Generalized Block Shift Network for Clusters

Yuzhong Sun, Xiaola Lin, Yi Pan, Rynson W. H. Lau, David A. Bader, and Paul Y. S. Cheung

***Abstract*—In this brief, a generalized topology of *block shift networks* (BSNs), named *generalized block shift network* (GBSN), is proposed for interconnection networks in clusters. The BSNs possess many desirable topological features, such as flexibility in node degree, small diameter and average distance, and easy VLSI implementation. However, the regular structure of each block in the BSN is not suitable for the networks in clusters that usually have arbitrary number of nodes. The proposed GBSN offers a balance between regularity and irregularity of the interconnection networks for clusters. We also analyze the embedding of the BSN into the GBSN, and discuss the versatility of the GBSN in terms of slowdown factors compared to the BSN.**

***Index Terms*—Block shift network (BSN), general block shift network (GBSN), routing, slowdown factor.**

I. Introduction

Clusters of workstations are fast being adopted as platforms for parallel computing. The advantages offered by clusters include high availability, scalable performance and low cost. High performance interconnection network is critical to the performance of a cluster [1], [2]. The network topology of a cluster is the pattern in which the nodes of the system are connected for communication. As cluster is a natural evolution from local area network (LAN), irregular topology is usually used in cluster as a consequence of the needs in LAN. In building clusters, switch-based interconnects with irregular topologies offer wring flexibility, scalability, and incremental expansion capability required in this environment. However, irregular topologies also lead to complicated routing protocol and poor and unbalanced link utilization [2].

The *block shift network* (BSN) is proposed by one of the authors in [1] for interconnection network in parallel systems. The BSNs possess many desirable topological features, such as flexibility in node degree and the number of nodes, small diameter and average distance, and easy VLSI implementation. However, the BSNs, like other popular network topologies such as hypercube and mesh [3], are regular networks that require the strict topological symmetry and integrity, which may not be suitable to be used in scalable clusters [2].

In this brief, we propose the *generalized block shift* network (GBSN) that consists of blocks with arbitrary number of nodes. The GBSN can be derived from a BSN in such a way that some nodes and the related links in one block in BSN can be removed simultaneously according to the selected parameters. In GBSN, all nodes may have different but bounded node degrees. We use the left-shift and right-shift links in [1] to construct the connections between blocks with different sizes. To facilitate message routing, each node has additional information about the parameters of shift operations in the block it resides. Through this extension, a GBSN can be used to represent a wide range of topologies including hypercube, shuttle-exchange, BSN, as well as certain irregular networks. By adjusting the parameters of the GBSN, such as the numbers of nodes in blocks, node degrees and connections, we can tune the performance of the underlying network to meet different system requirements. We thus achieve a desirable balance between regularity and irregularity, and provide better scalable performance for clusters, while retaining most of the desirable properties of the BSN.

II. Construction of the GBSN

The proposal of the BSN is motivated by the observation that, for a hypercube, link connections along all dimensions are not fully utilized in most cases [1], [4]. Thus a BSN only has part of the link connections along certain dimensions. A $\mathrm{BSN}(a, b)$ consists of three sets of links and each node has the address $a_{n-1}a_{n-2}\ldots a_1a_0$, where a_i, $0 \le i < n$, is a binary number [1]. The first set of links connects nodes to the nodes with addresses shifted cyclically b positions left in one step. These links connect the node $a_{n-1}a_{n-2}\ldots a_1a_0$ to the node $a_{n-b-1}a_{n-b-2}\ldots a_1a_0a_{n-1}\ldots a_{n-b}$. Data transferring over those links is called LEFT-SHIFT operations. Similarly, the second set of links connects nodes to those with addresses shifted cyclically b positions right in one step, called RIGHT-SHIFT links. Data transferring over these links is denoted by RIGHT-SHIFT operation. The two kinds of links will be referred as *shift* links or *block* links. The last set of links contains the connections over the rightmost b dimensions. One of the three connection methods in [1] is used to build the connection over the rightmost b dimensions. The links in the last set are named R-*change* links, and data transferring over these links is denoted as R-change operation. We also call the R-change links the *partial* links in terms of their constructions.

