A New Architecture and A New Metric for Lightwave Networks

Arunabha Sen*
Department of Computer Science and Engineering
Arizona State University
Tempe, AZ 85287, USA

Subir Bandyopadhyay
School of Computer Science
University of Windsor
Windsor, Ontario N9B 3P4, CANADA

Bhabani P. Sinha
Advanced Computing and Microelectronics Unit
Indian Statistical Institute
Calcutta 700035, INDIA

Abstract

The notion of *logically routed network* was developed to overcome the bottlenecks encountered during the design of a large *purely optical network.* In the last few years the researchers have proposed the use of *Torus, Perfect Shuffle, Hypercube, de Bruijn graph, Kautz graph and Cayley graph* as an overlay structure on top of a purely optical network. All these networks have regular structures. Although regular structures have many virtues, it is often difficult in a realistic setting to meet these stringent structural requirements. In this paper we propose *Generalized Multi-Mesh*, a *semi-regular* structure, as an alternate to the proposed architectures. In terms of simplicity of interconnection and routing, this architecture is comparable to the *torus* network. However, the new architecture exhibits significantly superior topological properties than the torus. For example, whereas a two dimensional torus with N nodes has a diameter of $\Theta(N^{0.5})$, a Generalized Multi-Mesh network with the same number of nodes and links has a diameter of $\Theta(N^{0.25})$.

In this paper we also introduce a new metric, *Flow Number*, that can be used to evaluate topologies for optical networks. For optical networks, a topology with a smaller flow number is preferable, as it is an indicator of the number of wavelengths necessary for full connectivity. We show that the flow numbers of a two dimensional torus, a Multi-Mesh and a de Bruijn network are $\Theta(N^{1.5})$, $\Theta(N^{1.25})$ and $\Theta(Nlog\, N)$ respectively, where N is the number of nodes in the network. The advantage of the

*Corresponding author: Arunabha Sen, Telephone: 480-965-6153, Fax: 480-965-2751, e-mail: asen@asu.edu

Generalized Multi-Mesh over the de Bruijn network lies in the fact that, unlike the de Bruijn network, this network can be constructed for any number of nodes and is incrementally expandable.

Keywords: Optical networks, mesh, torus, de Bruijn graph, flow number, Multi-Mesh, multihop networks.

1 Introduction

Optical networks use interconnection of high-speed broadband fibers to transmit information. These networks support *lightpaths* which are end-to-end communication paths passing through one or more fibers, using one WDM channel per fiber. Optical networks can be divided into two classes - *single hop* and *multihop* networks. Scalabilty of purely optical networks is a limiting factor that hinders the design of single hop networks with a large number nodes. As shown in [**?**], whereas a bidirectional ring using wavelength routed point to point optical connections with seven nodes need only six transceivers and six optical wavelengths per station, the number of required transceivers and wavelengths grow to twenty one and sixty one respectively, if the number of nodes are increased to twenty two. The notion of *logically routed networks* (also known as *multihop networks*) was introduced to alleviate this problem [**?**].

An important objective in a single hop network design is to minimize the total number of carrier wavelengths needed for communication among various end stations. A multihop optical network, in order to realize high throughput, must ensure that the average delay due to buffering at the intermediate nodes is small. Ideally, a network topology should be incrementally scalable, should have a simple routing strategy and should be able to communicate even in presence of node and link failures. The topological design of optical networks, both for *logical* as well as *physical* connections have been considered in [?, ?, ?, ?] and include the *Torus* (or *Manhattan Street Network*) [?, ?], the *Perfect Shuffle* (or *ShuffleNet*) [?], the Hypercube [?], the de Bruijn graph [?], the Cayley graph (or *CayleyNet*) [?], the Kautz graph [?] and the *GEMNET* [?]. None of the proposed topologies satisfy all these orthogonal, if not conflicting, requirements [?, ?].

In this paper we introduce a new overlay architecture called the *Generalized Multi-Mesh (GM)* and compare its performance with two other architectures, the torus and the de Bruijn graph. The GM is a generalization of the *Multi-Mesh (MM)* architecture proposed for parallel processing systems [?, ?, ?, ?]. The MM structure is defined for n^4 nodes for any integer n. Since it is difficult to satisfy such a stringent requirement on the number of nodes, in a LAN/MAN/WAN environment, we have generalized this architecture so that it can be defined for any number of nodes N, $1 \leq N \leq n^4$ for some integer n. The new architecture is also incrementally expandable, i.e., an N node network can be extended to an $N+1$ node network without a major reconfiguration of the existing structure, if $N < n^4$. It may be noted that although the authors in [?] refer to the network discussed in their paper as the *Multi-Mesh* network, in the research community this network is known as the *Manhattan Street Network* [?, ?]. The *Multi-Mesh* network in our paper refers to the network proposed in [?, ?, ?, ?].

The architecture proposed in this paper can be used both as a physical as well as a logical topology for optical networks. In terms of simplicity of interconnection and routing, the architecture is comparable to the regular mesh and the torus. However, it exhibits significantly superior topological properties than the mesh and the torus. For example, whereas a two dimensional torus with N nodes has a diameter of $\Theta(N^{0.5})$, a

GM network with the same number of nodes and links has a diameter of $\Theta(N^{0.25})$.

Since two different lightpaths cannot be assigned the same WDM channel if they share the same fiber [**?**], the demand for the number of channels per fiber will be higher if many lightpaths have to share the same fiber. As the length of the source-destination path is bounded by the *diameter* (distance between two nodes of the network that are furthest from each other), one might be inclined to think that the demand for the number of channels per fiber will be low, if the network topology has a small diameter. However, this is not necessarily true; even in a small diameter network, because of the topology, a large number of lightpaths may be forced to travel through a small number of fibers, thereby increasing the demand for the number of channels in those fibers.

At present there is no graph theoretic metric to evaluate this aspect of a network topology. To address the need for such a metric, we have also introduced the notion of *flow number*. This metric supplements the currently used metrics such as the *diameter* and the *connectivity* of evaluation network topology. Although this metric is particularly useful in evaluating topologies for optical networks, the concept is useful in any network. A topology with a low flow number is more suitable for lightwave applications than a topology with a high flow number. We show that the flow numbers of a two dimensional torus, a Multi-Mesh and a de Bruijn network are $\Theta(N^{1.5})$, $\Theta(N^{1.25})$ and $\Theta(Nlog\ N)$ respectively, where N is the number of nodes in the network. The advantage of a Generalized Multi-Mesh over a de Bruijn network lies in the fact that unlike the de Bruijn network, this network can be constructed for any number of nodes and is incrementally expandable.

Figure 1: Blocks in a Multi-Mesh Network

2 The Generalized Multi-Mesh Architecture

The Generalized Multi-Mesh (GM) architecture is an extension of the Multi-Mesh (MM) interconnection network [**?**, **?**, **?**, **?**]. The interconnection pattern in MM is a modification of the simple mesh connection. In an $n \times n$ mesh, the processors are arranged in n rows and n columns. The MM uses this as a building block for the construction of the network. The idea is to use n^2 such blocks arranged again in n rows and n columns as shown in figure 1. Thus a MM network has exactly n^4 nodes. Each of these n^4 nodes are identified with a four tuple label (α, β, x, y). The first two, (α, β), identifies the block and the last two, (x, y), identifies the node within a block. Each of these co-ordinates can take a value between 1 and n (both inclusive).

Multi-Mesh Interconnection

The $n \times n$ nodes within each block are connected as a regular two dimensional mesh.

Figure 2: Multi-Mesh Network with 81 nodes (Not all interblock links are shown for clarity)

The interblock connections are made using the following rules:

Rule 1: $\forall \beta, 1 \leq \beta \leq n$, the node $(\alpha, \beta, 1, y)$ is connected to the node (y, β, n, α) where $1 \leq y, \alpha \leq n$.

Rule 2: $\forall \alpha, 1 \leq \alpha \leq n$, the node $(\alpha, \beta, x, 1)$ is connected to the node (α, x, β, n) where $1 \leq x, \beta \leq n$.

It may be observed that the interconnection rules given above generate a regular topology where the *degree* (the number of edges incident on a node) of each node is four. An MM network with eighty-one nodes is shown in figure **??**.

Generalized Multi-Mesh Interconnection

The *Generalized Multi-Mesh* (GM) network can be constructed for any specified number of nodes N for a specified block size n, $1 \leq N \leq n^4$. The construction mechanism is as follows: The GM will have $m = \lceil N/n^2 \rceil$ blocks. The blocks will be arranged in rows and columns in a block array as shown in figure **??**.

