

Low Diameter Interconnections for Routing in High Performance Parallel Systems

Rami Melhem
Department of Computer Science
The University of Pittsburgh
melhem@cs.pitt.edu

Abstract

A new class of Low Diameter Interconnections (LDI) is proposed for high performance computer systems that are augmented with circuit switching networks. In these systems, the network is configured to match the communication patterns of applications, when these patterns exhibit temporal locality, and to embed a logical topology to route traffic that do not exhibit locality. The new LDI topology is a surprisingly simple directed graph which minimizes the network diameter for a given node degree and number of nodes. It can be easily embedded in circuit switching networks to route random traffic with high bandwidth and low latency.

Key words: Interconnection networks, circuit switching, fixed diameter graphs, directed graphs, deterministic routing, low diameter networks.

1. Introduction and motivation

Network interconnections for massively parallel systems have been extensively studied and many routing algorithms have been designed to efficiently route messages on these networks. The quality of routing algorithms is affected by two important characteristics of the network; its diameter and its node-degree. These two measures are tightly related, in the sense that a small node-degree implies a large network diameter, and small diameter leads to a large node degree. Specifically, if the maximum degree of the nodes in a directed graph is S , then any node cannot reach more than S other nodes in one hop, $S+S^2$ other nodes in at most two hops, and in general, $S+S^2+\dots+S^h$ other nodes in at most h hops. Hence, to be able to reach all $M-1$ other nodes in at most h hops, the value of h should satisfy $S+S^2+\dots+S^h \geq M-1$. In other words, for a given number of nodes, the relation between the node degree, S , and the diameter, h , is given by $(S^{h+1}-1)/(S-1) \geq M$, which is called the Moore's bound. It has been proven [2] that, except for the case of $S=1$ and $h=1$, there exists no graph that satisfies the equality in that bound. Asymptotically, however, the Moore's bound indicates that, in order to satisfy a fixed diameter of h , the node degree should be at least equal to $\sqrt[h]{M}$.

In many parallel applications, scalability is affected by communication latency, and consequently, by the diameter of the network. Hence, bounding the number of hops required to communicate between any pair of nodes in the network leads to better scalability. In this paper, we introduce a low diameter directed graph that minimizes the node degree. This work is motivated by the recent interest in using

circuit switching in scalable high performance computing systems [1,17,18,21]. Specifically, optical switches were recently proposed for establishing direct connections among processors to match the communication requirement of high performance applications [1,17]. This amounts to embedding the communication graph of the application into the switching network and is ideal for applications that exhibit communication locality. In order to clarify the concept, consider a 9-node system connected by three 9x9 cross-bar switches (called switching planes). If the communication pattern of an application requires the mesh-like interconnection shown in Figure 1(a), then the cross-bars can be set to match that pattern, as long as the number of cross-bars (switching planes) in the system is at least equal to the degree of the communication pattern. If the pattern for another application (or in another phase of the same application) requires the ring interconnection shown in Figure 1(b), then that pattern can be realized by appropriately setting the cross-bar switches. Note that time-division multiplexing of a single switch can be used to realize multiple switch settings, as proposed in [6,16]. Also note that other direct networks, such as fat trees or multistage networks can be used instead of the crossbars as switching planes to establish direct connections between the nodes in the system.

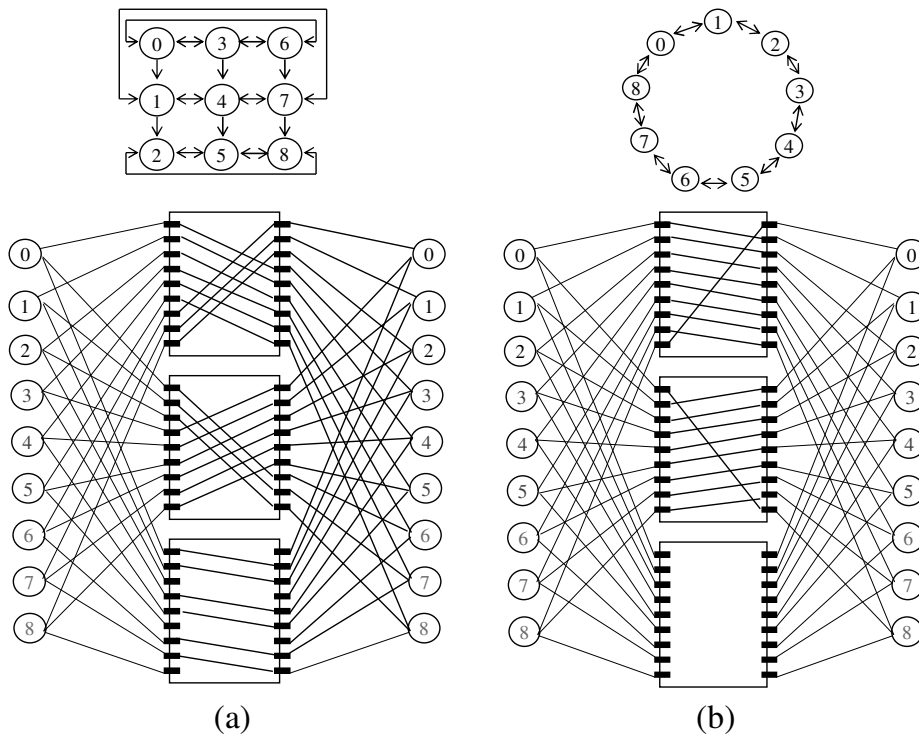


Figure 1 – A 9-node system connected by three crossbars. The cross-bars are configured to realize the shown interconnection topology. The circles on the left of the switches represent the output NICs (network interface cards) of the nodes and those on the right of the switches represent the input NICs of the nodes.

Establishing direct connections between communicating nodes increases the communication bandwidth and decreases its latency, compared to packet or wormhole switching. However, due to the large overhead of establishing connections, circuit switching is not suitable for communication that exhibit poor locality. The best way to deal with such communication in circuit switching networks is to use the switches to embed a suitable topology which will allow multi-hop routing of messages on this topology. Giving a specific node degree, implied by the number of available switching planes, the Low

Diameter Interconnection (LDI) introduced in this paper provides a means for finding the topology which minimizes the maximum number of hops for routing a message between any two nodes in the network.

The main result of this paper is to show that, for a given number of nodes, M , if M can be decomposed into $M = S^{h-1} G$, where $1 < G \leq S$, then it is possible to build a directed graph with M nodes whose diameter is h , and whose node degree is S , which, as shown earlier, is asymptotically optimal. In addition, the class of Low Diameter Interconnections can be easily embedded in circuit switched architectures and yields a simple routing algorithm which route messages in at most h hops. The practical impact of the restriction on the decomposability of the number of nodes is minimal since in most high performance computing systems, the number of nodes is a power of 2, which always allows for the above mentioned decomposability.

