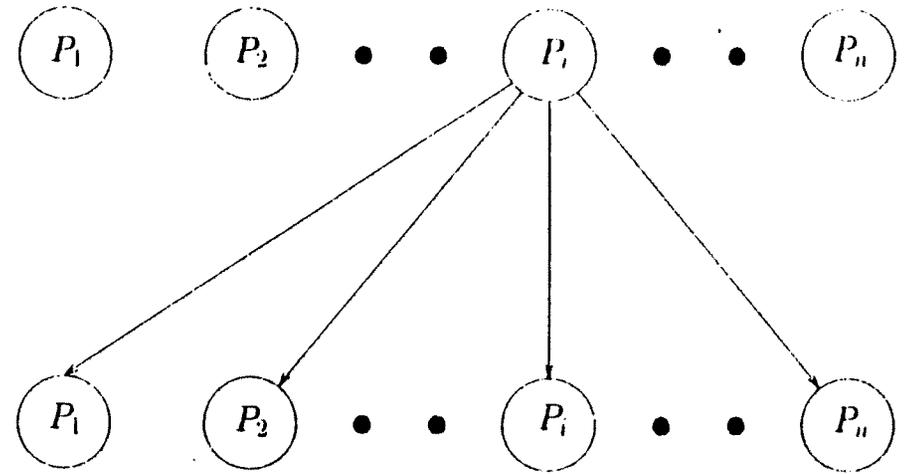


Figure 5.7. A multicast communication pattern with three destinations.

P0: .....	P1: .....	P2: .....	P3: .....
send(msg,P1)	.....	.....	.....
send(msg,P2)	recv(msg,P0)	recv(msg,P0)	recv(msg,P0)
send(msg,P3)	.....	.....	.....

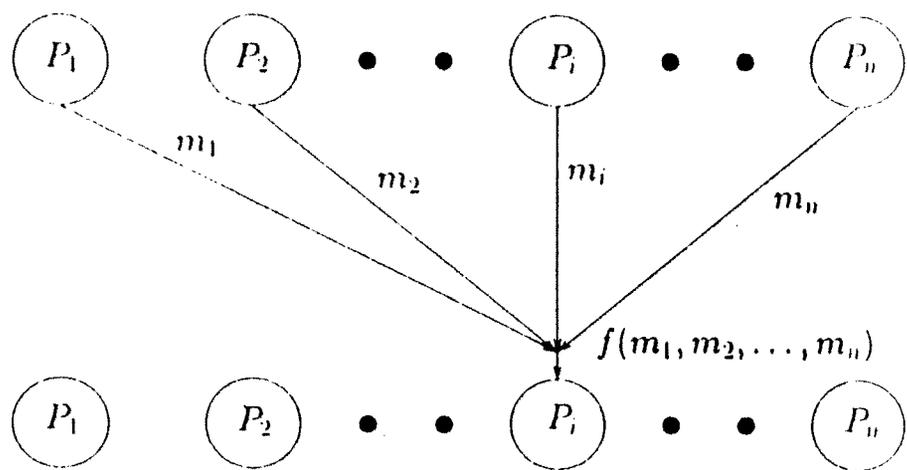


(a) Broadcast Communication.

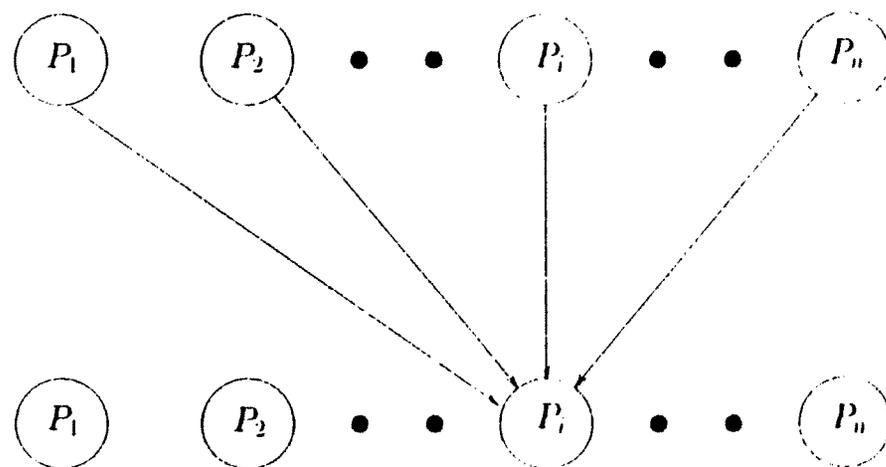


(b) Scatter Communication.

Figure 5.2. Two one-to-all communication patterns.

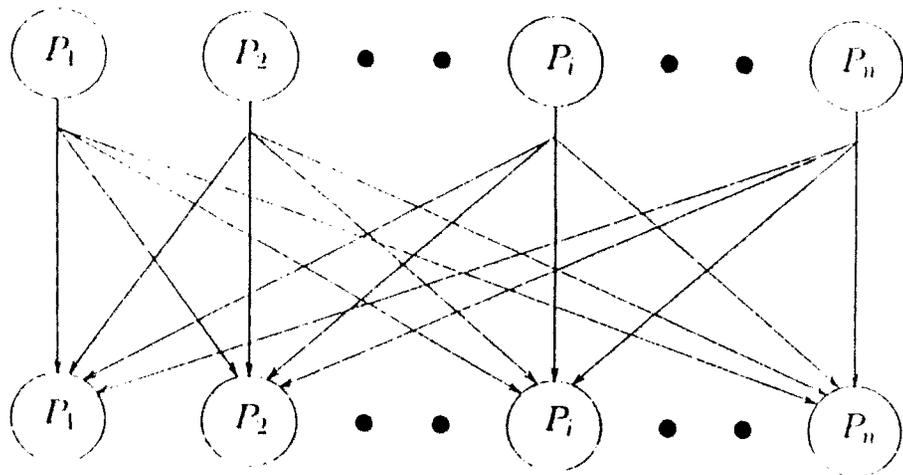


(a) Reduce Communication.