There will be $p = \lceil m/n \rceil$ rows, with each row having at most n blocks. The last row of blocks may be partially complete with $q = mod(m, n)$ blocks, $q > 0$. Every other row will have n complete blocks, each with $n \times n = n^2$ nodes. All blocks in the last row is a complete block except, possibly, the qth block which will have $N \bmod n^2$, $N \bmod n^2 > 0$ nodes. The qth block will have $r = \lceil N \bmod n^2/n \rceil$ rows with the first r - 1 rows having exactly n nodes. The rth row within the qth block will have s nodes, where $s = n$ if $N \bmod n = 0$, otherwise $s = N \bmod n$. This implies that the four tuple label of node N is (p, q, r, s). As in the MM network, the nodes within a block are connected as a regular two dimensional mesh. To describe the connections between the nodes in different blocks, the following four functions - $top(\alpha, \beta, i)$, $bottom(\alpha, \beta, i)$, $left(\alpha, \beta, i)$, $right(\alpha, \beta, i)$ are used. The parameters (α, β) in these functions identify the block and the parameter i identify a row or a column within the block.

(i) If the i-th column in the block has at least one node, then $top(\alpha, \beta, i)$ is the node in row 1, column i. Otherwise, it is not defined.

(ii) If the i-th column in the block has at least one node, then $bottom(\alpha, \beta, i)$ is the i-th column node in the highest indexed row that has at least i nodes. Otherwise, it is not defined.

(iii) If the i-th row in the block has at least one node, then $left(\alpha, \beta, i)$ is the node in row i, column 1. Otherwise, it is not defined.

(iv) If the i-th row in the block has at least one node, then $right(\alpha, \beta, i)$ is the highest indexed column node in row i. Otherwise, it is not defined.

Example: Let the incomplete block shown in figure **??** be in the third row and the

Figure 3: Blocks in a generalized Multi-Mesh network

Figure 4: Incomplete block with 10 nodes

second column of the array of blocks, i.e., $\alpha = 3, \beta = 2$. Then *top*(3, 2, 1) = 1, *top*(3, 2 ,4) = 4, *bottom*(3, 2, 1) = 9, *bottom*(3, 2, 4) = 8, $left$(3, 2, 1) = 1, $left$(3, 2, 3) = 9, $left$(3, 2, 4) is undefined, $right$(3, 2, 1) = 4, $right$(3, 2, 3) = 10 and $right$(3, 2, 4) is undefined.

The following observations follow directly from the definitions of top and bottom.

Observation 1: If a block has at least i nodes, $top(\alpha, \beta, i)$ as well as $bottom(\alpha, \beta, i)$ are defined. Otherwise, they are not defined.

Observation 2: If a block has x nodes, $i \leq x < n + i$, $top(\alpha, \beta, i) = bottom(\alpha, \beta, i)$.

Observation 3: If a block currently has x nodes, $i \leq x \leq n^2$, and we add more nodes to the block (keeping in mind that a block can never have more than n^2 nodes), the value of $top(\alpha, \beta, i)$ does not change but the value of $bottom(\alpha, \beta, i)$ will change if the number of nodes in the ith column chnages.

Observation 4: $top(\alpha, \beta, i)$ is defined if and only if $bottom(\alpha, \beta, i)$ is defined.

A similar set of observations can be made for $left(\alpha, \beta, i)$ and $right(\alpha, \beta, i)$. The interconnection rules between the nodes in different blocks are given next:

Rule 1.1: If the node $top(x, \beta, \alpha)$ exists, then $bottom(\alpha, \beta, x)$ is connected to the node $top(x, \beta, \alpha)$.

Rule 1.2: If the node $top(x, \beta, \alpha)$ does not exist and the node $top(\alpha+1, \beta, x)$ exists then $bottom(\alpha, \beta, x)$ is connected to the node $top(\alpha + 1, \beta, x)$.

Rule 1.3: If the node $top(x, \beta, \alpha)$ as well as the node $top(\alpha + 1, \beta, x)$ do not exist, then $bottom(\alpha, \beta, x)$ is connected to the node $top(i, \beta, x)$ where i is the lowest indexed row for which $top(i, \beta, x)$ is not connected using rule 1.1.

Rule 2.1: If the node $left(\alpha, x, \beta)$ exists then $right(\alpha, \beta, x)$ is connected to the node $left(\alpha, x, \beta)$.

Rule 2.2: If the node $left(\alpha, x, \beta)$ does not exist and the node $left(\alpha, \beta + 1, x)$ exists then $right(\alpha, \beta, x)$ is connected to the node $left(\alpha, \beta + 1, x)$.

Rule 2.3: If $left(\alpha, x, \beta)$ as well as $left(\alpha, \beta + 1, x)$ do not exist, then $right(\alpha, \beta, x)$ is connected to the node $left(\alpha, i, x)$, where i is the lowest indexed column for which $left(\alpha, i, x)$ is not connected using rule 2.1.

In general, a node having a 4 tuple (α, β, x, y) is connected to the nodes $(\alpha, \beta, x+1, y)$, $(\alpha, \beta, x, y+1)$, $(\alpha, \beta, x-1, y)$, $(\alpha, \beta, x, y-1)$, if they exist in block (α, β). If a node is not in the first row, the last row, the first column, or the last column, all these nodes exist and the node has a degree 4. If a node having a 4 tuple (α, β, x, y) is on the periphery (first row and/or last row and/or first column and/or last column) this node is used as the top

$$(\alpha, \beta, i)$$

and/or bottom(α, β, j) and/or left (α, β, k) and/or right(α, β, l) for some i, j, k, l. For each of these cases we have defined the node (possibly in a different block) that will be connected to the node (α, β, x, y). Therefore the degree of each node is 4.

It may be noted that the MM network is a special case of the GM network where the number of nodes $N = n^4$ is for some integer n. In this case, the nodes like $top(x, \beta, \alpha)$ and $left(\alpha, x, \beta)$ always exist and as such only the rules 1.1 and 2.1 are used for constructing the network. It may also be noted that each node in a GM network, just like in an MM network, has exactly 4 edges. A GM network with 44 nodes is shown in figure **??**.

Expansion in a Generalized Multi-Mesh Network

As mentioned earlier, the GM network can be constructed for any number of nodes N for a specified block size n, $1 \leq N \leq n^4$. In addition, a N node GM network can be expanded to a $N+1$ node GM network without a major reconfiguration if $N+1 \leq n^4$. The choice of n is somewhat critical because the effort for reconfiguration can be significant if $N + 1 > n^4$. This implies that n should be large enough so that this scenario is never encountered. The underlying assumption is that the network designer has a pretty good idea as to how large the network can grow in the future. If the designer feels that the network when it grows to its maximum size, will have at most M nodes, then during the design stage he/she chooses n as the smallest integer such that $M \leq n^4$. If n is chosen this way, the effort for reconfiguration during augmentation of the network from N nodes to $N + 1$ nodes is minimal. When a new node is added, clearly it has to be connected to four of its neighbors. Because of these new connections, some of the old connections may

Figure 5: Generalized Multi-Mesh network with 44 nodes

have to be removed. However, the number of old connections that have to be removed is at most 4 and in many cases even less than that. In figures **??**, **??** and **??** we have shown how a 44 node network can be expanded to 45 and 46 nodes respectively.

The assumption that the network designer would have a good idea as to how large the network can grow in the future seems reasonable as no network is designed with unlimited capacity. The reconfiguration needed for the augmentation of this network is simple because no more than 4 links have to be removed for this purpose.

In summary, the main advantages of the GM is that a) such a network can be constructed for any number of nodes, and b) it is *incrementally expandable* in the sense that an N node network can easily be augmented to an $N+1$ node network. This incremental expansion possibility of GM makes it very attractive for use in a LAN/MAN/WAN

Figure 6: Generalized Multi-Mesh network with 45 nodes

Figure 7: Generalized Multi-Mesh network with 46 nodes

environment where the number of nodes undergo frequent changes.

3 The Connectivity of Multi-Mesh Architecture

The survivability issues in IP over WDM networks is becoming increasingly important [?]. The survivability schemes that are considered belong to two broad classes, *protection* and *restoration*. A number of lightpath protection schemes have been proposed. In one scheme [?], a *dedicated* backup channel is provisioned for each primary channel requiring high availability. The lightpath used by the primary channel has to be completely *edge (node) disjoint* from the one used by the secondary channel, so that no single failure of link or optical cross-connect can disable both the primary and the secondary channel simultaneously. However, the number of edge (node) disjoint paths that exist between a pair of nodes is dependent on the topology of the network. In this section, we show that the MM network has the maximum connectivity that a regular graph with a node degree four can offer.

A MM network is a regular graph with a node degree four. The *node connectivity* between a pair of nodes in a graph is the number of *node disjoint* paths between these nodes. The *node connectivity* of a graph is the minimum of the node connectivity over all pairs of nodes. Since the degree of each node in a MM is four, it is clear that the upper bound of connectivity of MM is four. In this section, we show that, in a MM network, this upper bound is indeed realized, i.e., the connectivity of MM is four.

Theorem 1 *The connectivity of Multi-Mesh network is four* ($n \geq 3$).