Directed graphs have been extensively studied in the literature. For instance, the Kautz graphs and the DeBruijn graphs are the best known classes of graphs for maximizing the number of nodes for given node degrees and diameters[3,5,14,15]. Both the Kautz and the DeBruijn graphs may be generated through line digraph iterations on complete directed graphs (with and without self-loops, respectively). Specifically, a node in a Kautz/DeBruijn graph is created for each link in the complete graph, and a link in the Kautz/DeBruijn graph is created for each path of length 2 in the complete graph. Figure 2(a) and (b) show examples of the two graphs when node degree = 2 and diameter = 3. Although optimally generated, the Kautz and DeBruijn graphs are not useful for finding the graph with the smallest diameter, given the number of nodes and the node degree. The partial line digraph technique proposed in [11] adds the flexibility of removing edges from the complete graphs before applying the line digraph iterations, thus obtaining graphs with small diameters for arbitrary number of nodes. The construction of the LDI graphs introduced in this paper is simpler than the partial line digraph technique. Moreover, the routing algorithm for LDI graphs is uniform and depends only on the parameters of the graph. Finally, there is an explicit formula for decomposing the links in LDI graphs such that their embeddings in circuit switched networks is straight forward. In Section 2, it will be shown that for the special case when the number of nodes is $M = S^h$, the LDI network is equivalent to the DeBruijn graph.

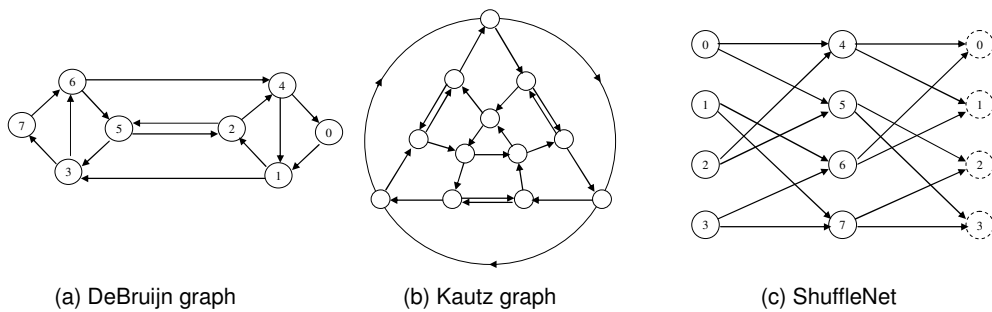


Figure 2 - Example graphs with node degree = 2 and diameter = 3.

Both directional and bi-directional ShuffleNets [12,13] have been proposed as interconnection networks. They are based on a generalization of the Perfect Shuffle connection patterns [19]. Specifically, a ShuffleNet with node degree, S , and parameter k , has $M = k S^k$ nodes arranged in k

columns of S^k nodes each, and the connections between the columns form a Perfect Shuffle (see Figure 2(c) for an example with $k=S=2$, where nodes 1, 2, 3 and 4 are duplicated for clarity). This leads to a network diameter of $2k-1$, which is larger than the diameter of the LDI network with the same number of nodes and node degree. For example, for $S=4$ and $M= 1024$ nodes, the diameter of the Shufflenet is 7 and that of the LDI is 5.

This paper is organized as follows: In the next section, the LDI graphs are defined and, in Section 3, their embeddings in circuit switched networks are demonstrated. In Section 4 two simple cases of LDI networks are presented to clarify the topologies and routing algorithms for these networks. In Section 5, the routing algorithm for a general class of LDI networks is derived. Concluding remarks are given in Section 6.

2. The LDI topology

For any $S>1$, $h>1$ and $S^{h-1} < M \leq S^h$, define the $LDI(M,S)$ topology as a directed graph, (V,E) , where $V = \{0, \dots, M-1\}$ is a set of M nodes, and E is the set of MS directed edges (links), given by

$$E = \{ \langle n, (Sn+L) \bmod M \rangle, \text{ for } n=0, \dots, M-1, \text{ and } L=0, \dots, S-1 \} \quad (1)$$

where $\langle u,v \rangle$ denotes a link from node u to node v . The link $\langle n, (Sn+L) \bmod M \rangle$ will be called the L^{th} link of node n . More descriptively, each node, n , in $LDI(M,S)$, has S output links connecting it to nodes $(Sn) \bmod M$, $(Sn+1) \bmod M$, ..., $(Sn+S-1) \bmod M$.

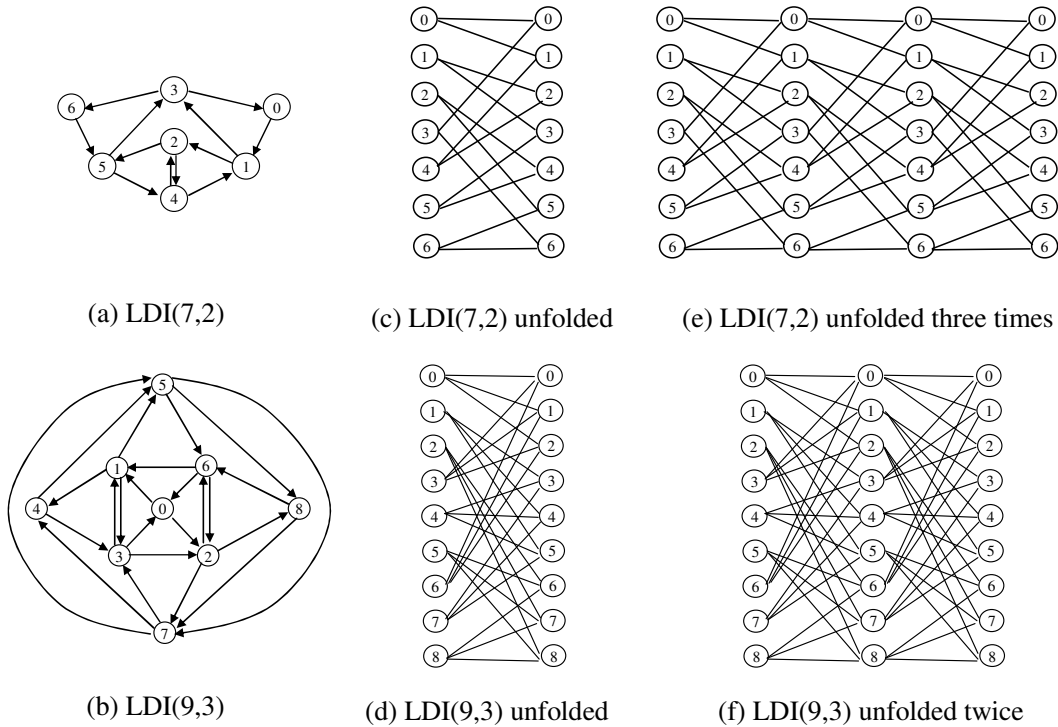


Figure 3 – Examples of Low Diameter Interconnections.

