Three-Dimensional Network Topologies †

John Nguyen[1], John Pezaris[2], Gill Pratt[2], and Steve Ward[2]

[1] University of Michigan
[2] Massachusetts Institute of Technology

johnn@eecs.umich.edu
pz, gill, ward@lcs.mit.edu

Abstract. This paper presents the derivation and performance results of several new three-dimensional topologies. Various transformations can be applied to the conventional six-neighbor mesh in order to construct these topologies, which vary both in number of neighbors (degree) and logical connectivity. Analysis shows that after normalization for constant pin-count, lower-degree topologies yield lower latencies for long messages on unloaded networks, while higher-degree topologies possess higher bandwidth capacities. Although simulation results generally verify these findings, we also observe a surprising amount of difference in the performance between distinct topologies of the same degree.

1 Introduction

The past few years have seen a rise in popularity of multiprocessors using direct networks that span two or three dimensions. Such networks typically follow the topology of a two or three-dimensional mesh or torus. Although topologies other than the mesh have been studied for two-dimensional space[7], there have been few investigations of alternate topologies in three-dimensional space. This paper proposes five such alternate topologies and presents some analytical and empirical performance results.

We restrict our study to direct topologies whose nodes all possess the same number of neighbors, or *degree*. Furthermore, the node degree for any topology remains constant no matter how many nodes are in the network. Thus non-constant degree topologies such as hypercubes are not considered. Also eliminated are indirect topologies such as butterflies and fat-trees.

Since high-degree topologies require a larger number of channels on each switch, we must somehow normalize performance to the hardware complexity required by the degree of the topology. This can be accomplished by requiring a constant switch complexity through reducing the channel width of higher-degree topologies. On such topologies, the narrower data path can in turn degrade performance by increasing the number of flits required by long messages. Conversely, a network requiring a small number of channels per node allows one to increase

† This research is sponsored by DARPA contract #DABT63-93-C-0008.

the data path size without increasing switch complexity and thus possibly decrease the latency of long messages. Many of the topologies that are presented here can be viewed as an attempt to reduce the degree of each node.

In the following sections, we present several new three-dimensional topologies as well as a formal definition of a topology, derive analytical results to predict performance, and discuss results of some routing simulations.

2 Topologies

We present in this section five topologies can be derived from various modifications to the conventional six-neighbor mesh. These modifications include splitting each six-neighbor node into several nodes as well as adding and removing links from the six-neighbor topology. Although physical representations are presented for clarity, the topologies are really defined by the logical connectivity of the nodes. The following section will focus on a more formal treatment of topologies.

Topology A: In the standard three-dimensional mesh, each node is represented as a point (n_1, n_2, n_3) in the cartesian three-dimensional space where each n_i is an integer. The six neighbors of a node are defined as nodes with points corresponding to $+1$ and -1 offsets in one of the three axes. Such a network is illustrated in Figure 1.

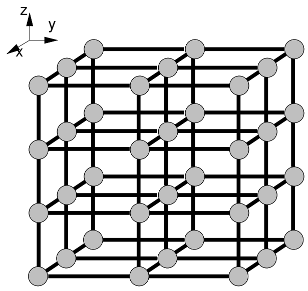


Figure 1: 6-neighbor cartesian mesh (Topology A)

Topology B: A second topology can be formed by inserting a node at each point $(n_1+\frac{1}{2}, n_2+\frac{1}{2}, n_3+\frac{1}{2})$ to the set of integral-coordinate nodes. The neighbors of a node (x, y, z) can then be defined as the eight nodes represented by coordinates $(x \pm \frac{1}{2}, y \pm \frac{1}{2}, z \pm \frac{1}{2})$, as shown in Figure 2. This topology can also be formed by adding diagonal links in the directions $(1, 1, 1)$ and $(-1, -1, -1)$ to Topology A. The equivalence of the two modifications will be discussed in the next section.

The remaining four topologies can be viewed as modifications that are derived by splitting each node of the six-neighbor mesh of Topology A into several *subnodes*. In particular, four-neighbor topologies can be formed by splitting each

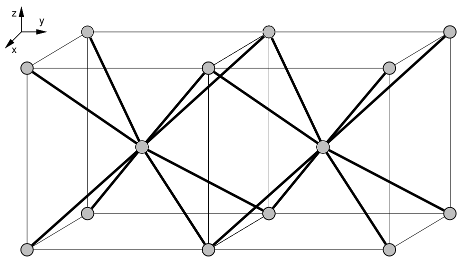


Figure 2: 8-neighbor topology (Topology B)

six-neighbor node into two subnodes, each with three external neighbors and one internal neighbor. Likewise, three-neighbor topologies can be formed by dividing each six-neighbor node into six subnodes, each with one external neighbor and two internal neighbors. For the following derivations, we label the six ports of a node in Topology A as $-x$, $+x$, $-y$, $+y$, $-z$, and $+z$. The pair $-x$ and $+x$ are called *opposing* ports, as are the pairs $-y, +y$ and $-z, +z$.