Proof: We show that there exists four node disjoint paths between the nodes $(\alpha_1, \beta_1, x_1, y_1)$ and $(\alpha_2, \beta_2, x_2, y_2)$ for any value of $(\alpha_1, \beta_1, x_1, y_1)$ and $(\alpha_2, \beta_2, x_2, y_2)$. We consider a general situation with $\alpha_1 \neq \alpha_2$ and $\beta_1 \neq \beta_2$. The cases where (i) $\alpha_1 = \alpha_2$ and $\beta_1 \neq \beta_2$, (ii)

Figure 8: Disjoint paths between the nodes $(\alpha_1, \beta_1, x_1, y_1)$ and $(\alpha_2, \beta_2, x_2, y_2)$

$\alpha_1 \neq \alpha_2$ and $\beta_1 = \beta_2$ and (iii) $\alpha_1 = \alpha_2$ and $\beta_1 = \beta_2$ can be treated similarly and is not shown in this paper for brevity. The four disjoint paths between the nodes $(\alpha_1, \beta_1, x_1, y_1)$ and $(\alpha_2, \beta_2, x_2, y_2)$ are shown in figure **??** ($\forall i$, $\alpha_i = a_i$ and $\beta_i = b_i$ in the figure). The four node disjoint paths are as follows:

Path 1: $(\alpha_1, \beta_1, x_1, y_1) \rightarrow (\alpha_1, \beta_1, x_1, n) \rightarrow (\alpha_1, \beta_1, \beta_3, n) \rightarrow (\alpha_1, \beta_3, \beta_1, 1) \rightarrow (\alpha_1, \beta_3, n, 1) \rightarrow (\alpha_1, \beta_3, n, \alpha_2) \rightarrow (\alpha_2, \beta_3, 1, \alpha_1) \rightarrow (\alpha_2, \beta_3, 1, n) \rightarrow (\alpha_2, \beta_3, \beta_2, n) \rightarrow (\alpha_2, \beta_2, \beta_3, 1) \rightarrow (\alpha_2, \beta_2, x_2, 1) \rightarrow (\alpha_2, \beta_2, x_2, y_2)$.

Path 2: $(\alpha_1, \beta_1, x_1, y_1) \rightarrow (\alpha_1, \beta_1, x_1, 1) \rightarrow (\alpha_1, \beta_1, \beta_3, 1) \rightarrow (\alpha_1, \beta_3, \beta_1, n) \rightarrow (\alpha_1, \beta_3, 1, n) \rightarrow (\alpha_1, \beta_3, 1, \alpha_2) \rightarrow (\alpha_2, \beta_3, n, \alpha_1) \rightarrow (\alpha_2, \beta_3, n, 1) \rightarrow (\alpha_2, \beta_3, \beta_2, 1) \rightarrow (\alpha_2, \beta_2, \beta_3, n) \rightarrow (\alpha_2, \beta_2, x_2, n) \rightarrow (\alpha_2, \beta_2, x_2, y_2)$.

Path 3: $(\alpha_1, \beta_1, x_1, y_1) \rightarrow (\alpha_1, \beta_1, n, y_1) \rightarrow (\alpha_1, \beta_1, n, \alpha_3) \rightarrow (\alpha_3, \beta_1, 1, \alpha_1) \rightarrow (\alpha_3, \beta_1, 1, n) \rightarrow (\alpha_3, \beta_1, \beta_2, n) \rightarrow (\alpha_3, \beta_2, \beta_1, 1) \rightarrow (\alpha_3, \beta_2, n, 1) \rightarrow (\alpha_3, \beta_2, n, \alpha_2) \rightarrow (\alpha_2, \beta_2, 1, \alpha_3) \rightarrow (\alpha_2, \beta_2, 1, y_2) \rightarrow (\alpha_2, \beta_2, x_2, y_2)$.

Path 4: $(\alpha_1, \beta_1, x_1, y_1) \rightarrow (\alpha_1, \beta_1, 1, y_1) \rightarrow (\alpha_1, \beta_1, 1, \alpha_3) \rightarrow (\alpha_3, \beta_1, n, \alpha_1) \rightarrow (\alpha_3, \beta_1, n, 1) \rightarrow (\alpha_3, \beta_1, \beta_2, 1) \rightarrow (\alpha_3, \beta_2, \beta_1, n) \rightarrow (\alpha_3, \beta_2, 1, n) \rightarrow (\alpha_3, \beta_2, 1, \alpha_2) \rightarrow (\alpha_2, \beta_2, n, \alpha_3) \rightarrow (\alpha_2, \beta_2, n, y_2) \rightarrow (\alpha_2, \beta_2, x_2, y_2)$.

We have assumed that if $\alpha_3 = n$ then $\beta_3 \neq n$ and vice-versa.

It can easily be verified that these paths are node disjoint. This proves the theorem.

4 The Diameter of Generalized Multi-Mesh

In a multihop network, the communication delay may be proportional to the number of intermediate optical cross-connects the packets have to pass through on their journey from the source to the destination. As a consequence, in such an environment, a topology with a small diameter is quite desirable [?].

Suppose that the source node s is specified by $(\alpha_s, \beta_s, x_s, y_s)$ and the destination node d is specified by $(\alpha_d, \beta_d, x_d, y_d)$. We show that the any destination node d can be reached from any source node s with at most $2.5n + 1$ hops.

Theorem 2 *The diameter of a Generalized Multi-Mesh network at most* $2.5n + 1$.

Proof: Without loss of generality, we assume that $\alpha_s \leq \alpha_d$. We consider the following cases:

Case 1.1 ($\alpha_d < p$): In [?] it was shown that the diameter of a Multi-Mesh when $N = n^4$ is $2n$. Two paths from the source $(\alpha_s, \beta_s, x_s, y_s)$ to the destination $(\alpha_d, \beta_d, x_d, y_d)$ were considered, one through the block (α_s, β_d) and the other through the block (α_d, β_s). It was shown that the sum of the lengths of these two paths is $4n$. As a result, at least one of the paths must be of length $2n$ or less. The same argument holds in this case $(\alpha_d < p)$, because the blocks (α_s, β_d) and (α_d, β_s) are complete with n^2 nodes.

Case 2.1 ($\alpha_d = p, x_d < n/2, y_d < p$): In this case a path from the source s to the destination d can be constructed as follows: $(\alpha_s, \beta_s, x_s, y_s) \rightarrow \cdots \rightarrow (y_d, \beta_d, n, p) \rightarrow (p, \beta_d, 1, y_d) \rightarrow (p, \beta_d, 2, y_d) \rightarrow \cdots \rightarrow (p, \beta_d, x_d, y_d) = (\alpha_d, \beta_d, x_d, y_d)$. All the nodes and links needed to construct such a path exist. From case 1.1, we can infer that the length of the path from $(\alpha_s, \beta_s, x_s, y_s) \rightarrow \cdots \rightarrow (y_d, \beta_d, n, p)$ is at most $2n$. The length of the path from $(y_d, \beta_d, n, p) \rightarrow (p, \beta_d, 1, y_d) \rightarrow (p, \beta_d, 2, y_d) \rightarrow \cdots \rightarrow (p, \beta_d, x_d, y_d)$ is less than $n/2$. Therefore, the length of the path from the source s to the destination d is at most $2.5n$.

Case 2.2 ($\alpha_d = p, x_d < n/2, y_d = p$) : In this case a path from the source s to the destination d can be constructed as follows: $(\alpha_s, \beta_s, x_s, y_s) \rightarrow \cdots \rightarrow (p-1, \beta_d, n, p) \rightarrow$

$(p, \beta_d, 1, p-1) \rightarrow (p, \beta_d, 2, p-1) \rightarrow \cdots \rightarrow (p, \beta_d, x_d, p-1) \rightarrow (p, \beta_d, x_d, p) = (\alpha_d, \beta_d, x_d, y_d)$. By the same argument as in case 2.1, the length of the path from the source s to the destination d is at most $2.5n + 1$.

Case 2.3 ($\alpha_d = p, x_d < n/2, y_d > p$) : In this case a path from the source s to the destination d can be constructed as follows: $(\alpha_s, \beta_s, x_s, y_s) \rightarrow \cdots \rightarrow (p-1, \beta_d, n, y_d) \rightarrow (p, \beta_d, 1, y_d) \rightarrow (p, \beta_d, 2, y_d) \rightarrow \cdots \rightarrow (p, \beta_d, x_d, y_d) = (\alpha_d, \beta_d, x_d, y_d)$.

There is a link from the node $(p-1, \beta_d, n, y_d) = bottom(p-1, \beta_d, y_d)$ to the node $(p, \beta_d, 1, y_d) = top(p, \beta_d, y_d)$, because the node $(y_d, \beta_d, 1, p-1) = top(y_d, \beta_d, p-1)$ does not exist, as $y_d > p$ and p is the highest indexed row, in the row of blocks that contains some nodes. As the length of the path from $(\alpha_s, \beta_s, x_s, y_s)$ to the node $(p-1, \beta_d, n, y_d)$ is at most $2n$ (Case 1.1) and the length of the path from $(p, \beta_d, 1, y_d)$ to $(\alpha_d, \beta_d, x_d, y_d)$ is less than $n/2$, the length of the path from $(\alpha_s, \beta_s, x_s, y_s)$ to $(\alpha_d, \beta_d, x_d, y_d)$ is at most $2.5n$.