Figure 3 shows LDI(7,2) and LDI(9,3) as two examples of LDI networks with node degrees 2 and 3, respectively. Note that the definition in (1) includes links directed from a node to itself. These links are removed in Figure 3(a/b). Figure 3(c/d) shows the unfolded LDI graphs, where the circle labeled n on the left of the graph and the circle labeled n on the right of the graph represent the same node (or represent the output Network Interface Card, NIC, and the input NIC of node n , respectively). These unfolded graphs demonstrate the regularity of the connections. Further unfolding of the graphs, as in Figures 3(e/f), shows that any destination in LDI (7,2) can be reached from any source in 3 hops, and that any destination in LDI(9,3) can be reached from any source in 2 hops.

It is straight forward to argue that the diameter for LDI(M,S) is h if $S^{h-1} < M \leq S^h$. Specifically, from any node, n , the definition (1) implies that we can reach S consecutive nodes from n in one hop, where two nodes u and v are said to be consecutive if $v = (u+1) \bmod M$. In two hops from n , we can thus reach S^2 consecutive nodes and, in general, in h hops, we can reach up to S^h consecutive nodes. Given that $M \leq S^h$, then any of the M nodes can be reached in at most h hops. For example, from Figure 3(c), it can be seen that node 2 can reach nodes 4 and 5 in one hop, nodes 1, 2, 3 and 4 in two hops, and all seven nodes in three hops.

As indicated in the introduction, when $M = S^h$, LDI(M,S) is equivalent to the DeBruijn graph. Specifically, for any h and S , the DeBruijn graph with S^h nodes may be constructed by labeling the nodes in the graph using the S^h strings of h characters over an alphabet of S symbols, say $A = \{a_0, \dots, a_{S-1}\}$. That is, nodes may be labeled with strings of the form “ x_{h-1}, \dots, x_0 ”, where each $x_i, i=0, \dots, h-1$, is in A . Edges are then added from each node X , labeled by “ x_{h-1}, \dots, x_0 ”, to every node Y labeled by “ x_{h-2}, \dots, x_0, y ”, where y in A . That is Y is obtained by shifting the label of X one position to the left and adding any of the symbols of A at the right-most position. The equivalence to the LDI network is obtained by taking A as the set of integers $\{0, \dots, S-1\}$ and interpreting the label “ x_{h-1}, \dots, x_0 ” of a node X as the integer $n = \sum_{i=0}^{h-1} x_i S^i$ (here S^i is S raised to the power i , that is n is the integer $x_{h-1} \dots x_0$ in the base- S number system). Hence, the label “ x_{h-2}, \dots, x_0, y ” of node Y is interpreted as the integer $n' = \sum_{i=1}^{h-1} x_{i-1} S^i + y$. Using simple arithmetic shows that $n' = (Sn + w) \bmod S^h = (Sn + w) \bmod M$ which is the same relation governing the connectivity in LDI.

3. Decomposition of LDI into permutations

The links in LDI(M,S) can be grouped into S sets, $\sigma_y, y=0, \dots, S-1$, defined as follows:

$$\sigma_y = \{ \langle n, (Sn+L) \bmod M \rangle; n=0, \dots, M-1 \text{ and } L \text{ satisfies } y = (n \bmod S + L) \bmod S \} \quad (2)$$

Clearly, each set, σ_y , contains M links. Given that $0 \leq L < S$, any particular set, σ_y , contains exactly one link from a given node, n , since for the same value of n , two different values of L give two different values of $(n \bmod S + L) \bmod S$.

Next, it will be shown that any two links in the same set σ_y terminate at two different nodes. For this, define $Dest(n,y)$ as the destination of the link in σ_y whose source node is n . It can be shown that

$$Dest(n,y) = (Sn + (y - n \text{ div } S) \text{ mod } S) \text{ mod } M \quad (3)$$

Specifically, if link $\langle n, (Sn+L) \text{ mod } M \rangle$ is in σ_y , then $y = (n \text{ div } S + L) \text{ mod } S$. By substituting this value of y in (3) and using Rule 1 from the appendix, we get

$$\begin{aligned} Dest(n,y) &= [Sn + ((n \text{ div } S + L) \text{ mod } S - n \text{ div } S) \text{ mod } S] \text{ mod } M \\ &= [Sn + (n \text{ div } S + L - n \text{ div } S) \text{ mod } S] \text{ mod } M \\ &= [Sn + L \text{ mod } S] \text{ mod } M = (Sn + L) \text{ mod } M \end{aligned}$$

Now, noting that $0 \leq (y - n \text{ div } S) \text{ mod } S < S$, we conclude from (3) that if $n' \neq n$, then $Dest(n',y)$ cannot be equal to $Dest(n,y)$.

There are two consequences for proving that the M links in each set σ_y have different source nodes and different destination nodes. First, it proves that each of the in-degree and the out-degree of a node in the LDI graph is equal to S , and second, it shows that, because the sources and destinations of the links in each of the S set $\sigma_y, y=0, \dots, S-1$, form a permutation, the realization of the LDI connectivity through any S non-blocking switches is straight forward.

In addition to demonstrating the network topology and the routing algorithm, we use the example of $M=9$ and $S = 3$ (see Figure 3) to illustrate the decomposition of the links in the LDI topology into $S=3$ sets as specified in equation (2). Specifically,

$$\begin{aligned} \sigma_0 &= \{ \langle 0,0 \rangle, \langle 1,3 \rangle, \langle 2,6 \rangle, \langle 3,2 \rangle, \langle 4,5 \rangle, \langle 5,8 \rangle, \langle 6,1 \rangle, \langle 7,4 \rangle, \langle 8,7 \rangle \} \\ \sigma_1 &= \{ \langle 0,1 \rangle, \langle 1,4 \rangle, \langle 2,7 \rangle, \langle 3,0 \rangle, \langle 4,3 \rangle, \langle 5,6 \rangle, \langle 6,2 \rangle, \langle 7,5 \rangle, \langle 8,8 \rangle \} \\ \sigma_2 &= \{ \langle 0,2 \rangle, \langle 1,5 \rangle, \langle 2,8 \rangle, \langle 3,1 \rangle, \langle 4,4 \rangle, \langle 5,7 \rangle, \langle 6,0 \rangle, \langle 7,3 \rangle, \langle 8,6 \rangle \} \end{aligned}$$

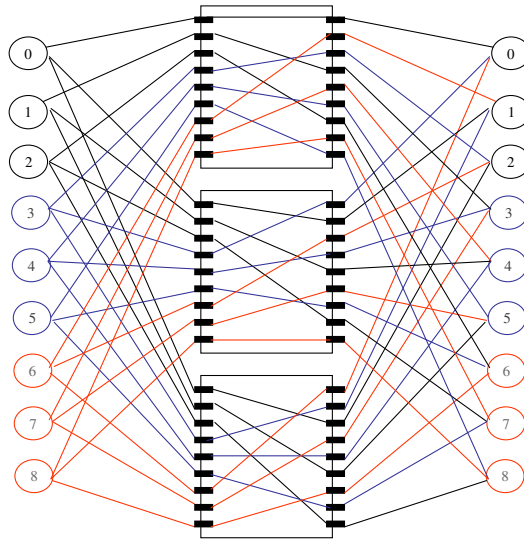


Figure 4. Switch setting to realize LDI(9,3).