Topology C: The first four-neighbor topology can be derived by splitting each six-neighbor node into two subnodes, each of which contains exactly two opposing ports. Without loss of generality, let the first subnode be assigned to ports $\{-x, +x, -z\}$, and let the second subnode be assigned to ports $\{-y, +y, +z\}$. The two subnodes are then connected by a vertical link, forming the topology of Figure 3. Alternately, one can derive this topology by removing x links in odd z planes and removing y links in even z planes of the six-neighbor mesh. Each node has four neighbors, with two in the directions $(0, 0, \pm 1)$. Nodes on even z planes contain neighbors in the directions $(\pm 1, 0, 0)$, while nodes on odd z planes contain neighbors in the directions $(0, \pm 1, 0)$.

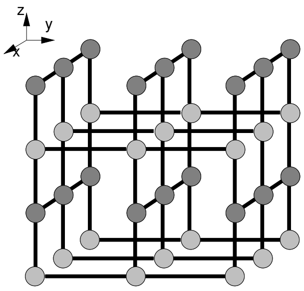


Figure 3: 4-neighbor topology with opposing ports (Topology C)

Topology D: The second four-neighbor topology can also be formed by splitting each six neighbor node into two subnodes, but with a different grouping where each subnode contains no opposing ports. For example, let the first subnode be assigned to ports $\{-x,-y,-z\}$, and the second subnode be assigned to ports $\{+x,+y,+z\}$. If one connects the two subnodes with a vertical link, then the topology can be viewed as the removal of alternating x and y links from the six-neighbor mesh, as shown in Figure 4.

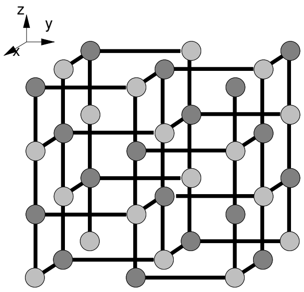
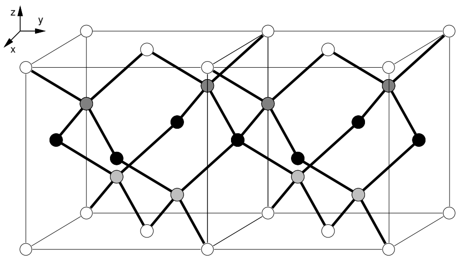
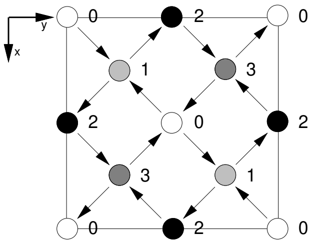


Figure 4: 4-neighbor topology with no opposing ports (Topology D)

If the orientation of links of the above topology are modified, then we obtain the same physical representation as the structure of carbon atoms in a diamond crystal. Two views of the diamond topology are shown in Figure 5, with the picture in Figure 5b representing the view from the top (z direction) of Figure 5a. In this two-dimensional projection, the number next to each node represents its z coordinate modulo 4, while an arrow represents a link that travel upwards towards the reader. The nodes in this lattice can be viewed as a subset of the integral nodes, specifically, nodes (n_1, n_2, n_3) such that $n_1 \bmod 2 = n_2 \bmod 2 = n_3 \bmod 2$. Since all links are diagonal, all neighbor offsets are in the set $(\pm 1, \pm 1, \pm 1)$. Like the previous topology, there are two types of nodes, each with four neighbors. The first type of node has an even number of +'s in all its neighbor offsets, while the second type has an odd number of +'s in all its neighbor offsets.