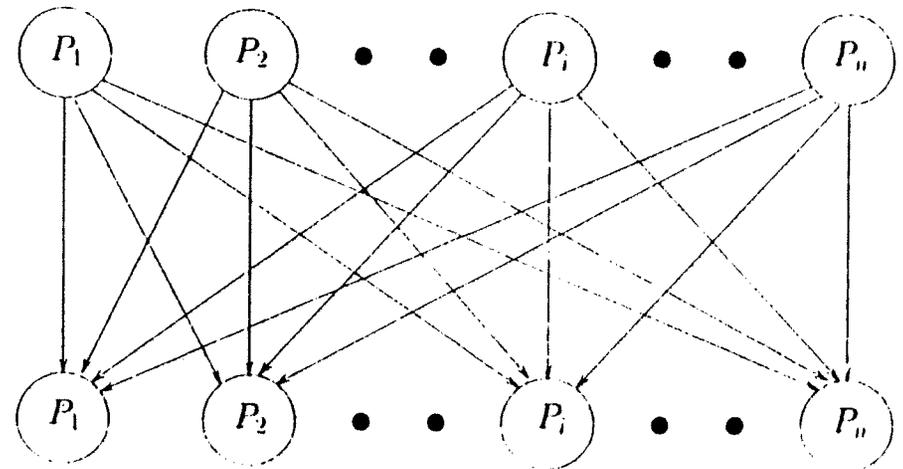


(b) Gather Communication.

Figure 5.3. Two all-to-one communication patterns.



(a) All-Broadcast Communication.



(b) All-Scatter Communication.

Figure 5.4. Two all-to-all communication patterns.

## SYSTEM SUPPORT FOR COLLECTIVE COMMUNICATION

- **Implementing the basic operation: multicast (one-to-many) communication**
  - Unicast approach
  - Software multicast
  - Hardware multicast
- **Performance of multicast communication:**
  - Mainly measured in terms of its latency in delivering a message to all destinations.
  - Other considerations: total traffic generated, deadlock-free routing

- **A wide-sense nonblocking multicast network**

- Terminology

- \* Interconnection network: A (space division) switching system which provides connection paths between input ports and output ports.
- \* One-to-one connection: A connection of an input port to one output port. A maximal set of one-to-one connections is called a permutation assignment.
- \* Multicast connection: A connection where an input port can be simultaneously connected to more than one output port (but an output port can be connected to at most one input port at a time). A maximal set of multicast connections is called a multicast assignment.

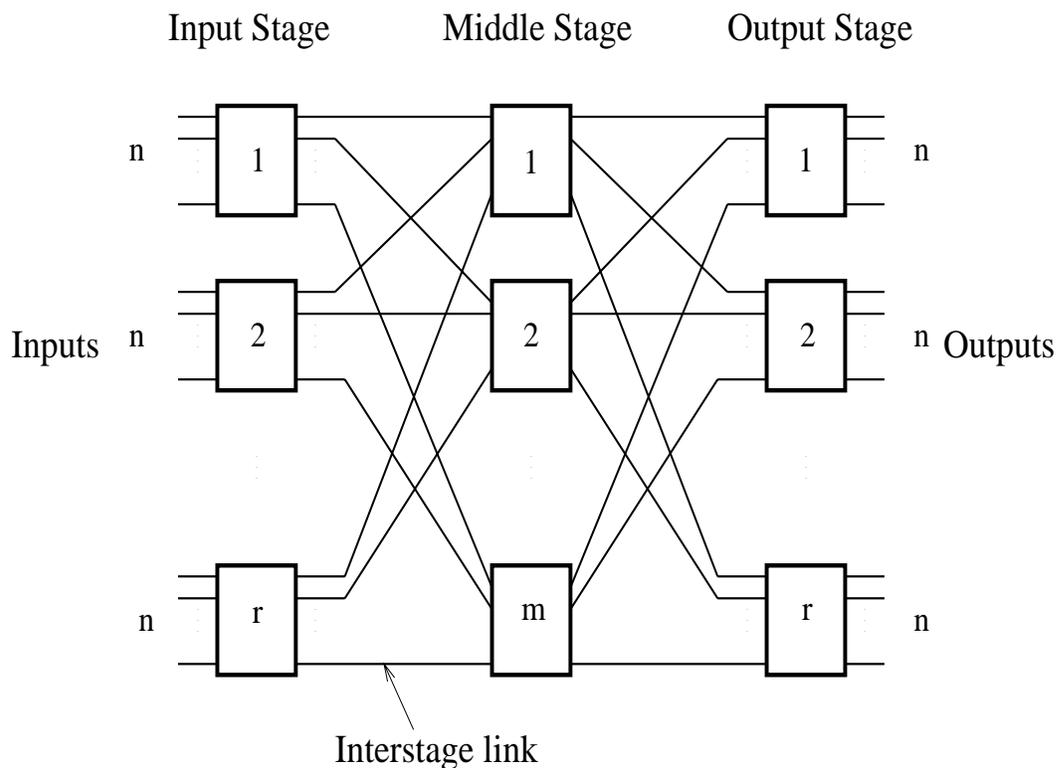
- \* Connection request: Idle input port requests connection path(s) to idle output port(s).
  - \* Multicast network: A network which can realize all possible multicast assignments.
  - \* Nonblocking network: Satisfies all connection requests and rearrangement is never required.
- Motivation:
- \* Multicast (one-to-many) communication is a fundamental communication pattern in both telecommunication networks and scalable parallel computers.
  - \* Examples:
    - In telecommunication:  
Teleconference calls and video-on-demand services.

- In parallel/distributed processing:  
Barrier synchronization, write update/  
invalidate in cache coherence protocols,  
and updates in a distributed database.
- \* A permutation network in general cannot  
support arbitrary multicast.  
For an  $N \times N$  network,  
# Permutation patterns:  $N!$   
# Multicast patterns:  $N^N$
- \* Efficient implementation for multicast is  
critical to system performance.
- \* Support multicast at interconnection net-  
work level.
- \* Many applications require nonblocking ca-  
pability.
  - Reduce overhead associated with rear-  
rangements
  - Avoid disturbances of existing connec-

tions in the network

- Especially important in real-time applications.