It is straight forward to check that each set is a permutation, and thus the LDI connectivity can be accomplished by appropriately setting three cross bar switches as shown in Figure 4.

Although it was shown that the diameter of $\text{LDI}(M,S)$ is h if $S^{h-1} < M \leq S^h$, only the case where $M = S^{h-1} G$ for some $0 < G \leq S$ results in a simple and regular routing algorithm. In the next section, we first illustrate the routing algorithm and prove its correctness for the special case that leads to two-hop routing.

4. Two-hop routing in $\text{LDI}(M,S)$ for $M = SG$ and $G \leq S$

Let's revisit $\text{LDI}(9,3)$. From Figure 3(d/f), it is easy to check that there is a 2-hop path between any two nodes. Specifically, the nine nodes are divided into three groups (nodes 0,1,2 is one group, nodes 3,4,5 is another group and nodes 6,7,8 is a third group). Each group can connect to all the nine nodes. The first node in any group can reach nodes 0,1,2, the second node in any group can reach nodes 3,4,5 and the third node in any group can reach nodes 6,7,8. Given that any node, n , is connected to all the nodes in some group, say group Γ , then the first routing step (hop) for a connection from n to a destination, d , is to reach the particular node in Γ that is connected to the destination, d , namely a node $n^{(1)}$ that satisfies $n^{(1)} \bmod S = d \bmod S$. The second routing step (hop) is to reach the destination. This argument is formalized and generalized in the following theorem:

Theorem 1: Assuming that $M = S^2$, then for any source $0 \leq n < M$ and destination $0 \leq d < M$, a node $n^{(1)}$ can be found such that links $\langle n, n^{(1)} \rangle$ and $\langle n^{(1)}, d \rangle$ are in $\text{LDI}(M,S)$.

Proof: We will find two integers, $0 \leq \lambda^{(0)} < S$ and $0 \leq \lambda^{(1)} < S$, such that

$$(S n + \lambda^{(0)}) \bmod M = n^{(1)} \quad (4)$$

$$(S n^{(1)} + \lambda^{(1)}) \bmod M = d \quad (5)$$

Let $\lambda^{(0)} = d \bmod S$ and substitute this value in (4) to get $n^{(1)} = (S n + d \bmod S) \bmod M$. Then, substitute $n^{(1)}$ along with $\lambda^{(1)} = d \bmod S$, in the left hand side of (5) we get

$$\begin{aligned} (S n^{(1)} + \lambda^{(1)}) \bmod M &= [S ((S n + d \bmod S) \bmod M) + d \bmod S] \bmod M \\ &= [S^2 n + S (d \bmod S) + d \bmod S] \bmod M \\ &= d \end{aligned}$$

In the above simplification, we used Rule 1 from the appendix and the fact that $S (d \bmod S) + d \bmod S = d$. Clearly, Equation (4) specifies that $\langle n, n^{(1)} \rangle$ is in $\text{LDI}(M,S)$ and (5) specifies that $\langle n^{(1)}, d \rangle$ is in $\text{LDI}(M,S)$. ♦

Theorem 1 specifies a path of length 2 in $\text{LDI}(S^2, S)$ between any source node n and destination node d . Thus, it provides a simple routing algorithm from n to d . Namely,

- 1) From node n , use the L^h link, where $L = d \bmod S$ to reach node $n^{(1)} = (S n + d \bmod S) \bmod M$

- 2) From node $n^{(1)}$, use the L^{th} link, where $L = d \bmod S$ to reach node d .

Note that the above routing will always take 2-hops, even if there is a direct link from n to d . In order to take advantage of single hop routes, a test should be performed at n before the first hop to test if $(Sn) \bmod M \leq d < (Sn+S) \bmod M$, and if so, link L , where $L = (d - Sn) \bmod M$, should be used to route directly to d . Moreover, if $n = n^{(1)}$, then the first routing step can be eliminated.

After considering the case of $M = S^2$, the more general case of $M = S G$, for $G \leq S$, is considered. Figure 5 shows the unfolded connections for LDI(12,4) which contains $M=12$ nodes. The node degree for this case is $S = 4$ and the value of G is 3. The unfolded graph shows that any node can reach any other node in a maximum of two steps. Specifically, the nodes are divided into 3 groups of 4 nodes each and any node can be reached from any of the groups. In fact, because there are fewer groups than nodes within a group, then some nodes can be redundantly reached from other groups resulting in multiple paths between some sources and destinations. For example, two paths from node 5 to node 9 are shown in bold in Figure 5.

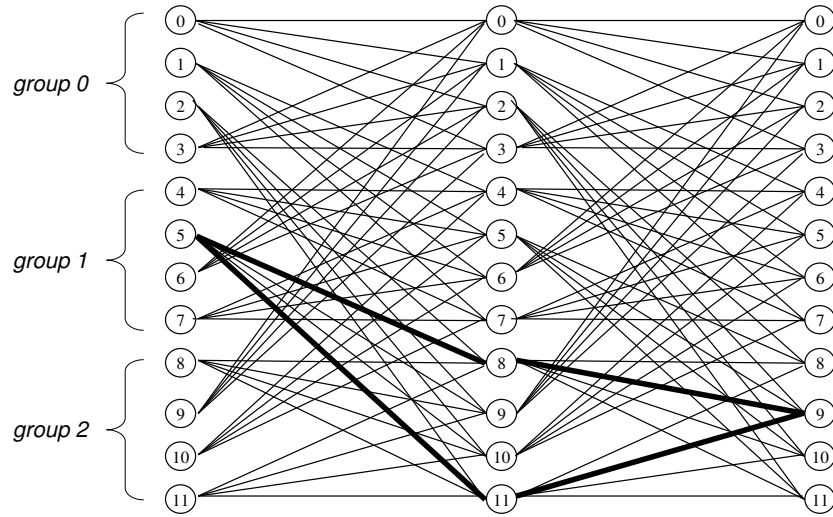


Figure 5 – The unfolded graph for LDI(12,4) showing multiple paths from node 5 to node 9.

The proof of the following theorem is similar to that of Theorem 1. It is omitted because it is a special case of the more general theorem proved in the next section.

Theorem 2: Assuming that $M = S G$, for $G \leq S$, then for any source $0 \leq n < M$ and destination $0 \leq d < M$, a node $n^{(1)}$ can be found such that links $\langle n, n^{(1)} \rangle$ and $\langle n^{(1)}, d \rangle$ are in LDI(M, S). ♦

The routing algorithm from any source node n to any destination node, d is as follows:

- 1) From node n , use the L^{th} link, where L is an integer that satisfies $(Sn + L) \bmod G = d \bmod S$ to reach node $n^{(1)}$
- 2) From node $n^{(1)}$, use L^{th} link, where $L = d \bmod S$ to reach node d .