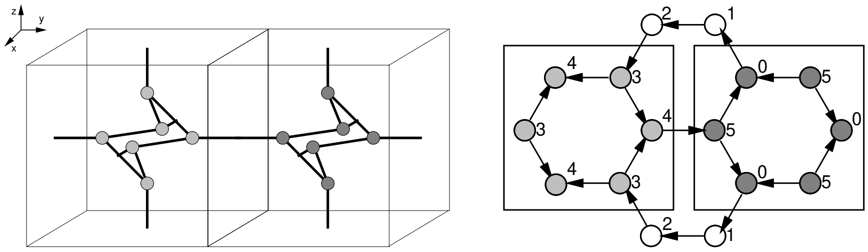
(a)



(b)

Figure 5: 4-neighbor diamond lattice (Topology D)

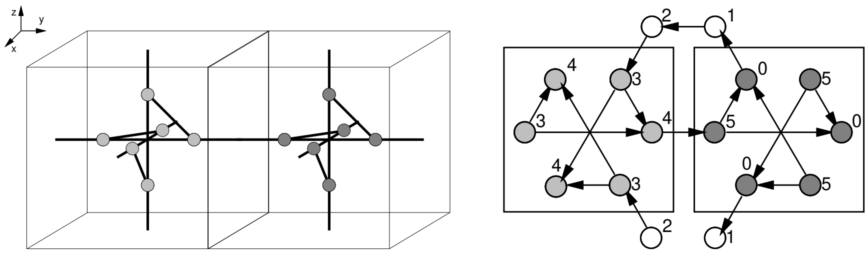
Topology E: This topology is created by splitting each node in the 6-neighbor mesh into six subnodes. Each subnode is associated with one of the 6-neighbor links. A ring is then formed among the six subnodes, with the constraint that no direct connections are formed between two subnodes with opposing links (for example, subnodes with $+x$ and $-x$ links are not connected). The resulting 3-neighbor topology is shown in Figure 6a. A two-dimensional projection of this topology can be formed by viewing the topology from the left upper front corner, producing the view shown in Figure 6b. Again, numbers on nodes represent the height of the node, while arrows represent links that travel out of the page.



(a) (b)

Figure 6: 3-neighbor Topology E

Topology F: The final topology is formed once again by splitting each node in the 6-neighbor mesh into six subnodes. Again, the subnodes are associated with the 6-neighbor links and are formed into a ring. However, this ring follows the constraint that subnodes with opposing links are always connected (for example, subnodes with $+x$ and $-x$ links are connected). The resulting topology is shown in Figure 7.



(a) (b)

Figure 7: 3-neighbor Topology F

3 Topology isomorphism

Since topologies are defined in terms of the logical node connectivity, many physical representations may exist for a particular topology. However, it may be very difficult to determine whether some of these physical representations are indeed equal. In this section, we present a strategy for formally defining topologies in terms of the logical connectivity. From this, a technique can be derived to detect isomorphism between different physical representations of topologies. Readers who are primarily interested in the performance comparisons between topologies may wish to defer this section until later.

A topology $T = [L, M]$ is defined as a set of links L and a set of paths M that can be reached by some traversals of the links from a *reference node*. We define the group $S(T)$ to represent all paths using the links in L. Thus each link in L can be viewed as generators for $S(T)$, and the group operator is merely the concatenation of paths. Since the links can be represented by vectors in space, $S(T)$ must be abelian (commutative).

As an example, consider Topology A (T_A), whose links can be defined as the set $\{X, Y, Z\}$. The mapping f_A from logical links to physical links can be defined as: $f_A(X) = (1, 0, 0)$, $f_A(Y) = (0, 1, 0)$, $f_A(Z) = (0, 0, 1)$. Any element in $S(T_A)$ can be represented in the form $X^aY^bZ^c$, which is translated to physical coordinates as the path from $(0, 0, 0)$ to (a, b, c). For a more complex example, consider Topology B (T_B). Let the links of T_B be $\{W, X, Y, Z\}$, with the mapping f_A to physical links as: $f_B(W) = (\frac{1}{2}, \frac{1}{2}, \frac{1}{2})$, $f_B(X) = (\frac{1}{2}, -\frac{1}{2}, -\frac{1}{2})$, $f_B(Y) = (-\frac{1}{2}, -\frac{1}{2}, \frac{1}{2})$, $f_B(Z) = (-\frac{1}{2}, \frac{1}{2}, -\frac{1}{2})$. Again, the group $S(T_B)$ can be represented by the elements of the form $W^aX^bY^cZ^d$. However, there is an important difference: whereas each representation $X^aY^bZ^c$ of $S(T_A)$ represents a different element for different values of a, b, c, the same is not true for $S(T_B)$. Notably, for any a, $W^aX^aY^aZ^a$ is equal to $W^0X^0Y^0Z^0$ or the identity element, representing a null path. This can be verified by using the mapping to vectors.