– Definition of the Clos network or  $v(m, n, r)$  network



Every switch module is assumed to have multicast capability.

– **Previous results on nonblocking conditions of the Clos network**

\* **A  $v(m, n, r)$  network is nonblocking for permutation assignments [Clos, *BSTJ*, '53], if the number of middle stage switches**

$$m \geq 2n - 1$$

\* **A  $v(m, n, r)$  network is nonblocking for multicast assignments [Masson, *Networks*, '72; Hwang and Jajszczyk, *IEEE Trans. Comm.*, '86], if the number of middle stage switches**

$$m \geq c(nr)$$

– More terminology:

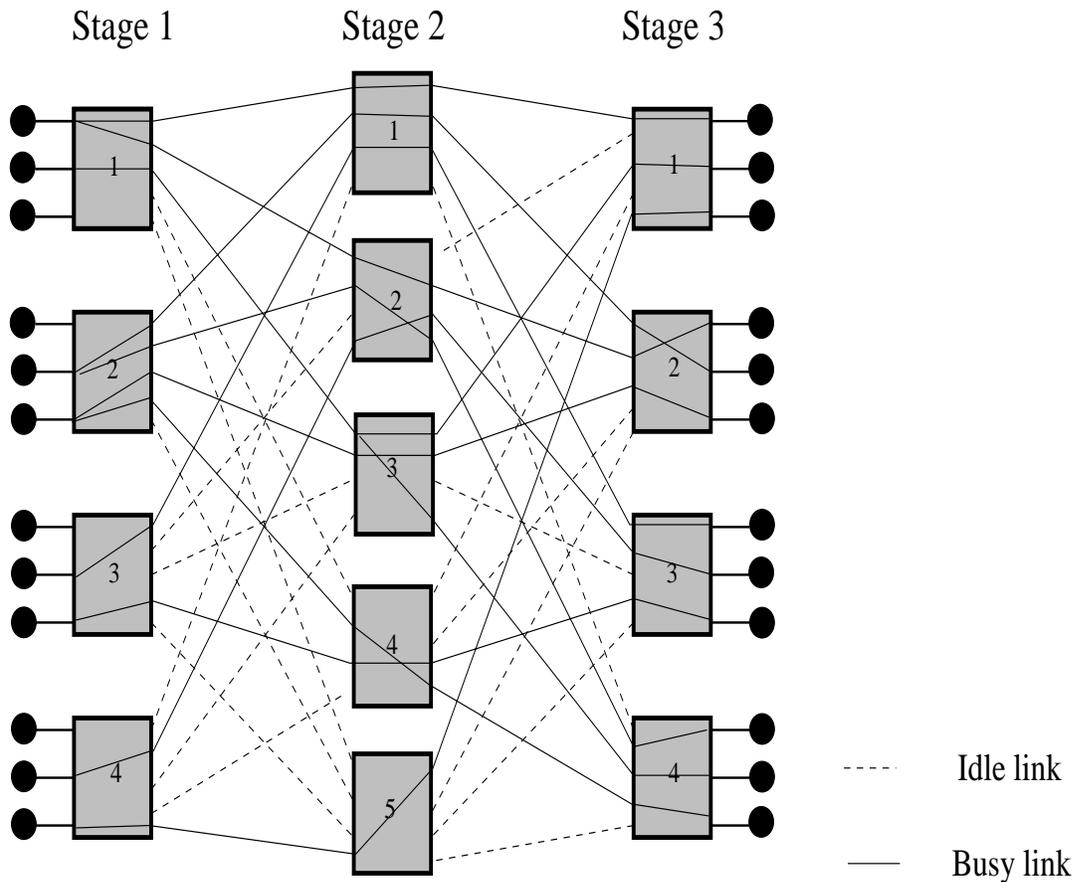
Fanout: A multicast connection from an input port to output ports on  $r'$  output switches is said to have fanout  $r'$ .

Input connection request: An input connection request  $I_j$  is the set of output switches to which input port  $j$  is to be connected.

Destination sets: Destination set  $M_j$  is the set of output switches to which the middle switch  $j$  is providing connection paths from the input ports.

Available middle switches: The available middle switches of input port  $j$  is the set of middle switches with currently unused links to the input switch associated with input  $j$ .

– **Example:**



A  $(5, 3, 4)$  network realizing a multicast assignment characterized as  $I_1 = \{1, 2\}$ ,  $I_2 = \{1, 4\}$ ,  $I_5 = \{2, 4\}$ ,  $I_6 = \{2, 4\}$ ,  $I_8 = \{3\}$ ,  $I_9 = \{3\}$ ,  $I_{11} = \{3\}$ ,  $I_{12} = \{1\}$ , all other  $I_j = \phi$ . The destination sets for the middle switches for the shown network state are  $M_1 = \{1, 2, 3\}$ ,  $M_2 = \{2, 3, 4\}$ ,  $M_3 = \{1, 2, 4\}$ ,  $M_4 = \{3, 4\}$ ,  $M_5 = \{1\}$ .

– Design of multicast networks

- \* Specify an “intelligent” control strategy for satisfying each multicast connection request: choose no more than a certain number of middle switches, say,  $x$ , whose destination set intersections are empty from available middle switches.
- \* Determine how many available middle switches can guarantee that these  $x$  middle switches can always be chosen. We are interested in finding as few as possible available middle switches with this property.
- \* Find the optimal value of  $x$  to minimize the number of middle switches.
- \* Develop an efficient control algorithm to actually find these  $x$  middle switches.
- \* Hardware implementation of the control algorithm to further speed up the routing

**process.**

– **Main results**

**Theorem 1** *We can satisfy a new connection request  $I_i$ ,  $i \in \{1, 2, \dots, nr\}$ , in a  $v(m, n, r)$  network using some  $x$  ( $x \geq 1$ ) middle switches, say,  $j_1, \dots, j_x$ , from among the available middle switches if and only if  $I_i$  and the current destination sets of these  $x$  middle switches satisfy*

$$I_i \cap \left( \bigcap_{k=1}^x M_{j_k} \right) = \phi. \quad (1)$$

**Theorem 2** *For all  $n'$ ,  $1 \leq n' \leq n$ , and for all  $x$ ,  $1 \leq x \leq \min\{n', r\}$ , let  $m'$  be the maximum number of middle switches whose destination sets have the following properties:*