Note that when $G < S$, there may be two values of L that satisfy $(Sn + L) \bmod G = d \operatorname{div} S$ in the first routing step, thus leading to the multiple path property.

Given the equivalence established in Section 2 between $\text{LDI}(S^2, S)$ and DeBruijn graphs and using previously established results about DeBruijn graphs[8,20], we can conclude that $\text{LDI}(S^2, S)$ remains connected in the presence of any $S-2$ faulty nodes, and that the diameter of the faulty network only increases from $h=2$ to $h=3$.

Although it seems that the multiple path property of LDI when $G < S$ may enhance its fault tolerance, it can be easily shown that $S - \lceil S/G \rceil$ faulty nodes can partition $\text{LDI}(SG, S)$, and clearly $\lceil S/G \rceil$ can be equal to, and even larger than 2. Specifically, the connectivity of $\text{LDI}(SG, S)$ for $G < S$ implies that there is at least one group of S nodes such that $\lceil S/G \rceil$ nodes in that group are only connected to nodes in the same group. Hence, those nodes can be isolated from the rest of the network if the other $S - \lceil S/G \rceil$ nodes in the group are faulty. For example, consider $\text{LDI}(15, 5)$ shown in Figure 6(a), which is composed of $G=3$ groups with $S=5$ nodes in each group. In this case $\lceil S/G \rceil = 2$ and, as shown in Figure 6(b), $S-2=3$ faults (namely in nodes 1, 2 and 4) can partition the networks such that nodes 0 and 3 are isolated from the other nodes. It can be shown, however, that in the presence of any $S - \lceil S/G \rceil - 1 = 2$ faulty nodes, the $\text{LDI}(15, 5)$ remains connected and its diameter increases from $h=2$ to $h=3$ (see Figure 6(c) for an example).

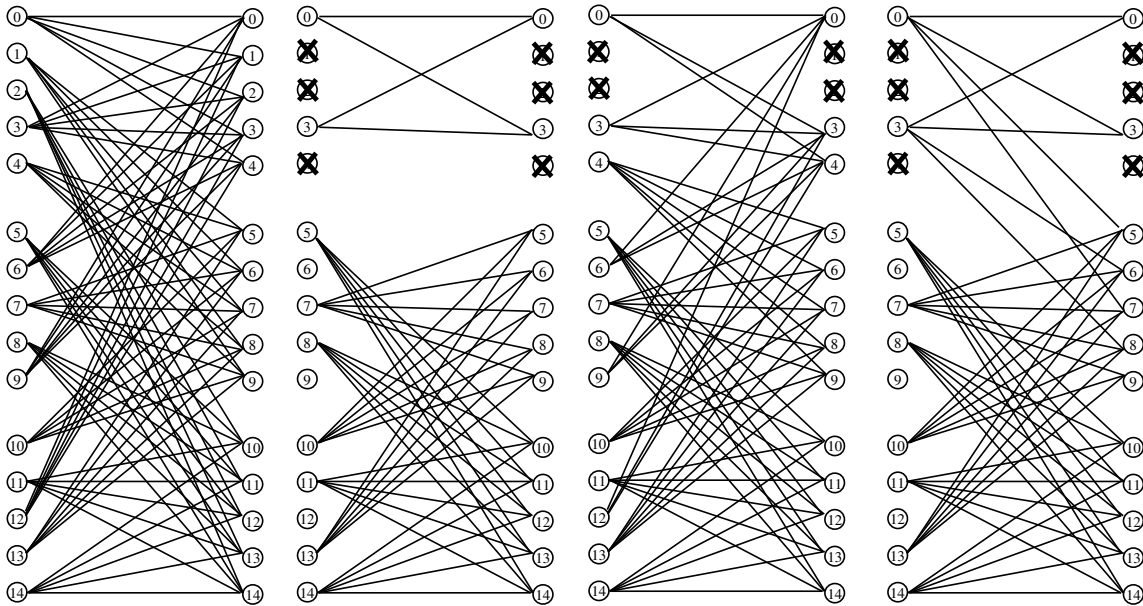


Figure 6 – $\text{LDI}(15,3)$: (a) with no faults, (b) with 3 faults that partition the network, (c) with 2 faults that do not partition the network, (d) after restoring connectivity for case (b)

At this point, it should be mentioned that, when non-blocking switches are used to embed an LDI network, as described in Section 3, the importance of network connectivity to fault tolerance diminishes. This is because of the capability to reconfigure the switches to restore connectivity. For example, in Figure 6(d), the connectivity of the network of Figure 6(b) is restored by the addition of links from nodes 0 and 3 to nodes 5, 6, 7 and 8. In fact, for any number of faults, f , in $\text{LDI}(M,S)$, it is always possible to restore connectivity by reconfiguring the switches to embed $\text{LDI}(M-f,S)$.

5. Routing in the general $\text{LDI}(S^{h-1} G, S)$ for $G \leq S$

In this section, Theorem 2 is generalized. Specifically, an h -hop route in $\text{LDI}(S^{h-1} G, S)$ will be found between any source node, n , and destination node, d . In order to simplify the notation, n and d will be denoted by $n^{(0)}$ and $n^{(h)}$, respectively.

Theorem 3: Assuming that $M = S^{h-1} G$ for some $G \leq S$ and some $h > 1$, then for any source node, $0 \leq n^{(0)} < M$, and destination node, $0 \leq n^{(h)} < M$, there are $h-1$ nodes, $0 \leq n^{(i)} < M$, $i = 1, \dots, h-1$, such that links $\langle n^{(i)}, n^{(i+1)} \rangle$, $i=0, \dots, h-1$, are in $\text{LDI}(M,S)$.

Proof: We will prove the theorem by finding explicit expressions for the links on an h -hop path from n to d . That is, finding $h-1$ integers between 0 and $M-1$, namely $n^{(1)}, \dots, n^{(h-1)}$ and h integers between 0 and $S-1$, namely $\lambda^{(0)}, \dots, \lambda^{(h-1)}$, that satisfy the following

$$(S n^{(i)} + \lambda^{(i)}) \bmod M = n^{(i+1)} \quad \text{for } i=0, \dots, h-1 \quad (6)$$

which proves that $\langle n^{(i)}, n^{(i+1)} \rangle$, $i=0, \dots, h-1$, are links in $\text{LDI}(M,S)$ forming an h -hop path from $n = n^{(0)}$ to $d = n^{(h)}$. In other words, we will solve (6) for $\lambda^{(0)}, \dots, \lambda^{(h-1)}$ and $n^{(1)}, \dots, n^{(h-1)}$ in terms of $n^{(0)}$ and $n^{(h)}$.

First, we divide both sides of equation (6) by S to obtain

$$[(S n^{(i)} + \lambda^{(i)}) \bmod M] \text{div } S = n^{(i+1)} \text{div } S, \quad \text{for } i=0, \dots, h-1$$

Using Rule 3 from the appendix and noting that $M = S^{h-1} G$, we get the following relations between the candidate nodes on the path from n to d :

$$n^{(i)} \bmod (S^{h-2} G) = n^{(i+1)} \text{div } S \quad \text{for } i=0, \dots, h-1 \quad (7)$$

The proof of the theorem follows directly from the proof of the following two lemmas.