The connections between two nodes are called the connections on dimension i to j if the two nodes differ only in bits from i to j and are connected by the links on these dimensions [1]. Three variations of the connections on dimension i to j are introduced as *concurrent connection*, *sequential connection*, and *partial connection* in [1]. The former two are the special cases of the last one. A partial connection with parameter a is a connection method that can change a whole subsection for a sequence of binary values at a time. It can change a subsection into any pattern in one step by modifying up to a bits in the subsection. The section (bits i to j) has b bits, and can be divided into b/a subsections of a bits [1].

The BSN topology has strict requirements on the number of blocks and node degrees. The size of the BSN is 2^n [1] for some integer n. For example, the BSN(2, 2) in Fig. 1(a) has 16 nodes and it consists of four blocks, each having four nodes. The four blocks should have the same connections between the nodes in each block. Adjusting the parameters in $\mathrm{BSN}(a, b)$ can change the connections in one block or among blocks. For BSN(1, 2), the four nodes form a circle in each block of the BSN(1, 2) while the four nodes build a complete graph in each block of the BSN(2, 2). An example of the GBSN derived for the BSN is shown in Fig. 1(b).

The GBSN is denoted by $\mathrm{GBSN}(b\,|\,m_0, \ldots, m_{k-1})$ with k blocks, where b is the possible maximal number of bits to match the address of the source node to the destination node for a message crossing one shift link, and m_i denotes the number of nodes in the ith block. Each block obtains the parameter for the partial connection independent of all other blocks in the GBSN. The node address for one node in GBSN

Manuscript received April 5, 2000; revised November 3, 2001. The work of Y. Sun was supported in part by the University of New Mexico under the Albuquerque High Performance Computing Center (AHPCC) Research Associateship. The work of X. Lin was supported in part by RGC under Grant 7048/98E. The work of Y. Pan was supported in part by the National Science Foundation under Grant CCR-9211621, Grant OSR-9350540, Grant CCR-9503882, and Grant ECS-0 196 569. The work of D. A. Bader was supported in part by NSF under Grant CAREER 00-93039, Grant ITR 00-81404, Grant DEB 01-20709, and Grant ITR 01-21 377. This paper was recommended by Associate Editor K. Thulasiraman.

Y. Sun and D. A. Bader are with the Department of Electrical and Computer Engineering, University of New Mexico, Albuquerque, NM 87131 USA.

X. Lin and R. W. H. Lau are with the Department of Computer Science, City University of Hong Kong, Hong Kong.

Y. Pan is with the Department of Computer Science, Georgia State University, Atlanta, GA 30303 USA.

P. Y. S. Cheung is with the Department of Electric and Electronic Engineering, The University of Hong Kong, Hong Kong.

Publisher Item Identifier S 1057-7122(02)03121-5.

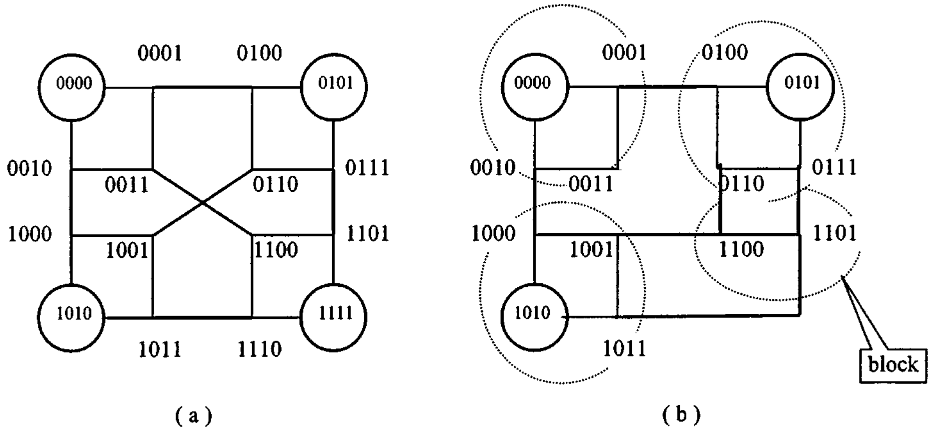

Fig. 1. Examples of the BSN(1, 2) and the $\mathrm{GBSN}(1\,|\,4,4,4,2)$.

is $a_{n-1}a_{n-2}\cdots a_1a_0$. In each block, the connections among nodes are determined by the parameter for the partial connection and the links among blocks are decided by shift operations. The blocks sizes are determined by the parameter vector $(m_0,\ldots,m_{k-1})$. The number of address bits is the sum of the two parts. The first part (*block* sub-address) is determined by the number of blocks, calculated by $\lceil \log k \rceil$, which corresponds to the leftmost bits. The second part (*internal* sub-address) is decided by the maximum number of nodes in blocks, calculated by $\max(\lceil \log m_i \rceil)$, $0 \le i \le k-1$. The second part corresponds to the rightmost bits.