1. *there are at most  $n'$  1's,  $n'$  2's,  $\dots$ ,  $n'$   $r$ 's distributed among the destination sets;*
2. *the intersection of any  $x$  of the destination sets is nonempty.*

***Then***

$$m' \leq n'r^{\frac{1}{x}}$$

**Proof sketch of Theorem 2:**

**WLOG**, suppose these  $m'$  middle switches are  $1, 2, \dots, m'$  with destination sets  $M_1, M_2, \dots, M_{m'}$ . Clearly,

$$\sum_{i=1}^{m'} |M_i| \leq n'r$$

**Let**

$$c_1 = \min_i \{|M_i|\}.$$

**Then**, we obtain that

$$m' \leq \frac{n'r}{c_1} \tag{2}$$

**WLOG**, suppose that the destination set of middle switch 1 has cardinality  $c_1$ , and  $M_1 = \{1, 2, \dots, c_1\}$ .

**Intersect** each of the destination sets  $M_1, M_2, \dots, M_{m'}$  with  $M_1$  and obtain  $m'$  sets  $M_1^1, M_2^1, \dots, M_{m'}^1$

$\dots, M_{m'}^1$  which consist of only elements in  $\{1, 2, \dots, c_1\}$ , and distributed among the  $M_i^1$ 's are at most  $n'$  1's,  $n'$  2's,  $\dots$ ,  $n'$   $c_1$ 's.

Let

$$c_2 = \min_i \{|M_i^1|\}$$

WLOG, suppose that  $M_2^1$  has cardinality  $c_2$ , and  $M_2^1 = \{1, 2, \dots, c_2\}$ . We then get that

$$m' \leq \frac{n'c_1}{c_2} \quad (3)$$

Then intersect each of  $M_1^1, M_2^1, \dots, M_{m'}^1$  with  $M_2^1$  and obtain  $m'$  sets  $M_1^2, M_2^2, \dots, M_{m'}^2$ , which consist of only elements in  $\{1, 2, \dots, c_2\}$ .

After repeating the above process  $x-1$  times, we have

$$m' \leq \min\left\{\frac{n'r}{c_1}, \frac{n'c_1}{c_2}, \dots, \frac{n'c_{x-2}}{c_{x-1}}\right\} \quad (4)$$

We now have a set of  $m'$  intersected destination sets  $M_1^{x-1}, M_2^{x-1}, \dots, M_{m'}^{x-1}$ . Moreover, each  $M_k^{x-1}$  consists of only elements in

$\{1, 2, \dots, c_{x-1}\}$ , and there are at most  $n'$  1's,  $n'$  2's,  $\dots$ , and  $n' c_{x-1}$ 's distributed among these  $m'$  sets. Thus, we have

$$m' \leq n' c_{x-1} \quad (5)$$

Therefore, from (3) and (4), we get that

$$m' \leq \min\left\{\frac{n'r}{c_1}, \frac{n'c_1}{c_2}, \dots, \frac{n'c_{x-2}}{c_{x-1}}, n'c_{x-1}\right\}$$

It can be shown that

$$\max_{c_1, c_2, \dots, c_{x-1}} \min\left\{\frac{n'r}{c_1}, \frac{n'c_1}{c_2}, \dots, \frac{n'c_{x-2}}{c_{x-1}}, n'c_{x-1}\right\} = n'r^{\frac{1}{x}}$$

Therefore,

$$m' \leq n'r^{\frac{1}{x}}$$

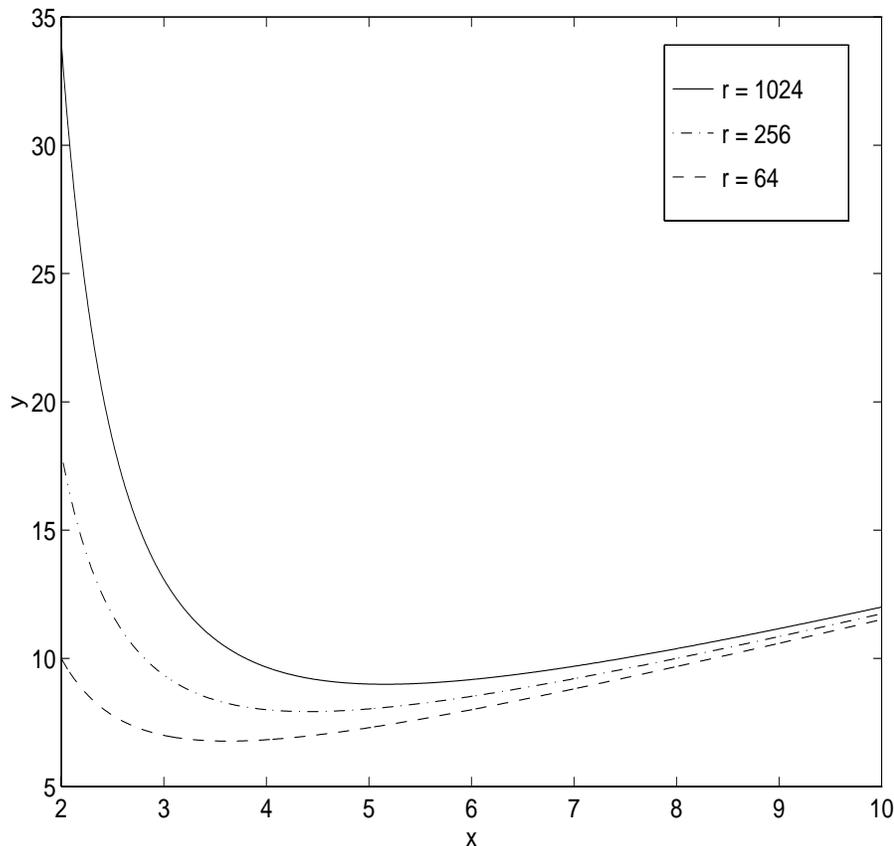
**Sufficient condition on realizing multicast assignment with nonblocking capability:**