Lemma 1: The values of $n^{(i)}$ that satisfy Equations (7) also satisfy the following equations:

$$n^{(i)} \bmod (S^{i-1} G) = n^{(h)} \text{div } S^{h-i} \quad \text{for } i=1, \dots, h-1 \quad (8)$$

Proof: It will be shown that (8) are true if (7) are true by backward induction on i . Clearly for $i=h-1$, equation (8) is equivalent to equation (7) which is true from the hypothesis. Next, assuming that (8) is true for $i = a$, where $1 < a \leq h-1$, we will prove that it is true for $i = a-1$. Specifically, the induction hypothesis is obtained by using $i=a$ in equation (8). That is

$$n^{(a)} \bmod (S^{a-1} G) = n^{(h)} \operatorname{div} S^{h-a}$$

By dividing both sides of the equation by S (i.e. taking $\operatorname{div} S$), we get

$$[n^{(a)} \bmod (S^{a-1} G)] \operatorname{div} S = n^{(h)} \operatorname{div} S^{h-a+1} \quad (9)$$

Now, applying Rule 3 from the appendix to the LHS of (9) and using equation (7) gives

$$\begin{aligned} [n^{(a)} \bmod (S^{a-1} G)] \operatorname{div} S &= [n^{(a)} \operatorname{div} S] \bmod (S^{a-2} G) \\ &= [n^{(a-1)} \bmod (S^{h-2} G)] \bmod (S^{a-2} G) \\ &= n^{(a-1)} \bmod (S^{a-2} G) \end{aligned}$$

Substituting back in (9) gives

$$n^{(a-1)} \bmod (S^{a-2} G) = n^{(h)} \operatorname{div} S^{h-a+1}$$

which shows that equation (8) is true for $i = a-1$, and thus completes the proof of Lemma 1. ♦

Lemma 1 specifies the relation that should hold between each of the intermediate nodes $n^{(1)}, \dots, n^{(h-1)}$ and the destination $n^{(h)}$. The following Lemma finds the links on the path connecting these nodes.

Lemma 2: With $n^{(1)}, \dots, n^{(h-1)}$ given by Equations (8), the values of $\lambda^{(0)}, \dots, \lambda^{(h-1)}$ that solve Equations (6) satisfy the following:

$$(S n^{(0)} + \lambda^{(0)}) \bmod G = n^{(h)} \operatorname{div} S^{h-1} \quad (10.a)$$

$$\lambda^{(i)} = (n^{(h)} \operatorname{div} S^{h-i-1}) \bmod S \quad \text{for } i = 1, \dots, h-1 \quad (10.b)$$

Proof: from (6) we get

$$[(S n^{(i)} + \lambda^{(i)}) \bmod M] \bmod (S^i G) = n^{(i+1)} \bmod (S^i G) \quad \text{for } i = 0, \dots, h-1$$

By applying Rule 2 from the appendix to the LHS we get:

$$(S n^{(i)} + \lambda^{(i)}) \bmod (S^i G) = n^{(i+1)} \bmod (S^i G) \quad \text{for } i = 0, \dots, h-1$$

Applying Lemma 1 to the RHS of the above equations gives

$$(S n^{(i)} + \lambda^{(i)}) \bmod (S^i G) = n^{(h)} \operatorname{div} S^{h-i-1} \quad \text{for } i = 0, \dots, h-1$$

Finally, leaving the equation for $i=0$ intact and taking $\bmod S$ of both sides of the equations for $i=1, \dots, h-2$, and applying Rule 2 gives,

$$(S n^{(0)} + \lambda^{(0)}) \bmod G = n^{(h)} \operatorname{div} S^{h-1}$$

$$(S n^{(i)} + \lambda^{(i)}) \bmod S = (n^{(h)} \operatorname{div} S^{h-i-1}) \bmod S \quad \text{for } i = 1, \dots, h-1$$

The proof of Lemma 2 follows directly. ♦

To complete the proof of Theorem 3, we have to argue that there exist values of $\lambda^{(0)}, \dots, \lambda^{(h-1)}$ between 0 and $S-1$ that satisfy equations (10). Clearly the value obtained from (10.b) fall in that range. Moreover, given that $G \leq S$, then there exist at least one value of $\lambda^{(0)}$ which is between 0 and $S-1$ and satisfies (10.a), thus completing the proof of Theorem 3. ♦

In addition to showing that an h -hop path exists from any node n to any other node d in $\text{LDI}(S^{h-1} G, S)$, Theorem 3 specifies the actual route, and thus a routing algorithm. Specifically,

The routing Algorithm from $n = n^{(0)}$ to $d = n^{(h)}$:

- 1) From the source node, $n^{(0)}$, send the message to the next node, $n^{(1)}$, on link $\lambda^{(0)}$, where $\lambda^{(0)}$ is an integer between 0 and $S-1$ which satisfies

$$(S n^{(0)} + \lambda^{(0)}) \bmod G = d \text{ div } S^{h-1}$$

- 2) For $i = 1, \dots, h-1$,

From the current node, $n^{(i)}$, send the message to the next node on link $\lambda^{(i)} = (d \text{ div } S^{h-i-1}) \bmod S$ ♦

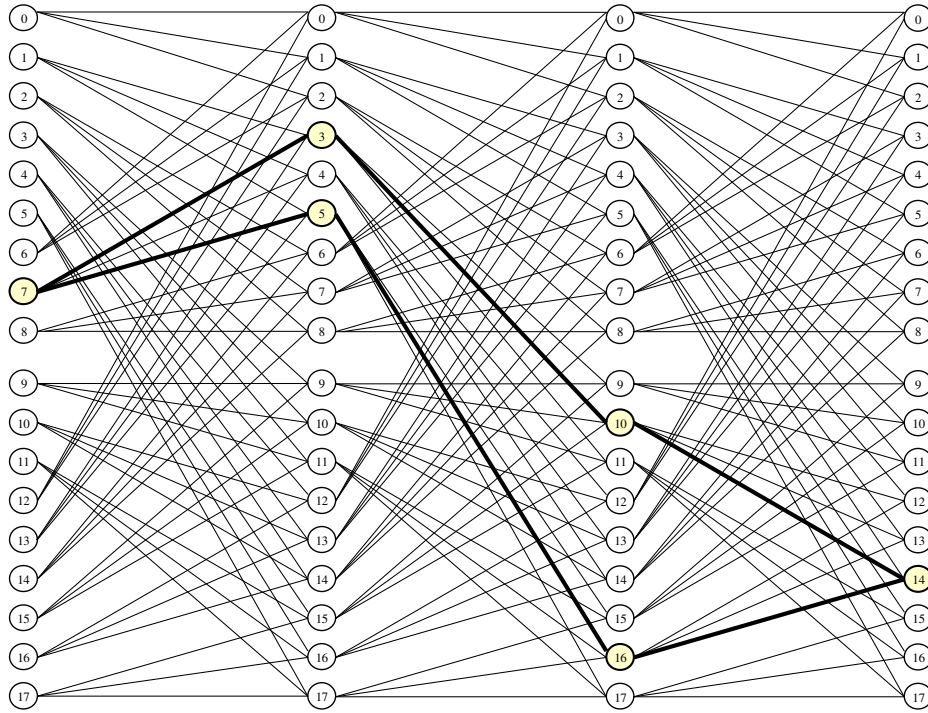


Figure 7 – 3-hop routing in $\text{LDI}(18,3)$ for which $S = 3$ and $G = 2$.