Case 3.1 ($\alpha_d = p, x_d \geq n/2, y_d < p$): In this case a path from the source s to the destination d can be constructed as follows: $(\alpha_s, \beta_s, x_s, y_s) \rightarrow \cdots \rightarrow (y_d, \beta_d, 1, p) \rightarrow bottom(p, \beta_d, y_d) \rightarrow \cdots \rightarrow (p, \beta_d, x_d, y_d) = (\alpha_d, \beta_d, x_d, y_d)$.

As the length of the path from $(\alpha_s, \beta_s, x_s, y_s)$ to the node $(y_d, \beta_d, 1, p)$ is at most $2n$ (Case 1.1) and the length of the path from $bottom(p, \beta_d, y_d)$ to $(\alpha_d, \beta_d, x_d, y_d)$ is less than $n/2$, the length of the path from $(\alpha_s, \beta_s, x_s, y_s)$ to $(\alpha_d, \beta_d, x_d, y_d)$ is at most $2.5n + 1$.

Case 3.2 ($\alpha_d = p, x_d \geq n/2, y_d = p$): In this case a path from the source s to the destination d can be constructed as follows: $(\alpha_s, \beta_s, x_s, y_s) \rightarrow \cdots \rightarrow (p-1, \beta_d, 1, p) \rightarrow bottom(p, \beta_d, p-1) \rightarrow \cdots \rightarrow (p, \beta_d, x_d, p-1) \rightarrow (p, \beta_d, x_d, p) = (\alpha_d, \beta_d, x_d, y_d)$.

As the length of the path from $(\alpha_s, \beta_s, x_s, y_s)$ to $(p-1, \beta_d, 1, p)$ is at most $2n$ and the length of the path from $bottom(p, \beta_d, p-1)$ to $(p, \beta_d, x_d, p-1)$ is at most $n/2 - 1$, the length of the path from the source s to the destination d is at most $2.5n + 1$.

Case 3.3 ($\alpha_d = p, x_d \geq n/2, y_d > p$): In this case a path from the source s to the destination d can be constructed as follows: $(\alpha_s, \beta_s, x_s, y_s) \rightarrow \cdots \rightarrow (p-1, \beta_d, 1, y_d) \rightarrow$ $bottom(p, \beta_d, y_d) \rightarrow \cdots \rightarrow (p, \beta_d, x_d, y_d) = (\alpha_d, \beta_d, x_d, y_d)$.

There is a link from the node $(p-1, \beta_d, 1, y_d) = top(p-1, \beta_d, y_d)$ to the node $bottom(p, \beta_d, y_d)$, because neither the node $(y_d, \beta_d, 1, p-1) = top(y_d, \beta_d, p-1)$ nor the node $top(p+1, \beta_d, y_d)$ exist, as $y_d > p$ and p is the highest indexed row in the row of blocks that contains some nodes. As the length of the path from $(\alpha_s, \beta_s, x_s, y_s)$ to the node $(p-1, \beta_d, 1, y_d)$ is at most $2n$ (Case 1.1) and the length of the path from $bottom(p, \beta_d, y_d)$ to $(\alpha_d, \beta_d, x_d, y_d)$ is less than $n/2$, the length of the path from $(\alpha_s, \beta_s, x_s, y_s)$ to $(\alpha_d, \beta_d, x_d, y_d)$ is at most $2.5n + 1$.

5 Flow Number

The metrics like the *connectivity* and the *diameter* of a graph $G = (V, E)$ are used to evaluate the relative merits of different topologies. However, these metrics are unable to capture one important aspect of a network topology. Suppose $d_{i,j}$ is the demand for data stream from node i to node j, $(1 \leq i, j \leq n)$. For example, $d_{i,j}$ may be 1Mbps, implying that 1M bits has to flow from node i to node j per second. Suppose that a demand vector $D = [d_{1,1}, \ldots, d_{n,n}]$ is specified $(d_{i,i} = 0\ \forall i, 1 \leq i \leq n)$. The traffic flow from node i to node j may be routed in many different ways. However, no matter how it is routed, such a flow may be impossible if the capacity (bandwidth) associated with the links is below a certain threshold value. Assuming that the capacity of all the links are identical, we would like to find this minimum threshold capacity of the links such that the demand D can be met through some routing. We refer to this minimum threshold capacity on the links that is able to sustain a traffic flow $d_{i,j} = 1\ \forall i, j,\ i \neq j$, as the *flow number* of the graph $G = (V, E)$.

Formally, the flow number $fn(G)$ of a graph $G = (V, E), (|V| = n, |E| = m)$ may be defined as follows: Suppose that $R_1, \ldots, R_k$ is the set of all possible ways of routing the traffic from the set of sources to the set of destinations. The traffic flow is $d_{i,j} = 1\ \forall i, j,\ i \neq j$. We refer to the traffic from a source i to a destination j as a data stream or a *flow*. Consider a $m \times k$ traffic matix T whose (i, j)-th entry, $T[i, j]$, indicates the number of data streams (or flows) through the edge (link) e_i of the graph $G = (V, E)$ under the routing scheme R_j. Then the flow number of $G = (V, E)$ is

$$fn(G) = \min_{1 \leq j \leq k} [\max_{1 \leq i \leq m} T[i, j]]$$

It may be noted that the notion of *load* introduced in [?] has some similarity with the notion of flow number being introduced in this paper. However, from the definition of load given in [?], it is clear that load is a function of (i) lightpath requests (identified by source-destination node pairs), (ii) the routing algorithm used for establishing the lightpaths and although not stated explicitly, (iii) the network topology. The flow number on the other hand is (i) independent of lightpath requests, (ii) independent of the routing algorithm used and is a function of only the topology of the network. Clearly load, which is a function of three variables (lightpath requests, routing algorithm and network topology) is different from flow number, which is a function of only one variable (network topology).

Although the notion of flow number is applicable to any networks, in the case of optical networks, it takes on additional significance. The number of wavelengths available for the establishment of lightpaths is a precious network resource and the goal of many *routing and wavelength assignment* algorithms is to optimize its use. If *wavelength conversion* is allowed, then the flow number of a network topology provides the number of wavelengths, *necessary and sufficient* to establish *full connectivity* [?] between the network nodes. If *wavelength conversion* is not allowed, then the flow number provides a *lower bound* on the number of wavelengths *necessary* to establish *full connectivity.* As a result, one topology with a low flow number is preferable over another with a high flow number.

It may be noted that the flow number of a graph can be computed using the techniques known for the *multicommodity flow* problem [?]. In the multicommodity flow problem, the capacity of the links is known and the objective is to find the maximum flow that can be attained, subject to the link capacity constraints. In flow number computation, traffic demand for all source-destination pair is known and the objective is to find the minimum link capacity that will be able to satisfy all the demands. The mathematical programming formulation to compute the flow number of a graph is given next.

For flow number computation $d_{i,j} = 1 \; \forall i, j, \;\; i \neq j$. This implies that every node has one unit of data flow to every other node of the network. In other words, there are $n(n-1)$ simultaneous flows in the network. The variable f is used to identify a flow, $1 \leq f \leq n(n-1)$. A binary indicator variable $x^f_{i,j}$ is associated with each link (i, j) of the directed graph $G = (V, E)$. If the variable $x^f_{i,j} = 1$, it indicates that the link (i, j) is a part of the path used by the flow f on its journey from the source to the destination and if $x^f_{i,j} = 0$ then it the link (i, j) is not a part of such a path.

Minimize C

Subject to the following constraints:

$$(i) \sum_{\{i|i \in V\}} x^f_{i,j} - \sum_{\{k|k \in V\}} x^f_{j,k} = \begin{cases} -1 & \text{if } j \text{ is the source node} \\ 1 & \text{if } j \text{ is the destination node} \\ 0 & \text{if } j \text{ is any other node} \end{cases}$$

$$(ii) \sum_{f=1}^{n(n-1)} x_{i,j}^f \ \le \ C, \ \forall \ (i,j) \in E$$

$$(iii) \ x_{i,j}^f \ = \ 0/1, \ \forall \ f, \ 1 \le f \le n(n-1); \ \forall (i,j) \in E$$

It is well known that the multicommodity flow problems are NP-complete [**?**]. Approximation algorithms for these problems with a guaranteed performance bound is presented in [**?**, **?**]. In the following two subsections, we show that if the graph has a regular structure, the computation of the flow number of a graph is not necessarily all that difficult. We compute the flow number of a two dimensional torus, Multi-Mesh, and a de Bruijn network and show that their flow numbers are $O(N^{1.5})$, $O(N^{1.25})$ and $O(Nlog\ N)$ respectively.

We compute the flow numbers in the following way: first we compute a lower bound of the flow number, then we compute an upper bound on the flow number and finally we show that these two are equal. Each undirected edge of both the torus and the Multi-Mesh is replaced by two oppositely directed edges. We compute the minimum capacity on these directed edges that will be able to satisfy the traffic demand of one unit for every source-destination pair ($d_{i,j} = 1 \ \forall i,j, \ i \neq j, d_{i,i} = 0 \ \forall i$).