Apparently, some links in the $\mathrm{BSN}(a,b)$ may no longer exist in the corresponding GBSN whose blocks may have different sizes. The address space in the GBSN may also not be continuous due to the changes in connections in the blocks. As a result, the LEFT-SHIFT, RIGHT-SHIFT, and partial connection operations may not always be valid. In the GBSN, a partial link within one block exists if and only if the two endpoints of the link satisfy the following conditions.1) The addresses of two nodes are valid. 2) The two nodes connect to each other by the partial connection method. The maximum block sub-address strategy is used to establish the connections between blocks. Each of nodes serves as a motivating node for establishing its block links according to the LEFT-SHIFT and RIGHT-SHIFT connection methods. Such a shift link can be marked by the motivating node's block address. Then, at one node, the shift links marked by the maximum block sub-addresses are valid. In fact, each node has two shift links. At most two shift links residing on one node can exist while all other shift links residing on the node are removed. The following algorithm constructs the $\mathrm{GBSN}(a_0,\ldots,a_{k-1}|b|m_0,\ldots,m_{k-1})$ (denoted as $\mathrm{GBSN}(b\,|\,m_0,\ldots,m_{k-1})$ for simplicity), where a_i is the parameter for the partial connection method in the ith block, and m_i is the size of the ith block. The algorithm description below is also served as the definition of the GBSN.

1) Construction Algorithm for a GBSN: **Input**: $b, m_0, \ldots, m_{k-1}$, a_i, $0 \le i \le k-1$.

Output: a $\mathrm{GBSN}(b\,|\,m_0,\ldots,m_{k-1})$.

Procedure:

1) Encode all nodes by determining the block and internal sub-addresses in ascending order.
2) Apply the partial connection method to each block to establish internal connections in the block. The link exists only if the codes of the two nodes are valid and reachable by each other. The partial connection method with the parameter a_i handles the rightmost $\lceil \log m_i \rceil$ bits on a block m_i, $0 \le i \le k-1$, $a_i \le \lceil \log m_i \rceil$. Each node should have at least two partial links.
3) In descending order of block addresses, each node applies the LEFT-SHIFT and RIGHT-SHIFT methods to form the two block links, each of which is marked by the block sub-address of the node. Such links exist if the two endpoints of one link exist using the shift operations. Regardless of the value a_i, it is guaranteed that there are at least two nodes having the complementary rightmost bits in each block and having at least one shift link on each of the two nodes.
4) At each node, the two possible block links marked by the two maximum block sub-addresses are preserved while all other links on the node are deleted.

Fig. 1(a) and (b) shows, respectively, the BSN(1,2) and $\mathrm{GBSN}(2\,|\,4,4,4,3)$ with the same parameter one for the partial connection method on the three blocks.

III. Topological Properties of GBSN

In this section, we examine certain topological properties of the proposed GBSN. The proofs of the following propositions are omitted and can be found in [5].

A. Network Size and Degree

For the $\mathrm{GBSN}(b\,|\,m_0,\ldots,m_{k-1})$, the number of the GBSN, $N = \sum_{i=0}^{k-1} m_i$. Theoretically, each block may have arbitrary number of nodes. There is no restriction in choosing the number of blocks for the construction of the GBSN.

In the $\mathrm{BSN}(a,b)$ with size $N = 2^n$, the network degree d is equal to $(2^a - 1)b/a + 2$. In the GBSN, the restriction that b/a is an integer is removed. Two factors are used to determine the degree of a node in the GBSN: the number of block links constituted by the LEFT- SHIFT and RIGHT-SHIFT, and the number of internal links built by the partial connection method with the parameter $\lceil \log m_i \rceil$ in the block m_i. LEFT-SHIFT and RIGHT-SHIFT add two block links to one node. The partial connection method increases the degree of the node by at most $(2^{\lceil \log m_i \rceil} - 1)b/\lceil \log m_i \rceil$. Therefore, the maximum network degree for the GBSN is $\max((2^{\lceil \log m_i \rceil} - 1)b/\lceil \log m_i \rceil + 2)$, $0 \le i \le k-1$, and the minimum node degree is two because each node has at least two partial links. Table I lists the number of nodes and node degree for GBSN, BSN and other popular topologies such as hypercube.