**Theorem 3** *A  $v(m, n, r)$  network is nonblocking for multicast assignments if*

$$m > \min_{1 \leq x \leq \min\{n-1, r\}} \{(n-1)(x + r^{\frac{1}{x}})\}$$

**Curve of function**

$$y = x + r^{\frac{1}{x}}$$



## Relationship between $r$ and the coefficient

of  $m$ :  $\min_{1 \leq x \leq \min\{n-1, r\}} (x + r^{\frac{1}{x}})$

$r$	$x$	$x + r^{\frac{1}{x}}$
1	<b>1</b>	<b>2</b>
2	<b>1</b>	<b>3</b>
$4 = 2^2$	<b>2</b>	<b>4</b>
$9 = 3^2$	<b>2</b>	<b>5</b>
$27 = 3^3$	<b>3</b>	<b>6</b>
$81 = 3^4$	<b>4</b>	<b>7</b>
$256 = 4^4$	<b>4</b>	<b>8</b>
$1024 = 4^5$	<b>5</b>	<b>9</b>
$4096 = 4^6$	<b>6</b>	<b>10</b>
$16384 = 4^7$	<b>7</b>	<b>11</b>
$78125 = 5^7$	<b>7</b>	<b>12</b>
$390625 = 5^8$	<b>8</b>	<b>13</b>
$1953125 = 5^9$	<b>9</b>	<b>14</b>
$10077696 = 6^9$	<b>9</b>	<b>15</b>

**A bound on  $m$  as a function of  $n$  and  $r$**

**Theorem 4** *A  $v(m, n, r)$  network is nonblocking for multicast assignments if*

$$m \geq 3(n - 1) \frac{\log r}{\log \log r}$$

**Generalization to restricted multicast assignments:**

**Corollary 1** *A  $v(m, n, r)$  network is nonblocking for restricted multicast assignments, in which each input port can be connected to at most  $d$  ( $1 \leq d < r$ ) output switches, if*

$$m > \min_{1 \leq x \leq \min\{n-1, d\}} \{(n - 1)(x + d^{\frac{1}{x}})\}$$

**In particular, we have**

$$m > (n - 1) \left( \frac{2 \log d}{\log \log d} + (\log d)^{\frac{1}{2}} \right)$$

**Permutation is a special case:**

**Corollary 2** *Setting  $d = 1$  in Corollary 1 yields  $m \geq 2n - 1$ , which is the bound on  $m$  associated with the classical Clos non-blocking permutation networks.*

**Generalization to  $(2k+1)$ -stage networks for  $k > 1$ :**

**Recursively applying the design criteria on each middle stage switch.**

**Crosspoint complexity:**

**Theorem 5** *For each fixed integer  $k \geq 1$ , the minimum number of crosspoints of our  $(2k+1)$  stage  $(N \times N)$  multicast network is*

$$O\left(N^{1+\frac{1}{k+1}}(\log N / \log \log N)^{\frac{k+2}{2}-\frac{1}{k+1}}\right)$$

## Crosspoints comparisons:

### \* Constructive multicast networks:

- Masson's three-stage network:

$$O(N^{\frac{5}{3}})$$

- Hwang and Jajszczyk's three-stage network:  $O(N^{\frac{5}{3}})$

- Feldman, Friedman and Pippenger's two-stage network:  $O(N^{\frac{5}{3}})$ ;

$$\text{three-stage network: } O(N^{\frac{11}{7}})$$

- Three-stage version of this network:

$$O(N^{\frac{3}{2}} \left( \frac{\log N}{\log \log N} \right))$$

### \* Nonconstructive multicast networks:

Feldman, Friedman and Pippenger  $k$ -stage network:

$$O\left(N^{1+\frac{1}{k}} (\log N)^{1-\frac{1}{k}}\right).$$

The control algorithm is co-NP-complete.

– **A linear routing control algorithm**

\* **Given a  $v(m, n, r)$  network satisfying the nonblocking condition on  $m$  in Theorem 3.**

\* **Some  $x$ ,  $1 \leq x \leq \min\{n - 1, r\}$**

\* **A connection request  $I_i$  with**

$$|I_i| = r' \leq r$$

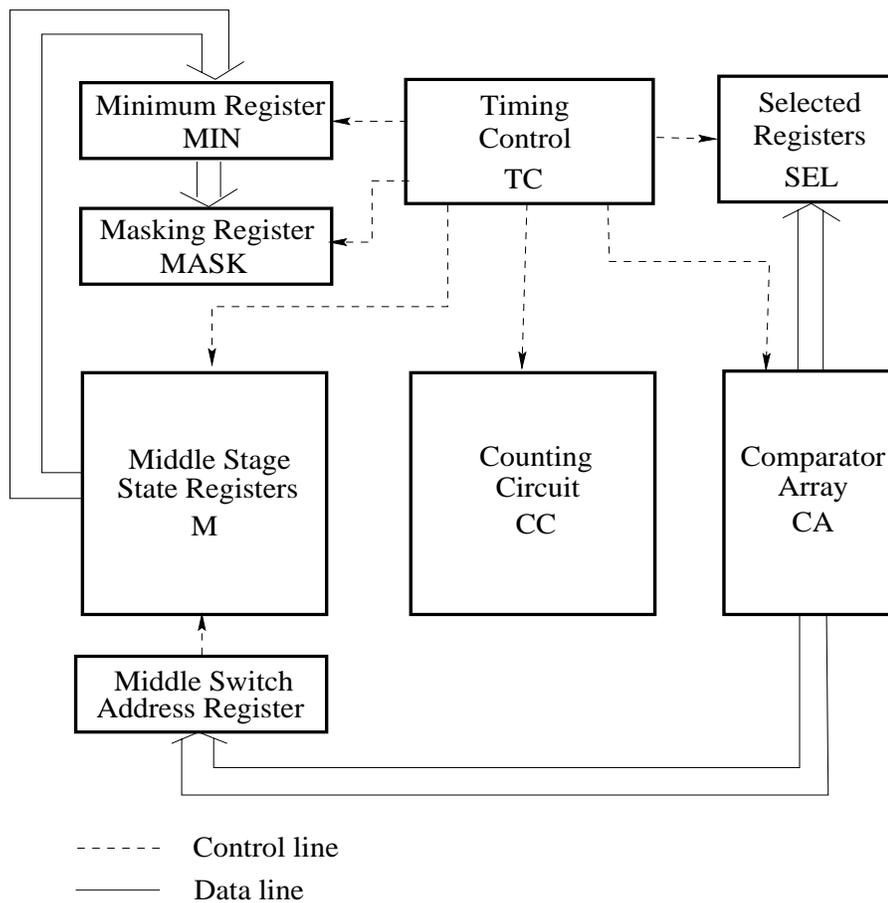
\*  **$m' = (n - 1)r'^{\frac{1}{x}} + 1$  available middle switches with destination sets**