It may be noted that the concept of *forwarding index* of communication networks studied in [**?**] has some similarity with the notion of *flow number* of communication networks being introduced in this paper. However they are not same and the results presented in this paper are completely different from the ones presented in [**?**].

Figure 9: Torus Network

5.1 Flow Number of Torus Network

Consider the $N = M \times M$ torus network shown in figure **??**. It may be observed that only the links between the nodes (1,4) and (1,5) and the nodes (1,1) and (1,8) are shown as the directed links, all other links are undirected links. This is done only to keep the diagram uncluttered. For the purpose of analysis we will consider an undirected edge as comprising of two oppositely directed edges. For simplicity of analysis, we assume that $M = 4m$ for some positive integer m.

Lemma 3 *The lower bound on flow number of a two dimensional torus network with N nodes is* $N^{1.5}/8$.

Proof: Every node in this $M \times M$ torus network has to transfer one unit of data to every other node. Consider the vertical line AB that divides the torus in two equal halves. Therefore, each of the $M^2/2$ source nodes on the left half has to transfer data to each of the $M^2/2$ destination nodes on the right half of the torus. The links *cut* by the line AB has to carry all these $M^2/2 * M^2/2 = M^4/4$ traffic from the left half of the torus to the right half. There are exactly $2M$ directed edges (links) connecting the nodes on the left half to the nodes on the right half. (Each of the M rows of the torus has two such links; in figure **??**, on the first row these links are from the node (1,1) to (1,8) and the node (1,4) to (1,5)). Since $2M$ links has to carry $M^4/4$ flows, the maximum of the minimum flows on these $2M$ links must be $(M^4/4)/2M = M^3/8 = N^{1.5}/8$. This is a *lower bound* on the flow number of a $N = M \times M$ torus.

Lemma 4 *The upper bound on flow number of a two dimensional torus network with N nodes is* $N^{1.5}/8$.

Proof: The upper bound is established by first describing a routing algorithm. We show that if this algorithm is executed, no directed edge (link) will be required to carry more than a certain number of data streams or flows. This establishes an upper bound on the flow number.

As indicated earlier, we assume that $M = 4m$ for some positive integer m. Assume that the torus is divided into two halves by the line AB as shown in figure **??**. There are $M^2/2 = 8m^2$ nodes on each of these two halves. The $8m^2$ nodes are distributed in $4m$ rows, with each row having $2m$ nodes. Consider another cut line CD shown in figure **??** that divides the left half into two parts. The $2m$ nodes on the first row in the left half is divided into two equal parts of m modes each by this line. These $2m$ nodes (the nodes $(1,1), \ldots, (1,4)$ in figure **??**) have to transfer one data stream to the nodes on the right side (the nodes $(1,5), \ldots, (1,8)$ in figure **??**). The links $(1,1) \rightarrow (1,8)$ and $(1,4) \rightarrow (1,5)$ can be used for this purpose. Therefore, the traffic on any one of these two links will be $m * 2m = 2m^2$. The routing algorithm will route the traffic in such a way that the traffic from the nodes $(i, m+1), \ldots, (i, 2m), 1 \leq i \leq 4m$ will use the link $(j, 2m) \rightarrow (j, 2m+1)$, to travel to the nodes $(j, 2m+1), \ldots, (j, 4m), 1 \leq j \leq 4m$. For example, the path taken by the traffic to reach the destination node $(j, 4m)$ from the source node $(i, m+1)$ will be as follows: $(i, m+1) \rightarrow (i+1, m+1) \rightarrow \cdots \rightarrow (j, m+1) \rightarrow (j, m+2) \rightarrow \cdots \rightarrow (j, 2m) \rightarrow (j, 2m+1) \rightarrow (j, 2m+2) \rightarrow \cdots \rightarrow (j, 4m)$. The traffic from the nodes $(i, 1), \ldots, (i, m), 1 \leq i \leq 4m$ will use the link $(j, 1) \rightarrow (j, 4m), 1 \leq i, j \leq 4m$ to travel to the nodes $(j, 2m+1), \ldots (j, 4m)$. Therefore the total traffic on the links, $(j, 1) \rightarrow (j, 4m)$ and $(j, 2m) \rightarrow (j, 2m+1)$ for all j, $1 \leq j \leq 4m$ will be $2m^2 * 4m = 8m^3 = M^3/8 = N^{1.5}/8$. Since, the torus has a completely symmetric structure, no directed edge will carry a traffic that is higher than this value. Thus $N^{1.5}/8$ is the upper bound on the flow number of the torus network.

Theorem 5 *The flow number of a two dimensional torus network with N nodes is* $N^{1.5}/8$.

Proof: Follows from lemmas 3 and 4.

5.2 Flow Number of Multi-Mesh Network

Consider the $N = M \times M$ node Multi-Mesh network shown in figure **??**. For the purpose of analysis we will consider an undirected edge as comprising of two oppositely directed edges.

Lemma 6 *The lower bound on flow number of a Multi-Mesh network is* $N^{1.25}/2$.

Proof: As shown in figure 1, the nodes in the $M \times M$ Multi-Mesh are arranged in blocks of smaller meshes of size $n \times n$, where $n = \sqrt{M}$. The total number of nodes in the network $N = M^2 = n^4$ is distributed in n^2 blocks with each block having n^2 nodes. Every node in this $M \times M$ Multi-Mesh network has to transfer one unit of data to every other node. Consider the vertical line AB that divides the Multi-Mesh in two equal halves. Therefore, each of the $M^2/2 = n^4/2$ source nodes on the left half has to transfer data to each of the $M^2/2 = n^4/2$ destination nodes on the right half of the Multi-Mesh. Since each block has exactly n^2 nodes, the number of blocks on both sides of the cut line AB is $n^2/2$. The number of rows of blocks on each sides is n. Therefore the number of blocks in each row of blocks on each side of the line AB is $n/2$. Each block on the i-th row of blocks on the left half is connected to each block on the i-th row of blocks on the right half by two links, $(1 \leq i \leq n)$. Thus the total number of links that can carry traffic from the i-th row blocks on the left half to the i-th row blocks on the right half is $n/2 * n/2 * 2 = n^2/2$. The total number of rows of block is n. Thus the total number of links that can carry traffic from the left half of the network to the right half is $n * n^2/2 = n^3/2$.

These links *cut* by the line AB has to carry all these $M^2/2 * M^2/2 = M^4/4 = n^8/4$ traffic from the left half of the Multi-Mesh to the right half.

Since $n^3/2$ links has to carry $n^8/4$ flows, the maximum of the minimum flows on these $n^3/2$ links must be $(n^8/4)/(n^3/2) = n^5/2 = M^{2.5}/2 = N^{1.25}/2$. This is a *lower bound* on the flow number of a $M \times M$ Multi-Mesh.

Lemma 7 *The upper bound on flow number of a Multi-Mesh network is* $N^{1.25}/2$.

Proof: The upper bound is established by first describing a routing algorithm. We show that if this algorithm is executed, no directed edge (link) will be required to carry more than a certain number of data streams or flows. This establishes an upper bound on the flow number.

As indicated earlier, the nodes in the $M \times M$ Multi-Mesh are arranged in n^2 blocks of n^2 nodes each, where $n = \sqrt{M}$. The blocks are arranged in n rows and n columns. Each node in the Multi-Mesh interconnection is identified with a four tuple label (i, j, k, l).

The first two (i, j) identifies the block and the last two (k, l) identifies the node within a block. Due to the construction rules of the Multi-Mesh, any two blocks that belong to the same row or same column form a *cycle*, in the sense that these two blocks are connected by two links. That is two blocks (i, j) and (i, k) have the following two links connecting them: $(i, j, k, 1)$ to (i, k, j, n) and (i, j, k, n) to $(i, k, j, 1)$.

Every node in this $M \times M$ Multi-Mesh network has to transfer one unit of data to every other node. Suppose the following routing technique is used: Consider a flow from the node $(i1, j1, k1, l1)$ to $(i2, j2, k2, l2)$. The flow first has to travel to the block identified by $(i2, j2)$ from the block $(i1, j1)$. The routing technique, tries to match *the row first and then the column.* That is to travel from $(i1, j1)$ to $(i2, j2)$, first it travels to the intermediate block $(i2, j1)$ and then to the block $(i2, j2)$. Within a block also the routing algorithm follows the same *row first* policy. Suppose a flow has to travel from the node $(i1, j1, k1, l1)$ to $(i2, j2, k2, l2)$. According to the *row first* policy, the flow should travel from the $(i1, j1)$ block to the $(i2, j1)$ block first. The two nodes on the horizontal (vertical) boundary of a block responsible for communicating with other blocks in the network are referred to as the *communicating nodes.* Two communicating nodes on the horizontal boundary of the $(i1, j1)$ block is connected to two corresponding nodes on the horizontal boundary of the $(i2, j1)$ block. According the routing technique considered here, the flow originating from the node $(k1, l1)$ within the $(i1, j1)$ block traverses to the *nearest* communicating node to leave the $(i1, j1)$ block to enter the $(i2, j1)$ block.