B. Network Diameter

The major obstacle in using switch-based irregular networks [6] is the unbound diameters of the networks. In GBSN, it is guaranteed that the diameter is bounded.

TABLE I
NUMBERS OF NODES AND NODE DEGREES FOR VARIOUS NETWORKS

Network	**Network Size (N)**	**Node Degree (D)**
n-hypercube	2^n	n
2-D mesh	$a \times b$	2, 3, 4
n-star graph	$n!$	$n-1$
MS(l, n)	$(n \cdot l + 1)!$	$n + l - 1$
BSN(a,b)	2^k	$(2^a - 1)\cdot b/a + 2,\ a \le b \le n$
GBSN($b\vert m_0,\ldots,m_{k-1}$)	$\sum_{i=0}^{k-1} m_i$	$1+1 \le D \le \max((2^{\lceil \log m_i \rceil} - 1)\cdot b/\lceil \log m_i \rceil + 2), 0 \le i < k$

Proposition 3.1: The diameter D in the $\mathrm{GBSN}(b|m_0,\ldots,m_{k-1})$ is

$$\max\left(\sum_{i=1}^{\lceil n/b\rceil-1}(b/\log m_i')\cdot\lceil n/b\rceil\right)\le D \le \max\left(\sum_{i=1}^{\lceil n/b\rceil-1} b\cdot\lceil n/b\rceil\right)$$

where $m_i' \in \{m_0,\ldots,m_{k-1}\}$, $m_i' \neq m_j'$ if $i \neq j$, i and $j \in \{1,\ldots,\lceil n/b\rceil - 1\}$, n is the maximal length of addresses of the nodes in the GBSN.

Compared to the diameter of the $\mathrm{BSN}(a,b)$ with the same size, the diameter of the $\mathrm{GBSN}(b|m_0,\ldots,m_{k-1})$ increases at most by a factor of b. It is shown in [1] that, when $n > 8$, the diameter of the $\mathrm{BSN}(b,b)$ is smaller than that of the hypercube with the same link complexity. In general, we should consider the effect of the partial connection method within one block with the m_i' nodes that can be measured by the parameter $\log m_i'$. Adjusting the two parameters provides a valid and simple way to balance the tradeoff between topology regularity and its diameter, which is especially important to the clusters with high availability.

C. Average Distance

The method for deriving the average distance of the BSN in [1] can be directly used to calculate the average distance of the GBSN. Constructing $\mathrm{BSN}(a,b)$ implies that b should be dividable by a in [1]. In the GBSN, the maximum value of the parameter for the partial connection method acting on a block with m_i nodes is $\lceil \log m_i \rceil$. In the best case, b is equal to $\lceil \log m_i \rceil$. That is, one shift can match at most b bits of the source address to the destination address for message delivering. The average distance $\bar{d}_{\mathrm{BSN}}$ of the $\mathrm{BSN}(a,b)$ with 2^n nodes is given as follows in [1]:

$$\bar{d}_{\mathrm{BSN}} = \left(2+\frac{b}{a}\right)\left(\frac{n}{b} - \frac{\frac{1}{2^b}}{1-\frac{1}{2^b}}\right) - \frac{2\left(1-\frac{1}{2^b}\right)}{2^{n-b}}. \tag{1}$$

In the best case, message crossing a block requires $2 + 1 = 3$ hops instead of $2 + (b/a)$. Replacing the $2 + (b/a)$ in (1), we get the lower bound of the average distance of the $\mathrm{GBSN}(b|m_0,\ldots,m_{k-1})$

$$\bar{d}_{\mathrm{BSN}} = F_{\mathrm{slowdown}} \cdot \left[3\cdot\left(\frac{n}{b} - \frac{\frac{1}{2^b}}{1-\frac{1}{2^b}}\right) - \frac{2\left(1-\frac{1}{2^b}\right)}{2^{n-b}}\right] \tag{2}$$

where F_{slowdown} is the slowdown factor of the routing algorithm, TWSR [1], for the BSN emulated in the GBSN. According to Proposition 4.1 in next section, $F_{\mathrm{slowdown}} = 1$ in this case. Thus, we get the lower bound on $\bar{d}_{\mathrm{GBSN}}$

$$\bar{d}_{\mathrm{BSN}} = 3\cdot\left(\frac{n}{b} - \frac{\frac{1}{2^b}}{1-\frac{1}{2^b}}\right) - \frac{2\left(1-\frac{1}{2^b}\right)}{2^{n-b}} \tag{3}$$