$$M_1, M_2, \dots, M_{m'}.$$

**Algorithm:****Step 1:**  $mid\_switch \leftarrow \phi$ ;**for**  $j = 1$  **to**  $m'$  **do** $S_j \leftarrow M_j \cap I_i$ ;**Step 2:** **repeat****find**  $S_k$  ( $1 \leq k \leq m'$ ) **such that** $|S_k| = \mathbf{min}\{|S_1|, |S_2|, \dots, |S_{m'}|\}$ ; $min\_set \leftarrow S_k$ ; $mid\_switch \leftarrow mid\_switch \cup \{k\}$ ;**if**  $min\_set \neq \phi$  **then****for**  $j = 1$  **to**  $m'$  **do** $S_j \leftarrow S_j \cap min\_set$ ;**until**  $min\_set = \phi$ ;**Step 3:** connect  $I_i$  through the middle switches in  $mid\_switch$  and update the destination sets of these middle switches.

– More efficient implementation of control algorithm in hardware

\* Overview structure



**\* Sequential implementation**

- **Sequentially evaluate the cardinality of each destination set of middle switches and find the minimum cardinality set.**

**\* Parallel implementation**

- **Parallel evaluation of the cardinalities of all destination sets of middle switches by a sequential circuit or a combinational circuit.**

**\* Summary of the various designs of the controller**

<b>Design</b>	<b>Gates</b>	<b>Clocks</b>	<b>Gate Delays</b>
<b>Seq./ Counter</b>	$O(\log r)$	$O\left(\frac{mr \log r}{\log \log r}\right)$	—
<b>Seq./ Adder</b>	$O(r)$	$O\left(\frac{m \log r}{\log \log r}\right)$	—
<b>Parall./ Counter</b>	$O(m \log r)$	$O\left(\frac{r \log r}{\log \log r}\right)$	—
<b>Parall./ Adder</b>	$O(mr)$	$O\left(\frac{\log r}{\log \log r}\right)$	$O\left(\frac{(\log r)^2}{\log \log r}\right)$

– **Necessary nonblocking condition**

- \* **Can the sufficient condition we obtained be further reduced?**
- \* **What is the optimal design for this type of multicast network?**
- \* **Derive necessary conditions for supporting arbitrary multicast assignment under different control strategies by constructing worst case network states which force us to use a certain number of middle stage switches.**
- \* **Employ these necessary conditions to guide the design process of multicast networks.**

– Routing control strategies

*Strategy 1 For each input connection request,  $I_i$ ,  $i \in \{1, 2, \dots, nr\}$ , in the network, always choose the middle switch with the minimum cardinality of destination sets with regard to the unsatisfied portion of  $I_i$  from available middle switches, until  $I_i$  is satisfied, that is, until all middle switches chosen satisfy condition (1).*

Note: This is the strategy used by the routing control algorithm we discussed earlier.

*Strategy 2 For each input connection request  $I_i$ ,  $i \in \{1, 2, \dots, nr\}$ , choose the minimum number of middle switches that satisfy condition (1) for the current network state from the available middle switches.*

*Strategy 3 For each input connection request  $I_i$ ,  $i \in \{1, 2, \dots, nr\}$ , use an empty avail-*

*able middle switch (i.e., middle switch with no connections) only when there is no any subset of non-empty available middle switches can satisfy condition (1).*

- A fundamental lemma for constructing worst case network states:

**Lemma 1** *For sufficiently large  $n$ ,  $r$  and  $\frac{m}{n}(m > n)$ , there exist  $m+n$  subsets of set  $\{1, 2, \dots, r\}$ ,  $\mathcal{I}_1, \mathcal{I}_2, \dots, \mathcal{I}_n, \mathcal{M}_1, \mathcal{M}_2, \dots, \mathcal{M}_m$ , which satisfy the following conditions:*

- 1. the flattened set of  $\{\mathcal{I}_1, \mathcal{I}_2, \dots, \mathcal{I}_n, \mathcal{M}_1, \mathcal{M}_2, \dots, \mathcal{M}_m\}$  is a multiset chosen from set  $\{1, 2, \dots, r\}$  with multiplicity of each element no more than  $n$ ;*
- 2. for some  $x = \Theta\left(\frac{\log r}{\log m - \log n}\right)$  and for any  $\mathcal{I}_i$  ( $1 \leq i \leq n$ ) and any  $\mathcal{M}_{j_1}, \mathcal{M}_{j_2}, \dots, \mathcal{M}_{j_x}$  ( $1 \leq j_1 < j_2 < \dots < j_x \leq m$ )  $\mathcal{I}_i \cap (\cap_{k=1}^x \mathcal{M}_{j_k}) \neq \phi$ .*

- **Lemma 1** can be used to construct a sequence of connection/disconnection requests under these strategies and obtain the following necessary conditions:

**Theorem 6** *The necessary condition for a  $v(m, n, r)$  network to be strictly nonblocking for multicast assignments is  $m \geq \Theta\left(n \frac{\log r}{\log \log r}\right)$ .*

**Theorem 7** *The necessary condition for a  $v(m, n, r)$  multicast network to be nonblocking under Strategy 1 is  $m \geq \Theta\left(n \frac{\log r}{\log \log r}\right)$ .*