The traffic that flows through the network can be divided into two classes: *inter-block traffic* and *intra-block traffic.* We first analyze the inter-block traffic and later we analyze the intra-block traffic.

Inter-block Traffic Analysis

As indicated earlier, the n^2 blocks of a Multi-Mesh network are arranged in n rows and n columns. Suppose the blocks are numbered from (1, 1) through (n, n). Consider the (i, j)-th block with n^2 nodes. The inter-block traffic that enters (or leaves) the (i, j)-th block can be divided into two classes: (A) The traffic that either originates or terminates at the (i, j)-th block, (B) The transit traffic that uses the (i, j)-th block as an intermediate block to get to the destination.

Class A: Originating or Terminating Traffic at the (i, j)-th block:

The (i, j)-th block of the Multi-Mesh network has n^2 nodes. The remaining $n^4 - n^2$ nodes of the network have data streams for each node of the (i, j)-th block. Similarly, each node of the (i, j)-th block have data streams for each of the remaining $n^4 - n^2$ nodes of the network. Thus the total traffic originating from other blocks and terminating in the (i, j)-th block is $(n^4 - n^2)n^2$. The same number of data streams originate at the (i, j)-th block and terminate at the other blocks. Like any other block of the network, the (i, j)-th block has $4(n - 1)$ incoming (outgoing) links. Traffic from the blocks $(s, t), 1 \leq s, t \leq n, t \neq j$ will enter the block (i, j) through the nodes $(i, j, t, 1)$ and the node (i, j, t, n). Thus the number of data streams on these two incoming links will be $(n^2 * n^2 * n * (n-1))/2(n-1) = n^5/2 = M^{2.5}/2$. The outgoing data streams on these links will $(n^2 * n^2 * (n-1))/2(n-1) = n^4/2 = M^2/2$.

Traffic from the blocks $(s, j), 1 \leq s \leq n, s \neq i$ will enter the block (i, j) through the nodes $(i, j, 1, s)$ and the node (i, j, n, s). Similarly, the traffic to the blocks $(s, j), 1 \leq s \leq n, s \neq i$ will leave the (i, j)-block through the nodes $(i, j, 1, s)$ and the node (i, j, n, s). Thus the incoming traffic on these links will be $(n^2 * n^2 * (n - 1))/2(n - 1) = n^4/2$. The outgoing traffic on these links will be $(n^2 * n^2 * n * (n - 1))/2(n - 1) = n^5/2$.

Class B: Transit Traffic at the (i, j)-th block

The traffic from the blocks $(s, j), 1 \leq s \leq n, s \neq i$ to the blocks $(i, t), 1 \leq t \leq n, t \neq j$ will enter the block (i, j) through the nodes $(i, j, 1, s)$ and (i, j, n, s) and will leave the block (i, j) through the nodes $(i, j, t, 1)$ and (i, j, t, n). Thus the total transit traffic on these incoming and outgoing links will be $(n^2 * n^2 * (n-1)^2)/2(n-1) = n^5/2 - n^4/2$.

Thus the total incoming (outgoing) traffic at the nodes $(i, j, 1, s)$ and (i, j, n, s), ($(i, j, t, 1)$ and (i, j, t, n)) will be $n^4/2 + n^5/2 - n^4/2 = n^5/2 = M^{2.5}/2 = N^{1.25}/2$.

Thus it can be seen that the number of data streams on inter-block links never exceeds the lower bound value of $N^{1.25}/2$.

Intra-block Traffic Analysis

In this part we consider the traffic carried one directed edge (link) within a block. We consider the link $(i, j, k, l) \rightarrow (i, j, k, l+1),\ 1 \leq i, j, k, l+1 \leq n$ as shown in figure **??**. The traffic carried by this link can be divided into four different categories:

(i) *Incoming Traffic:* Traffic originating in block $(s, t),\ 1 \leq s, t \leq n,\ s \neq i,\ t \neq j$ and terminating in block (i, j).

(ii) *Outgoing Traffic:* Traffic originating in block (i, j) and terminating in block $(s, t),\ 1 \leq s, t \leq n,\ s \neq i,\ t \neq j$.

(iii) *Transit Traffic:* Traffic neither originating nor terminating in block (i, j) but traversing through the block (i, j) as an intermediate block.

(iv) *Internal Traffic:* Traffic originating and terminating in block (i, j).

Case (i): Incoming Traffic on the link $(i, j, k, l) \rightarrow (i, j, k, l+1),\ 1 \leq i, j, k, l+1 \leq n$

The link $(i, j, k, l) \rightarrow (i, j, k, l+1)$ is shown in figure **??**. The incoming traffic enters the block (i, j) through the communicating nodes on either the horizontal or the vertical boundary. The nodes $(i, j, p, 1)$ and $(i, j, p, n),\ 1 \leq p \leq n,\ p \neq j$ will have incoming traffic from the blocks $(r, p),\ 1 \leq p, r \leq n,\ p \neq j$. Half of the traffic from the block

Figure 10: Nodes in one block of a Multi-Mesh Network

(r,p) will enter the block (i,j) through the node $(i,j,p,1)$ and the other half through the node (i,j,p,n). It may be noted that the traffic from the block (r,p) first enters the block (i,p) through the nodes $(i,p,1,k)$ and (i,p,n,k), $1 \leq k \leq n, k \neq i$ and then leaves the block (i,p) through the nodes $(i,p,j,1)$ and (i,p,j,n). The routing algorithm ensures that half of the traffic from the block (r,p) enters the block (i,p) through the node $(i,p,1,k)$ and leaves through the node $(i,p,j,1)$ and the other half enters through the node (i,p,n,k) and leaves through the node (i,p,j,n). The total traffic (flow) from the block (r,p) to the block (i,j) is $n^2 * n^2 = n^4$. Half of this traffic, i.e., $n^4/2$ will enter the block (i,j) through the node $(i,j,p,1)$. This traffic will be divided into n equal parts to reach the destinations on n different rows of the (i,j)-th block. Thus the part of this traffic on the k-th row is $n^3/2$. Each node on this row consumes n^2 amount of the flow before it reaches the node (i,j,k,l). Therefore the part of this traffic on the link $(i,j,k,l) \rightarrow (i,j,k,l+1)$ is at most $n^3/2 - ln^2$. Thus the total traffic from the block

(r,p) on the link $(i,j,k,l) \rightarrow (i,j,k,l+1)$ is $(n^3/2 - ln^2) * (n^2 - n)$, for all possible values of r and p, $(1 \leq p, r \leq n,\ p \neq j)$.

The nodes $(i,j,1,p)$ and (i,j,n,p), $1 \leq p \leq n,\ p \neq i$ will have incoming traffic from the blocks $(r,j),\ 1 \leq r \leq n,\ r \neq i$. Half of the traffic from the block (r,j) will enter the block (i,j) through the node $(i,j,1,p)$ and the other half through the node (i,j,n,p). The total traffic (flow) from the block (r,j) to the block (i,j) is $n^2 * n^2 = n^4$. Half of this traffic, i.e., $n^4/2$ will enter the block (i,j) through the node $(i,j,1,p)$. This traffic will be divided into n equal parts to reach the destinations on n different rows of the (i,j)-th block. Thus the part of this traffic on the k-th row is n^3. Each node on this row consumes n^2 amount of the flow before it reaches the node (i,j,k,l). Therefore the part of this traffic on the link $(i,j,k,l) \rightarrow (i,j,k,l+1)$ is $n^3 - ln^2$. The link $(i,j,k,l) \rightarrow (i,j,k,l+1)$ will carry traffic from the block (r,j) as long as $1 \leq r \leq l$. Thus the total traffic from the block (r,j) on the link $(i,j,k,l) \rightarrow (i,j,k,l+1)$ is $l(n^3 - ln^2)$, for all possible values of r $(1 \leq r \leq l)$.

Thus the total incoming traffic on the link $(i,j,k,l) \rightarrow (i,j,k,l+1)$ is $(n^3/2 - ln^2) * (n^2 - n) + l(n^3 - ln^2) = n^5/2 - (l+1/2)n^4 + 2ln^3 - l^2n^2$.

Case (ii): Outgoing Traffic on the link $(i,j,k,l) \rightarrow (i,j,k,l+1),\ 1 \leq i,j,k,l \leq n$

In the following, we present the analysis for the case $l \leq n/2$. The analysis is similar for the case when $l > n/2$ and is not presented here. The outgoing traffic on the link $(i,j,k,l) \rightarrow (i,j,k,l+1),\ 1 \leq i,j,k,l+1 \leq n$ when $l \leq n/2$ is zero unless $k = 1$ or $k = n$. We consider the case when $k = n$. In this case only the nodes $(i,j,x,y),\ n/2 < x \leq n,\ 1 \leq y \leq l$ will be using the link $(i,j,k,l) \rightarrow (i,j,k,l+1)$ to transmit data to the $(l+1,z)$-th $(1 \leq z \leq n)$ block in the network. Thus the total traffic on the $(i,j,k,l) \rightarrow (i,j,k,l+1)$ will be at most $n/2 * l * n * n^2 = ln^4/2$.