Consider the upper bound of $\bar{d}_{\mathrm{GBSN}}$. In the worst case of a message crossing a block, only a single bit of the source address is matched to that of the destination address. Therefore, $2 + (b/a)$ in (1) is replaced by $2 + b$. In this case, F_{slowdown} will reach its maximum value a. In the GBSN, the parameter a in the BSN is covered by the numbers of nodes in all blocks. That is, a is equivalent to $\beta = \min\{\lceil \log m_0 \rceil, \ldots, \lceil \log m_{k-1} \rceil\}$. Replacing the F_{slowdown} and $2 + (b/a)$ in (1), the upper bound on $\bar{d}_{\mathrm{GBSN}}$ is given as

$$\bar{d}_{\mathrm{BSN}} = \beta\cdot\left[(2+b)\left(\frac{n}{b} - \frac{\frac{1}{2^b}}{1-\frac{1}{2^b}}\right) - \frac{2\left(1-\frac{1}{2^b}\right)}{2^{n-b}}\right]. \tag{4}$$

Based on the average distance and diameter of the GBSN, the methods for the localized communication and basic data movement operations, such as finding minimum in a set, broadcast, ascend and descend, and data circulation, proposed for the BSN in [1] can be directly applied to the GBSN with a bound slowdown factor. In addition, the slowdown factor is controllable by dynamically changing the topology of the GBSN.

IV. COMMUNICATION ON THE GBSN

In the $\mathrm{BSN}(a,b)$, assume that n is the bit number of node address. In the *Two Way Shift Routing* (TWSR) for the BSN in [1], the n bits of each address are partitioned to $\lceil n/b \rceil$ sections. Each section corresponds to a block in which all nodes are connected by the partial connection method with parameter a [1]. Traversing from a block to a neighboring block is to right-shift the rightmost section of the address using shift links between blocks. The bits in the moved section can be updated using partial links in the block in advance, which is also called *update* in [1]. Such a shift operation can match one address section of a source to that of the destination for message delivering. Using right shift and updating section by section, the address of the source will match the address of the destination eventually in BSN. However, in GBSN, each section may have different number of bits determined by the number of nodes in one block. When a message traverses a block to its destination, at most b bits are updated to match the destination address. In fact, in the block, the partial connection method and the size of nodes in the block determine the number of possible matched bits in such a traverse. In the worst case, the whole partial connection operation cannot be executed in one block. However, the construction of the GBSN guarantees that at least one pair of nodes with the complementary rightmost bits exist in one block. Hence, passing through the block can still contribute to at least one new matched bit to the source address to enter a new block. Routing within one block remains the partial connection method regardless of the block size. Therefore, the TWSR algorithm in BSN can still be applied to GBSN.

Proposition 4.1: Any routing algorithm in the $\mathrm{BSN}(a,b)$ with the number of nodes 2^n can be emulated on the $\mathrm{GBSN}(b|m_0,\ldots,m_{k-1})$ for $1 \le a \le \max(\log m_0,\ldots,\log m_{k-1})$ and $2^n = \sum_{i=0}^{k-1} m_i$. The slowdown factor is $\min(a, a/x)$, where $x = \min(\log m_0,\ldots,\log m_{k-1})$.

The Proposition 4.1 implies that the maximum slowdown factor of emulating routing algorithm for the $\mathrm{BSN}(a,b)$ on the $\mathrm{GBSN}(b|m_0,\ldots,m_{k-1})$ is a and the minimal slowdown factor is one. Through emulating routing algorithms of the BSN, we can obtain simple algorithm to route a message between any pair of nodes in the GBSN in three time unit steps needed in the BSN.

Many popular networks such as hypercube, shuttle-exchange, and complete networks are instances of the BSN [1]. It is desirable that the BSN can be embedded into the GBSN with a small constant overhead. The *dilation* and *congestion* are often used to measure embedding overhead from one graph to another [7]. The dilation of one embedding refers to the maximum length of paths that are formed in the way that each node in the BSN is mapped to a node in the GBSN, and each link in the BSN is mapped to such a path in the GBSN. The maximum number of such paths that are mapped to a link in the GBSN is called the congestion of the embedding.