**Theorem 8** *The necessary condition for a  $v(m, n, r)$  multicast network to be nonblocking under Strategy 2 is  $m \geq \Theta\left(n \frac{\log r}{\log \log r}\right)$ .*

**Theorem 9** *The necessary condition for a  $v(m, n, r)$  multicast network to be nonblocking under Strategy 3 is  $m \geq \Theta\left(n \frac{\log r}{\log \log r}\right)$ .*

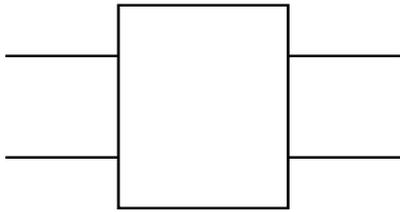
**Conjecture 1** *The necessary condition for a  $v(m, n, r)$  multicast network to be non-blocking under any strategy is  $m \geq \Theta\left(n \frac{\log r}{\log \log r}\right)$ .*

**Based on this conjecture,  $m = O\left(n \frac{\log r}{\log \log r}\right)$  is optimal for nonblocking  $v(m, n, r)$  multicast network.**

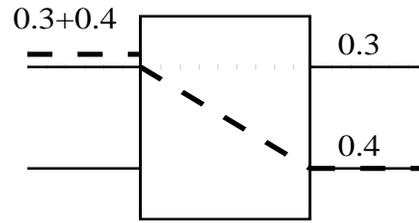
- **Generalization to multirate network**

- **Single rate connection:** A connection which uses up the entire bandwidth of the link carrying it.
- **Multirate connection:** A connection which can consume an arbitrary fraction of the bandwidth of the link carrying it.
- **Why design multirate networks?**  
More general than single rate network. Can be operated in packet switching manner.

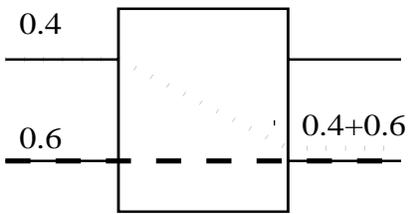
## – Multirate switch



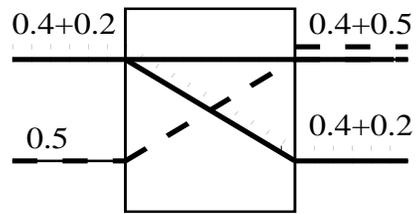
(a)



(b)



(c)



(d)

## – Previous work

### \* Single rate network

- A  $v(m, n, r)$  network [Clos '53] is non-blocking for permutation assignments if

$$m \geq 2n - 1$$

- A  $v(m, n, r)$  network [Yang and Masson '92] is nonblocking for multicast assignments if

$$m \geq 3(n - 1) \frac{\log r}{\log \log r}$$

### \* Multirate network

- A  $v(m, n, r)$  multirate network [Melen and Turner '89; Chung and Ross '91] is nonblocking for permutation assignment if

$$m > 2 \lfloor 1/b \rfloor (n - 1)$$

where  $b$  ( $0 < b \leq 1$ ) is the fraction of

bandwidth a minimum size packet occupies.

– More terminology:

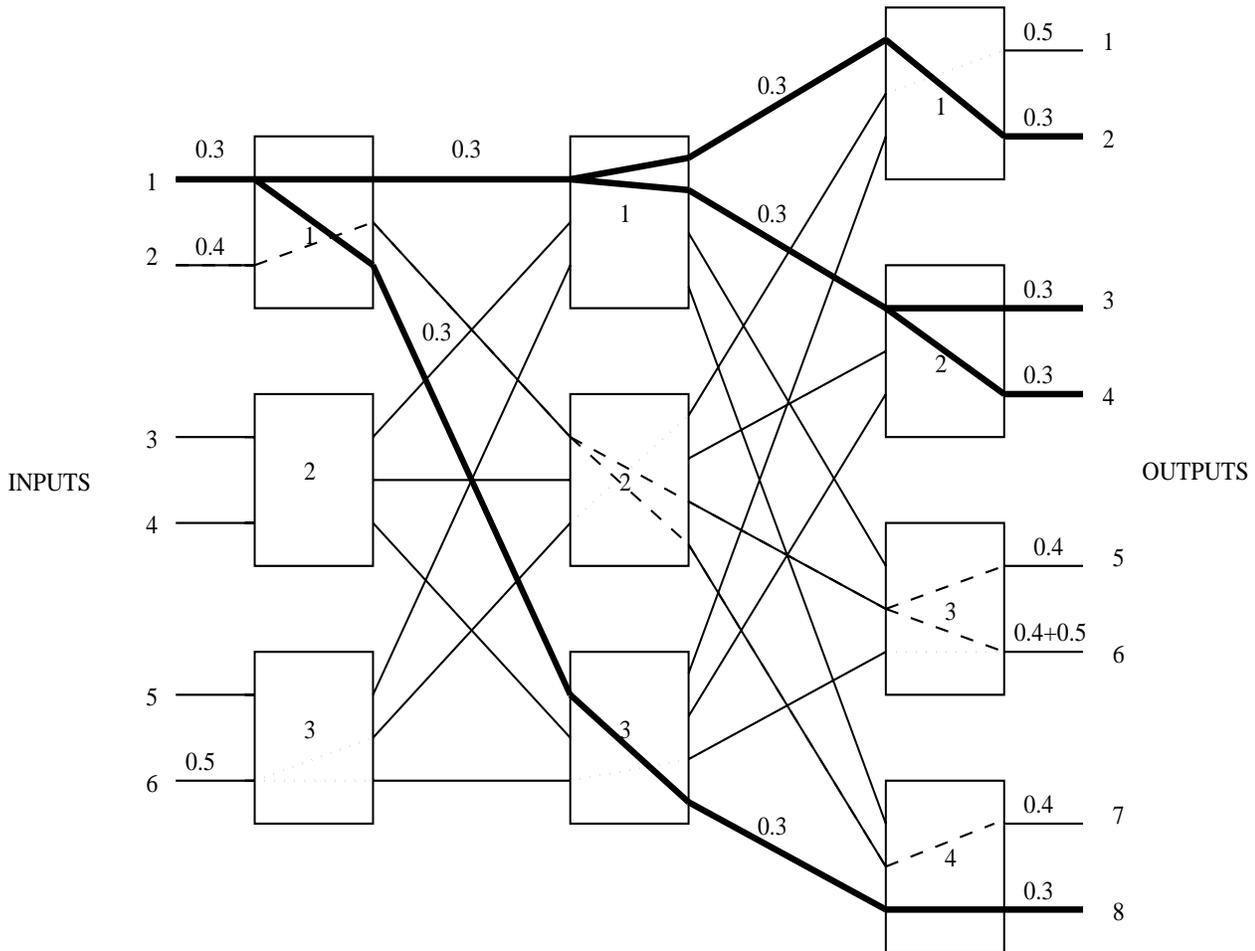
$\omega$ -idle input port or output port: an input port or an output port is  $\omega$ -idle,  $b \leq \omega \leq 1$ , if the fraction of available bandwidth of that port is at least  $\omega$ .