Case (iii): Transit Traffic on the link $(i, j, k, l) \rightarrow (i, j, k, l+1),\ 1 \leq i, j, k, l+1 \leq n$ As in the previous case we consider the case $l \leq n/2$. When $l \leq n/2$, then the transit traffic through the link $(i, j, k, l) \rightarrow (i, j, k, l+1)$ is zero, beacuse the routing algorithm uses the *row first* policy. If $l > n/2$, then the traffic is non-zero and the expression for the traffic the reader is referred to [**?**].

Case (iv): Internal Traffic on the link $(i, j, k, l) \rightarrow (i, j, k, l+1),\ 1 \leq i, j, k, l+1 \leq n$

The internal traffic on the link $(i, j, k, l) \rightarrow (i, j, k, l+1),\ 1 \leq i, j, k, l+1 \leq n$ is $nl(n-l)$.

Thus the total traffic on the link $(i, j, k, l) \rightarrow (i, j, k, l+1),\ 1 \leq i, j, k, l+1 \leq n$ is

$$n^5/2 - (l+1/2)n^4 + 2ln^3 - l^2n^2 + ln^4/2 + 0 + ln^2 - l^2n$$

which after simplification is

$$n^5/2 - (l+1)n^4/2 + 2ln^3 - (l^2 - l)n - l^2n$$

The above expression is less than equal to $n^5/2 = M^{2.5} = N^{1.25}/2$ if $n \geq 4$. This establishes an upper bound on the intra-block traffic.

Theorem 8 *The flow number of a Multi-Mesh network with N nodes is $N^{1.25}/2$.*

Proof: Follows from lemmas 3 and 4.

5.3 Flow Number of de Bruijn Network

Lightwave networks based on de Bruijn graphs was proposed in [**?**]. For the sake of completeness, we describe the network first. An N node de Bruijn network can only be constructed if $N = \Delta^D$, for two integers Δ and D. Each node v has a D digit representation $(v_1, v_2, \ldots, v_D)$, where $v_i \in \{0, 1, \ldots, \Delta - 1\}, 1 \leq i \leq D$. There is a directed edge from the node u, $(u_1, u_2, \ldots, u_D)$ to the node v, $(v_1, v_2, \ldots, v_D)$ if $v_i = u_{i+1}$ for $1 \leq i \leq D-1$. Each node of de Bruijn network has Δ incoming and outgoing edges. The *diameter* and *connectivity* of a de Bruijn network with Δ^D nodes are D and $\Delta - 1$ respectively.

Edge loading L_i for the edge e_i was defined in [?] as the number of lightpaths passing through edge e_i. Assuming that a lightpath needs to be established between every pair of nodes in the network, the authors in [?] computed *average* and *maximum* edge loading in the graph. Clearly the edge loading will be dependent of the routing algorithm used for message transfer between the nodes. Two routing algorithms - *shortest path routing* and *longest path routing* was considered in [?] and the maximum loading values for the two algorithms presented for de Bruijn networks with different values of Δ and D.

In terms of edge loading considered in [?], the notion of flow number presented in this paper is the *optimal edge loading*. That is, the flow number is the maximum edge loading of the best possible routing algorithm. The maximum edge loading using even the better of the two algorithms (longest path routing) considered in [?] is not optimal. Using the mathematical programming formulation of the flow number in section 5, we computed the flow number of de Bruijn graphs using the CPLEX optimization package on a SUN Ultra 5 workstation. The results of edge loading with the longest path routing $L_{max}(LP)$ and the shortest path routing $L_{max}(SP)$ and the flow number of de Bruijn network with different values for Δ and D is presented in table ??. Due to the memory limitation of our workstation, we were unable to compute the flow number of de Bruijn graphs of larger size. The mathematical programming formulation of flow number for de Bruijn graph with $\Delta = 4$ and $D = 2$, has 144000 variables and 3900 constraints. The workstation took less than two minutes to compute the result.

Table 1: Comparison Table of $L_{max}(SP)$, $L_{max}(LP)$ and Flow Number

Δ	D	$L_{max}(SP)$	$L_{max}(LP)$	Flow Number
2	2	3	4	3
2	3	11	12	9
2	4	29	32	26
2	5	81	80	66
3	2	7	6	6
3	3	31	27	25
4	2	9	8	8

Next we present the flow number of a de Bruijn graph.

Lemma 9 *The lower bound on flow number of a de Bruijn network with N nodes is $\Omega(Nlog\ N)$.*

Proof: The *bisection width* of a network is the minimum number of links that need to be removed in order to disconnect the network into two halves with identical (within one) number of nodes [?]. It has been shown in [?], that the bisection width of a de Bruijn network is $\Theta(N/log\ N)$, where N is the number of nodes in the graph. If the network is divided into two equal size groups of $N/2$ nodes each, then the total traffic from the nodes in one group to the nodes on the other group will be $N/2 \times N/2$. Since the bisection width is $\Theta(N/log\ N)$, the flow number is $\Omega((N^2/4)/(N/log\ N)) = \Omega(Nlog\ N)$.

Lemma 10 *The upper bound on flow number of a de Bruijn network with N nodes is $O(Nlog\ N)$.*

Proof: In [?] it was shown that if *longest path routing* is used then the maximum loading on an edge, $L_{max}(LP)$, is given by

$$L_{max}(LP) \leq D\Delta^{D-1} = (Nlog_{\Delta}\ N)/\Delta$$

Clearly this is an upper bound of the flow number of a de Bruijn graph.

Theorem 11 *The flow number of a de Bruijn network with N nodes is $\Theta(Nlog\ N)$.*

Proof: Follows from the previous lemmas 6 and 7.

In table **??** we present a comparison between various parameters of torus, Multi-Mesh and de Bruijn networks.

Table 2: Comaprison Table of Various Network Attributes

Network Type	Node Degree	Connectivity	Diameter	Flow Number
Torus	4	4	$\Theta(N^{0.5})$	$\Theta(N^{1.5})$
Multi-Mesh	4	4	$\Theta(N^{0.25})$	$\Theta(N^{1.25})$
De Bruijn	$\Delta, \Delta - 1$	$\Delta - 2$	$\Theta(log\ N)$	$\Theta(Nlog\ N)$

6 Simulation Results

In this section, we study the routing and wavelength assignment (RWA) problem in a wavelength routed single hop network with a physical topology based on the Multi-Mesh architecture through simulation. The simulation model used for this study is identical to the one used in [**?**]. The features of the study are:

- The network topology under evaluation is Generalized Multi-Mesh

a) The physical topology of the network is GM,

b) the maximum number of duplex connections is m, i.e., each node of the network has a specified number, m, of optical transmitters and receivers,

c) the maximum blocking probabilty is given,

d) the problem is to find the number of wavelengths necessary to ensure that the call blocking probability p will be below an acceptably low threshold value.

As in the case of the de Bruijn graph and the Kautz graph studied in [?], [?] the analytical solution of the RWA problem for the Multi-Mesh network appears difficult [?]. Following [?], [?], we use Monte-Carlo simulation to determine the number of wavelengths necessary to ensure a small call blocking probability. We assume that the wavelength continuity constraint is satisfied so that a light path traversing a number of fibers always uses the same wavelength on every fiber on the path. We carried out the experiment for three different values of m, 5, 10 and 15, (m is the number of transmitters and receivers in each node). In our Monte-Carlo simulation of the Multi-Mesh network we varied the number of wavelengths N used in the network to observe the variation of the call blocking probability p.

The following assumptions are made in our study:

a) The traffic between all source-destination pairs are equally likely.

b) Each source node has the capability of tuning its transmitter to any one of the N wavelengths used in the network.

The following steps were taken to observe the variation of the call blocking probability p with the number of wavelengths used in the network λ, when each node of the network has exactly m transmitters and receivers.

1) The simulation is initialized by setting the number of blocked calls to 0 and the total number of attempted connections to 0.

2) The steps 3-8 were repeated one hundred times.

3) The network is initialized by ensuring that no edge is carrying any lightpath and all m transmitters and receivers of each node in the network are available for use.

4) While at least one node S has a transmitter free and one node D has a receiver free ($S \neq D$), repeat steps 5 - 8.

5) Select at random, a pair (S, D) of nodes such that

a) the node S has at least one of its m transmitters free and

b) the node D has at least one of its m receivers free and

c) S is distinct from D and

d) either the node pair (S, D) was never selected earlier or the last attempt for a connection from S to D was successful.

6) Examine the shortest path P from S to D (or one of the shortest paths if there were more than one) to find out if there exists a wavelength $\lambda_i, 0 \leq i < N$ such that none of the edges in the path P currently has a lightpath with a wavelength λ_i.

7) If no such wavelength is available, we conclude that the call is blocked. In this case we

a) increase the number of blocked calls by 1,

b) insert the pair (S, D) in a list of source and destination pairs of nodes that were blocked in previous attempts and

c) return to step 4.

8) If a wavelength λ_i is found, we

a) allocate the wavelength λ_i on each fiber of the path P for the communication from S to D,

b) reduce the number of transmitters of the source node S and the number of receivers of the destination node D by 1 and

c) return to step 4.