To clarify the routing algorithm with an example, consider the unfolded $\text{LDI}(18,3)$ graph shown in Figure 7 and apply the above algorithm for routing from node 7 to node 14. Given that, for this example, $S = 3$ and $G = 2$, the first routing step should be on link $\lambda^{(0)}$ which satisfies $(21 + \lambda^{(0)}) \bmod 2 = 14 \text{ div } 9$. That is $\lambda^{(0)} = 0$ or 2, which indicates that from node 7, we can either route on link 0 to node $(3*7 + 0) \bmod 18 = 3$ or on link 2 to node $(3*7 + 2) \bmod 18 = 5$. From either node, the second routing

step should be on link $\lambda^{(1)} = (14 \text{ div } 3) \text{ mod } 3 = 1$ (to either node 10 or 15, respectively), and the third routing step should be on link $\lambda^{(2)} = (14 \text{ div } 1) \text{ mod } 3 = 2$ to the destination, 14.

For the special case of $S = G$ that is $M = S^h$, the formula in the first routing step (Equation (10.a)) can be simplified resulting in $\lambda^{(0)} = d \text{ div } S^{h-1}$. Noting that $d \text{ div } S^{h-1} < S$, then the routing algorithm may be rewritten as follows to specify the unique path between the source and destination:

The simplified routing Algorithm for the case of $M = S^h$:

For $i = 0, \dots, h-1$,

From the current node, $n^{(i)}$, send the message to the next node on link
 $\lambda^{(i)} = (d \text{ div } S^{h-i-1}) \text{ mod } S$ ♦

In order to show the flexibility of the LDI topology, consider a system with 4096 nodes connected by multiple planes of circuit switches. For a specific number of switch planes, S , the switches can be configured to minimize the number of routing hops. Table 1 shows the maximum number of routing hops that result from embedding the LDI topology in the given number of switch planes, S .

Table 1 – The diameter of LDI(M, S) for $M = 4096$ nodes and different node degree, S .

Number of switching planes, S	The LDI topology to be used	The maximum number of routing hops	The average number of routing hops
64	LDI(4096, 64)	2	1.9
16	LDI(4096, 16)	3	2.9
8	LDI(4096, 8)	4	3.8
4	LDI(4096, 4)	6	5.6

Note that, in an h -hop routing, a message traverses the network h times. Hence, assuming that the maximum aggregate bandwidth of each switching plane is B , then the maximum aggregate effective bandwidth of using S switching planes with h -hop routing is $B S / h$. In other words, increasing the number of switching planes results in a larger than linear increase in the maximum bandwidth as well as a decrease in the maximum number of routing steps. For example, if a 4096-node system has 4 switching planes, then the embedding of LDI(4096,4) results in a diameter of 6 and maximum aggregate bandwidth of $4B/6 = 0.75B$. Increasing the number of planes to 8 allows the embedding of LDI(4096,8) which reduces the diameter to 4 while increasing the maximum bandwidth to $8B/4 = 2B$. Although this analysis is based on the diameter rather than the average number of hops, it is fairly accurate, especially in large systems where the maximum number of hops is not much larger than the average number of hops. This point is made in Table 1 by showing the average number of hops (obtained from simulations) for LDI networks with 4096 nodes.

The flexibility of the circuit switching architecture is also demonstrated by the ability to simultaneously embed predetermined connections to take advantage of communication locality and an LDI for routing random traffic. For example, with 4096 nodes and 8 switching planes, it is possible to dedicate 4 planes for establishing 2D torus connections, and use the remaining four planes for embedding LDI(4096,4) for routing random traffic with a maximum of 6 hops. Alternatively, it is possible to double the bandwidth for the torus connections by embedding two 2D torii in the 8 planes, while routing the random traffic on the torii with a maximum of 32 hops. The choice depends on the

ratio of mesh traffic to random traffic and on the delay tolerance of each type of traffic.

6. Concluding Remarks

The goal of the LDI directed graphs introduced in this paper is to minimize the network diameter for a given number of nodes and a given node degree. Specifically, for a given node degree, S , the LDI graph with M nodes, where $S^{h-1} < M \leq S^{h-1} G$ and $1 < G \leq S$, has a diameter of h , which is asymptotically optimal. The construction of the LDI networks is modular, and surprisingly simple. Moreover, for $M = S^{h-1} G$ the algorithm for routing between any two nodes in the LDI graph is straight forward. Deadlock-free routing in an LDI network with diameter h is guaranteed if h virtual channels are used [7].

The main advantage of LDI graphs is their straight forward embeddings in parallel computer systems that are interconnected by circuit switching networks. However, in addition to their applicability to routing random traffic in circuit switched networks, the LDI graphs may be applied whenever low diameter directed graphs are needed. Examples of such applications are overlay networks[4], WDM routing in light-wave networks[13] and the pre-scheduling of collective communication patterns[9,10]. It is worth noting that routing in LDI is performed using modular arithmetic operations, which leads to efficient hardware implementations.

Given that circuit switching networks are most beneficial for establishing connections that match application's regular communication patterns, and that LDI allows those networks to efficiently route random traffic, the next step is to study the efficient simultaneous routing of both regular and random traffic. Specifically, given a number, S , of switching planes in an architecture, if K planes are dedicated to regular traffic (e.g. mesh connections), an interesting problem is to find the configurations of the remaining $S-K$ planes such that random traffic is routed on all S planes with the minimum number of hops.

Fault tolerance is a very important issue in the design of interconnection networks for high performance systems. Using previously known results about DeBruijn graphs, we can conclude that, when $M = S^h$, $LDI(M, S)$ remains connected in the presence of any $S-2$ faulty nodes, and that the diameter of the faulty network only increases from S to $S+1$ [8,20]. A detailed study of the fault-tolerance capabilities of the LDI network when $G < S$ is left for future work.

Acknowledgment

I would like to thank Alex Jones, Ray Hoare, Eugen Schefeld and Zhu Ding for discussing and commenting on the work presented in the paper. I would also like to thank Seth Hornes for simulating the routing algorithms and demonstrating their correctness before the actual proofs were obtained. This work was partially supported by the IBM PERCS project supported by DARPA's HPCS program under contract NBCH3039004.