Multicast connection with weight  $\omega$ : A connection from an  $\omega$ -idle input port  $j$  to a set of  $\omega$ -idle output ports  $I_i$ , denoted as  $(I_j, \omega)$ , uses  $\omega$  fraction of bandwidth on all links carrying it.

Multirate multicast assignment: A maximum set of multirate multicast connections where the weight of all input ports and output ports involved is no more than 1.

**Available middle switches:** The available middle switches of an input connection  $(I_j, \omega)$  is the set of middle switches with link weight no more than  $1 - \omega$  to the input switch associated with input port  $j$ .

**Destination sets:** Destination set  $M_{j,\omega}$  is the set of output switches to which the middle switch  $j$  is providing connection paths with weights greater than  $1 - \omega$  from the input ports.



**A multirate multicast assignment in a  $v(3, 2, 3, 2, 4)$  network where  $(I_1, 0.3) = \{1, 2, 4\}$ .**

## ● Main results

Let  $T(n_1, \omega, x)$  denote the number of output links with weights great than  $1 - \omega$  in the input switch associated with the new connection request  $(I_i, \omega)$ , that is, the number of middle switches not available to the new connection request.

**Lemma 2** *For  $T(n_1, \omega, x)$ , we have three cases:*

**Case 1.** *If  $b > 1 - \omega$ , then*

$$T(n_1, \omega, x) = \lfloor 1/b \rfloor (n_1 - 1)x.$$

**Case 2.** *If  $b \leq 1 - \omega$  and  $1 - \lfloor \frac{1}{1-\omega} \rfloor (1 - \omega) < b$ , then*

$$T(n_1, \omega, x) \leq \left\lfloor \frac{1}{1-\omega} \right\rfloor (n_1 - 1)x + \left( x - \left\lfloor x / \left\lfloor \frac{1-\omega}{b} \right\rfloor \right\rfloor \right) U(n_1),$$

$$\text{where } U(n_1) = \begin{cases} 1 & \text{if } n_1 > 1 \\ 0 & \text{otherwise} \end{cases}$$

**Case 3.** *If  $b \leq 1 - \omega$  and  $1 - \lfloor \frac{1}{1-\omega} \rfloor (1 - \omega) \geq b$ , then*

$$T(n_1, \omega, x) \leq \left\lfloor \frac{n_1 - 1}{1 - \omega} \right\rfloor x + x.$$

### Lemma 3

$$\max_{b \leq \omega \leq 1} T(n_1, \omega, x) = \lfloor 1/b \rfloor (n_1 - 1)x.$$

**Lemma 4** *If a new connection request  $(I_i, \omega)$  is valid, then for each output switch  $k$ ,  $1 \leq k \leq r_2$ , there are at most  $T(n_2, \omega, 1)$  links with the weights greater than  $1 - \omega$  connecting to output switch  $k$  from the middle stage.*

**Lemma 5** *We can satisfy a new connection request  $(I_i, \omega)$ ,  $|I_i| = r_2$ , using some  $x$  ( $x \geq 1$ ) middle switches, say,  $i_1, \dots, i_x$ , from among the available middle switches of a  $v(m, n_1, r_1, n_2, r_2)$  network if and only if the current destination sets of these  $x$  middle switches are such that*

$$\bigcap_{j=1}^x M_{i_j, \omega} = \phi$$

**Lemma 6** *Assume that a  $v(m, n_1, r_1, n_2, r_2)$  network is in a state in which there exist at most  $n'$  connection paths with the weights greater than  $1 - \omega$ ,  $1 \leq n' \leq T(n_2, \omega, 1)$  to each of the output switches. Then the intersection of more than  $n'$   $M_{i, \omega}$ 's is empty.*

**Theorem 10** *A  $v(m, n_1, r_1, n_2, r_2)$  network is non-blocking for multirate multicast assignments if*

$$m > \max_{b \leq \omega \leq 1} \min_{1 \leq x \leq \min\{T(n_2, \omega, 1), r_2\}} \{T(n_1, \omega, x) + T(n_2, \omega, 1)r_2^{1/x}\}$$

**Corollary 3** *For the symmetrical  $v(m, n, r)$  multicast network, the nonblocking condition under multirate traffic model becomes*

$$m > \max_{b \leq \omega \leq 1} \min_{1 \leq x \leq \min\{T(n, \omega, 1), r\}} \{T(n, \omega, x) + T(n, \omega, 1)r^{1/x}\}.$$

*In particular, this condition can be written as*

$$m > 3 \lfloor 1/b \rfloor (n - 1) \frac{\log r}{\log \log r}$$

**Corollary 4** *Setting  $d = 1$ ,  $n_1 = n_2 = n$  and  $r_1 = r_2 = r$  in Corollary 1 yields*

$$m > 2\lfloor 1/b \rfloor (n - 1)$$

*which is the nonblocking bound on  $m$  associated with the one-to-one connection multirate  $v(m, n, r)$  network.*

**Corollary 5** *Setting  $b = 1$  in Corollary 3 yields  $m \geq 2n - 1$ , which is the nonblocking bound on  $m$  associated with the circuit switching Clos one-to-one connection network.*

Finally, the proofs for the theorems also implied an efficient routing algorithm for the multirate network.