Shortest Path Algorithm for Multimedia Data Transmission Environment

Input: The directed graph $G = (V, E)$, $(V = \{1, 2, \ldots, n\})$, two $n \times n$ matrices δ and s, the (i, j)-th entry of the matices stores the delay factor δ and the size factor s of the link from the node i to node j. If there is no link from the node i to j, both $\delta_{i,j}$ and $s_{i,j}$ is taken to be ∞. Without any loss of generality, we assume that the node 1 is the source node and node n is the destination node.

Output: Array $D(1, \ldots, n)$, that stores the shortest path length from node i to the destination node n for all i, $1 \leq i \leq n$.

Comments: The algorithm starts from the destination node and in each iteration finds the shortest path from a node i in the graph to the destination node n, $1 \leq i \leq n-1$.

begin

  $C := \{1, 2, \ldots, n-1\}$;

  for $i := n-1$ **downto** 1 **do**

    $D[i] := \delta[i, n]$

  repeat $n-2$ **times**

    begin

      $v := i$ such that $D[i]$ has the minimum value, $1 \leq i \leq n-1$;

      $C := C \setminus \{v\}$;

      for each $w \in C$ **do**

        $D[w] := min(D[w], \delta[w, v] + s[w, v]D[v])$;

    end

end

At the completion of the simulation, we calculated the relative frequencies of the blocked connections to obtain an estimate of the blocking probability when the network has λ available wavelengths for connection establishment. This simulation had to be repeated many times with increasing values of λ so that the call blocking probability p

Figure 11: Call blocking probability versus number of wavelengths in 81 node Multimesh

falls below an acceptablly low threshold value.

In figures **??**, **??**, **??** we plotted the blocking probability versus the number of wavelengths using a logarithmic scale. In the graphs, the squares, diamonds and crosses represent the cases with $m = 5$, 10 and 15 respectively. These graph may be used by a network designer to determine an appropriate value of λ for a network with a specified value of m and acceptable limit of the call blocking probability p. The graphs show a steep decline in the values of p as λ is increased. This implies that it is possible reduce the call blocking probability to an arbitrarily small value by a modest increase in the number of wavelengths.

Figure 12: Call blocking probability versus number of wavelengths in 256 node Multimesh

Figure 13: Call blocking probability versus number of wavelengths in 625 node Multimesh

7 Conclusion

We have adapted the multimesh topology proposed for multiprocessor interconnection wavelength division multiplexed optical networks. In order to do so we have extended the multimesh topology so that the new topology allows any number of nodes without sacrificing the attractive properties of the MM. Even though researchers have proposed complex topologies (e.g., de Bruijn, Hypercube) for optical networks, existing networks are based on simple structures such as the Ring. The MM is attractive for optical networks because it is simple and resembles the familiar Torus network but has superior topological properties. This paper shows that with some simple modifications, it is possible to use the novel interconnection pattern of MM in the optical network domain. In addition, we also have introduced a new metric for the evaluation of interconnection topologies for lightwave networks. Using previously known metrics, such as the connectivity and the diameter, as well as the newly introduced metric, the flow number, we have shown that the new architecture has many attractive features to be considered for use as an alternate to the other topologies proposed for use in the optical network domain.

Acknowledgement: The research of S. Banyopadhyay was supported by a research grant from the National Sciences and Engineering Research Council of Canada.

References

[1] A.S. Acampora,"A multichannel multihop local lightwave network", *Proc. of IEEE Globecom*, pp. 1459-1467, Nov. 1987.

[2] S. Chatterjee and S. Pawlowski,"All optical networks", *Communications of the ACM*, vol. 42, no. 6, pp. 75-83, June 1999.

[3] F. R. K. Chung, E. G. Coffman, M. I. Reiman and B. Simon, "The forwarding index of communication networks", *IEEE Transactions on Information Theory*, vol. IT 33, no. 2, pp. 224-231, 1987.

[4] D. Das and B. P. Sinha, “Multi-Mesh - An efficient topology for parallel processing”, *Proc. of 9th Int. Parallel Processing Symp.*, Santa Barbara, California, pp. 17-21, April 1995.

[5] M. De, B. Kundu and B. P. Sinha, “Wormhole routing for complete exchange in Multi-Mesh”, *Proc. of 4th Int. Conf. on High Performance Computing*, Bangalore, India, December 1997.

[6] M. De, D. Das, M. Ghosh and B. P. Sinha, “An efficient sorting algorithm on the Multi-Mesh network”, *IEEE Transactions on Computers*, vol. C 46, no. 10, pp. 1132-1137, 1997.

[7] D. Das, M. De and B. P. Sinha, “A new network topology with multiple meshes”, *IEEE Transactions on Computers*, vol. C 48, no. 5, pp. 536-551, 1999.

[8] P.K. Dowd, “Wavelength division multiple access channel hypercube processor interconnection”, *IEEE Transactions on Computers*, 1992.

[9] N. Ghani and S. Dixit, “Channel provisioning for higher layer protocols in WDM networks”, *Proc. of SPIE All Optical Networking Conf: Architecture, Control and Management Issues*, Boston, MA, Sept. 1999.

[10] N. Ghani, S. Dixit, T.S. Wang “On IP over WDM Integration”, *IEEE Communications Magazine*, vol. 38, no. 3, pp. 72-84, March 2000.

[11] F. T. Leighton, Introduction to Parallel Algorithms and Architectures, Morgan Kaufman.

[12] M.R. Garey and D.S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness. W.H. Freeman Publishers, San Francisco, 1978.

[13] M.G. Hluchyj and M.J. Karol, “Shufflenet: An application of generalized perfect shuffles to multihop lightwave networks”, *Journal of Lightwave Technology*, vol. 9, no. 10, October, 1999.

[14] T. C. Hu, Integer Programming and Network Flows, Addison Wesley Publishing Company, 1970.

[15] P. Klein, S. Plotkin, C. Stein, E. Tardos, “Faster approximation algorithms for the unit capacity concurrent flow problem with applications to routing and finding sparse cuts”, *SIAM Journal of Computing*, vol. 23, no. 3, pp. 466-487, 1994.

[16] M. A. Marsan, A. Bianco,E. Leonardi and F. Neri, “Topologies for Wavelength-Routing All-Optical Networks”, *IEEE/ACM Transactions on Networking*, vol. 1, pp. 534-546, 1993.

[17] M. F. Maxemchuk, “Regular mesh topologies in local and metropolitan area networks”, *AT & T Technical Journal*, vol. 64, pp. 1659-1686, September, 1985.

[18] M. F. Maxemchuk, “Routing in Manhattan Street networks”, *IEEE Transactions on Communications*, vol. 35, pp. 503-512, May 1987.

[19] B. Mukherjee, Optical Communication Networks, McGraw-Hill, 1997.

[20] G. Panchapakesan and A. Sengupta, "On multihop optical network topology using Kautz digraph," *Proc. IEEE INFOCOM'95*, pp. 675-682, April, 1995.

[21] R. Ramaswami and K. Sivarajan, "Design of logical topologies for wavelength routed all optical network," *Proc. IEEE INFOCOM'95*, pp. 1316-1325, 1995.

[22] R. Ramaswami and K. Sivarajan, "Routing and wavelength assignment in all-optical networks," *IEEE/ACM Transactions on Networking*, vol. 3, no. 5, pp. 489-500, 1995.

[23] R. Ramaswami and K. Sivarajan, Optical Networks: A Practical Perspective, Morgan Kaufmann Publishers Inc, 1998.

[24] A. Sen, T. Shah, S. Bandyopadhyay and B.P. Sinha, "On new architectures for WDM networks", *Proc. of 37th Annual Allerton Conference on Communication, Control and Computing*, pp. 402-413, University of Illinois at Urbana-Champaign, September 1999. Extended version available as *Technical Report*, Department of Computer Science and Engineering, Arizona State University, 1999.

[25] F. Shahrokhi and D.W. Matula, "The maximum concurrent flow problem", *Journal of the Association of Computing Machinery*, vol. 37, no. 2, pp. 318-334, 1990.

[26] A. Sengupta, S. Bandyopadhyay, A. R. Balla and A. Jaekel, "Algorithms for dynamic routing in all-optical networks", vol. 2, no. 2, pp. 163-184, 2000.

[27] K. Sivarajan and R. Ramaswami, "Lightwave networks based on de Bruijn graphs," *IEEE/ACM Transactions on Networking*, vol. 2, no. 1, pp. 70-79, 1994.

[28] T.E. Stearn and K. Bala, Multiwavelength Optical Networks: A Layered Approach, Addison Wesley Longman Inc, 1999.

[29] K.W. Tang, "CayleyNet: A multihop WDM-based lightwave network," *Proc. IEEE INFOCOM'94*, June, 1994.

[30] T. D. Todd and E. L. Hahne,"Local and metropolitan Multi-Mesh networks,"*Proc. of IEEE International Communication Conference (ICC)*, pp. 900-904, 1992.