References

- [1] K. Baker, A. Benner, R. Hoare, A. Hoisie, A. Jones, D. Kerbyson, D. Li, R. Melhem, R. Rajamony, E. Schenfeld, G. Stunkel and P. Walker, "On the Feasibility of Optical Circuit Switching for High Performance Computing Systems". *Proceedings of Supercomputing '05*, Seattle, WA, November 2005.
- [2] W. Bridges and S. Toueg, "The Impossibility of Directed Moore Graphs", *J. of Combinatorial Theory*, vol 29, no 3, pp. 339-341, 1980.
- [3] F. Comellas and M. Fiol, "Vertex Symmetric Digraphs with Small Diameters", *Discrete Applied Mathematics*, vol 58, pp. 1-11, 1995.
- [4] V. Dalagiannis, A. Mauthe and R. Steinmetz, "Overlay Design Mechanisms for Heterogeneous, Large Scale, Dynamic P2P Systems". *Journal of Networks and System Management*", vol 12, no. 3, pp. 371-395, 2004.
- [5] N. De Bruijn, "A combinatorial problem," *Proc. Akademie Van Wetenschappen*, vol 49, part 2, pp.758-764, 1946.
- [6] Z. Ding, R. Hoare, A. Jones, D. Li, S. Shao, S. Tung, J. Zheng and R. Melhem, "Switch Design to Enable Predictive Multiplexed Switching in Multiprocessor Networks", *Proceedings of the International conference on Parallel and Distributed Processing (IPDPS)*, Denver, CO, April 2005.
- [7] J. Duato, S. Yalmanchili and L. Ni, "Interconnection Networks, an Engineering Approach", *IEEE Computer Society Press*, 1997.
- [8] A. Esfahanian and S. Hakimi, "Fault-Tolerant Routing in De Bruijn Communication Networks," *IEEE Transactions on Computers*, vol 34, no. 9, pp. 777-788, 1985.
- [9] A. Faraj and X. Yuan, "Message Scheduling for All-to-all Personalized Communication on Ethernet Switched Clusters", *The 19th IEEE International Parallel & Distributed Processing Symposium (IPDPS)*, Denver, Colorado, April 2005.
- [10] A. Faraj and X. Yuan, "Automatic Generation and Tuning of MPI Collective Communication Routines", *The 19th ACM International Conference on Supercomputing (ICS'05)*, Cambridge, MA, June 2005.
- [11] M. Fiol and A. Llado, "The Partial Line Digraph Technique in the Design of Large Interconnection Networks", *IEEE Trans. on Computers*, vol 41, no. 7, pp. 848-857, 1992.
- [12] M. Gerla, E. Leonardi, F. Neri and P. Palnati, "Routing in the Bidirectional ShuffleNet", *IEEE Transactions on Networking*, vol 9, no 1, pp. 91-102, 2001.
- [13] M. Hluchyi and M. Karol, "ShuffleNet: An Application of Generalized Perfect Shuffle to Multihop Lightwave Networks, *Proc. of INFOCOM'88*, New Orleans, LA, March 1988.

- [14] M. Imase and M. Itoh, "A Design for Directed Graphs with minimum Diameter," *IEEE Transactions on Computers*, vol 32, no. 8, pp. 782-784, 1983.
- [15] W. Kautz, "Bounds on Directed (d,k) graphs, "Theory of cellular Logic Networks and Machines", AFcKL-68-0668 Final report, 1968, pp.20-28.
- [16] C. Qiao, R. Melhem, "Reconfiguration with Time Division Multiplexed MINs for Multiprocessor Communications," *IEEE Trans. on Parallel and Distributed Systems*, vol. 5, no. 4, pp. 337-352, 1994.
- [17] J. Shalf, S. Kamil, L. Oliker and D. Skinner, "Analyzing Ultra-scale Application Communication Requirements for a Reconfigurable Hybrid Interconnect", *Proceedings of Supercomputig'05*. Seattle, WA, November 2005.
- [18] L. Smarr, A. Chien, T. DeFanti, J. Leigh, and P. Papadopoulos, "The OptIPuter", *Communications of the ACM*, vol 46, no 11, pp. 58-67, 2003.
- [19] H. Stone, "Parallel Processing with the Perfect Shuffle", *IEEE Trans. on Computers*, vol C-20, no 2, pp. 153-161, 1971.
- [20] M. Sridhar and C. Ragavendra, "Faut-Tolerant Networks Based on the De Bruijn Graph," *IEEE Transactions on Computers*, vol 40, no. 10, pp. 1167-1174, 1991.
- [21] M. Veeraraghavan, X. Zheng, H. Lee, M. Gardner, and W. Feng, "CHEETAH: Circuit-Switched High-Speed End-to-End Transport Architecture," *SPIE/IEEE Optical Networking and Computer Communications Conference (OptiComm)*, Dallas, TX, October 2003.

Appendix:

In this appendix, we prove a few rules of modulus arithmetic used in the papers.

Rule 1: $(a (b \text{ mod } k) + c) \text{ mod } k = (a b + c) \text{ mod } k$

Proof: let $b = x k + y$, where $0 \leq y < k$. Hence, LHS = $(a y + c) \text{ mod } k$, and RHS = $(a x k + a y + c) \text{ mod } k = (a y + c) \text{ mod } k \blacklozenge$

Rule 2: If $M = k g$, then $(a \text{ mod } M + b) \text{ mod } g = (a + b) \text{ mod } g$

Proof: let $a = x M + y$, where $0 \leq y < k$. Hence, LHS = $(y + b) \text{ mod } g$, and RHS = $(x k g + y + b) \text{ mod } g = (y + b) \text{ mod } g \blacklozenge$

Rule 3: $(a \text{ mod } (kg)) \text{ div } k = (a \text{ div } k) \text{ mod } g$

Proof: let $a = x k g + y k + z$, where $0 \leq y < g$ and $0 \leq z < k$. Hence, LHS = $(y k + z) \text{ div } k = y$, and RHS = $(x g + y) \text{ mod } g = y \blacklozenge$

Rami Melhem received a B.E. in Electrical Engineering from Cairo University in 1976, an M.A. degree in Mathematics and an M.S. degree in Computer Science from the University of Pittsburgh in 1981, and a Ph.D. degree in Computer Science from the University of Pittsburgh in 1983. He was an Assistant Professor at Purdue University prior to joining the faculty of the University of Pittsburgh in 1986, where he is currently a Professor of Computer Science and Electrical Engineering and the Chair of the Computer Science Department. His research interest include Real-Time and Fault-Tolerant Systems, Optical Networks, High Performance Computing and Parallel Computer Architectures. Dr. Melhem served on program committees of numerous conferences and workshops. He was on the editorial board of the IEEE Transactions on Computers and the IEEE Transactions on Parallel and Distributed Systems. He is serving on the advisory boards of the IEEE technical committees on Computer Architecture. He is the editor for the Springer Book Series in Computer Science and is on the editorial board of the Computer Architecture Letters, The International Journal of Embedded Systems and the Journal of Parallel and Distributed Computing. Dr. Melhem is a fellow of IEEE and a member of the ACM.