Proposition 4.2: The $\mathrm{BSN}(a,b)$ with the number of nodes 2^n can be embedded in the $\mathrm{GBSN}(b|m_0,\ldots,m_{k-1})$ for $1 \le a \le \max(\log m_0,\ldots,\log m_{k-1})$ and $2^n = \sum_{i=0}^{k-1} m_i$, with at most dilation a for mapping of partial links, and at most dilation $\max(\sum_{j=1}^{c}(m'_j/2))$ for mapping of shift links, and congestion $2a - 2 + \max(\sum_{j=1}^{c}(m'_j/2))$, where $c = \mathrm{Ham}(L_B \rightarrow L_{\mathrm{GB}})/b)$, $\mathrm{Ham}(L_B \rightarrow L_{\mathrm{GB}})$ is the Hamming distance of the identifiers of blocks where the two endpoints of L_{GB} are located, and $m'_0,\ldots,m'_{c-1}$ is the path to the mapping of a shift link L_B in the BSN to the shift link L_{GB} in GBSN.

For message routing in GBSN, we propose a simple routing algorithm for the GBSN without using counters introduced in routing algorithm TWSR for BSN in [1]. The proposed routing algorithm, named *GBSN routing*, works as follows. Given the current and destination addresses for delivering a message, the block identifiers of the current and destination addresses are compared. If they are identical, the message enters into the block where the destination is located. Using the partial connection method, the message can arrive at the destination similar to that in the TWSR. If they are different, then select the next block using right-shift acting on the rightmost section of the address for the current node, such that the block identifier of the next block is closer to that of the current block. Obviously, the partial connection method can be applied to find an appropriate node with the desired shift link. The process continues repeatedly until the message reaches the destination.

The justification of the proposed GBSN routing is that one right shift can change a section composed of at least one bit in the block identifier of the current block. The partial connection method guarantees that the rightmost bit of the section can be controlled to assign any binary values. Deadlock can be avoided by using virtual channel method in [3]. The formal description of the routing algorithm and justification can be found in [5].

V. Conclusion

Clusters of workstations are the popular platforms for parallel computer and require high flexibility in network topologies. It is desirable that such network topologies have the arbitrary network size and node degree as those in irregular networks, and small diameter and average distance like those in regular networks. In this brief, the GBSN is proposed to achieve a desirable tradeoff between irregular networks and regular networks in order to satisfy the high availability and scalability of the clusters. Furthermore, it is possible to tune the performance of the underlying GBSN in a cluster by changing its parameters to meet the changing requirements. The topological properties of the GBSN, such as the bounds on the diameters and average distances, and routing algorithm have also been presented and analyzed. One of the nice features of the GBSN is that the slowdown factors are controllable. The proposed GBSN could be an ideal candidate for building interconnection network in clusters.

References

[1] Y. Pan and H. Y. H. Chuang, “Properties and performance of the block shift networks,” *IEEE Trans. Circuits Syst. I*, vol. 44, Feb. 1997.

[2] J. Duato, A. Robles, F. Silla, and R. Beivide, “A comparison of router architectures for cirtual cut-through and wormhole switching in a NOW environment,” in *Proc. 1999 Int. Parallel Processing Symp.*, 1999.

[3] W. J. Dally and C. L. Seitz, “Deadlock-free message routing in multiprocessor interconnection networks,” *IEEE Trans. Comput.*, vol. 36, pp. 547–553, May 1987.

[4] Y. Pan, “Fault tolerance in the Block-Shift network,” *IEEE Trans. Rel.*, vol. 50, pp. 85–91, Mar. 2001.

[5] Y. Sun, X. Lin, Y. Pan, R. W. H. Lau, and D. A. Bader, “Generalized Block Shift Network for Clusters,” Dept. of Computer Science, City Univ. of Hong Kong, Hong Kong, Tech. Rep. 103, 2001.

[6] M. D. Schroeder *et al.*, “Autonet: A high-speed, self-configuring local area network using point-to-point links,” *IEEE J. Select. Areas Commun.*, vol. 9, Aug. 1991.

[7] Y. Sun, P. Y. S. Cheung, and X. Lin, “Recursive cube of ring: A new topology for interconnection networks,” *IEEE Trans. Parallel Distrib. Syst.*, vol. 11, pp. 275–286, Mar. 2000.