In the above two examples, each path in $S(T)$ from the reference node is also in the set of topology paths M. However, this is not the case for other topologies, such as the diamond topology T_D. Although four links $\{W, X, Y, Z\}$ also exist for T_D, with $S(T_D) = S(T_B)$ and $f_D = f_B$, the topologies are different. This can be explained by observing that not all paths in $S(T_D)$ from the reference node are legal. Indeed, any of the eight links $\{W^{\pm 1}, X^{\pm 1}, Y^{\pm 1}, Z^{\pm 1}\}$ are legal from any node in T_B, while only four of the links are available from any node in T_D. Thus only a subset of the paths in $S(T_D)$ can be considered legal paths from the reference node. In this case, we allow links $\{W, X, Y, Z\}$ to be used at any even number of hops from the reference node, and links $\{W^{-1}, X^{-1}, Y^{-1}, Z^{-1}\}$ to be used at an odd number of hops away. From this, we derive the constraint that any legal path for T_D must be of the form $W^aX^bY^cZ^d$ where $a+b+c+d \in \{0, 1\}$. Note that the case of $W^aX^aY^aZ^a = W^0X^0Y^0Z^0$ for T_B is no longer relevant for T_D.

In summary, a topology T is defined as a tuple $[L, M]$ consisting of links and paths from a reference node. The links of L can be used as generators for an abelian group $S(T)$ which defines all paths from the reference node. The set M

is a subset of $S(T)$ and represents the actual legal paths that can be taken from the reference node to form the topology.

This formalism of a topology can then be used to prove isomorphism between different physical representations of topologies. As an example, let us consider the two representations of Topology D, one formed from removing alternate x and y links from the six-neighbor mesh, and the other defined as the physical structure of the diamond lattice. Since a definition for T_D is already derived above, we can show isomorphism merely by showing two consistent mappings to the two physical representations. The mapping to the diamond lattice representation is already discussed above, with $f_D(W) = (\frac{1}{2}, \frac{1}{2}, \frac{1}{2})$, $f_D(X) = (\frac{1}{2}, -\frac{1}{2}, -\frac{1}{2})$, $f_D(Y) = (-\frac{1}{2}, -\frac{1}{2}, \frac{1}{2})$, $f_D(Z) = (-\frac{1}{2}, \frac{1}{2}, -\frac{1}{2})$. The second mapping can be defined as follows: $f'_D(W) = (0, 0, -1)$, $f'_D(X) = (1, 0, 0)$, $f'_D(Y) = (0, 1, 0)$, $f'_D(Z) = (0, 0, 1)$. Note that the restriction of using $\{W, X, Y, Z\}$ on an even number of hops from the reference node and $\{W^{-1}, X^{-1}, Y^{-1}, Z^{-1}\}$ on an odd number of hops is consistent with the illustration in Figure 4.

4 Analytical comparisons

We first compare topologies by applying the conventional analytical measurements of maximum latency and minimum bisection bandwidth. In the following comparisons, we assume that pin-count forms the primary constraint on complexity of the routing chip, and thus normalize the topologies by keeping the pin-count constant. For a number of pins P, a k-neighbor topology requires $k+1$ ports for connections to neighbors and the local processor. Thus the number of bits in the communication path for each channel is $P/(k+1)$.

4.1 Latency

For a given topology, let the volume growth function $V(r)$ be the number of nodes reachable within a distance r from a center node. The set of nodes counted by $V(r)$ can be viewed as a "sphere" of radius r in the given topology. Thus for an n-dimensional topology, $V(r)$ grows as r^n. The maximum latency $l_{\max}$ of a topology can then be measured as the maximum distance between any two nodes in the "sphere", and can be calculated using the inverse of the growth function: $l_{\max} = 2V^{-1}(N)$ for N nodes. The following section discusses strategies for computing the volume growth of different topologies and presents some results of these computations.

For Topology A, the "sphere" of radius r actually resembles the shape of an octahedron with vertices at $(\pm r, 0, 0)$, $(0, \pm r, 0)$, and $(0, 0, \pm r)$. The number of nodes in the octahedron can then be estimated by multiplying its volume by the density of nodes in space. The volume of the octahedron is equal to twice the volume of the pyramid formed from all points above the plane $z = 0$, which has base area $2r^2$ and height r. The volume of the octahedron is thus $\frac{4}{3}r^3$, and since the density of nodes in space is equal to one per unit volume, the number of nodes in the sphere grows as $\frac{4}{3}r^3$.

For Topology B, the shape of the "sphere" becomes a cube with diagonal $\sqrt{3}r$. The volume of such a cube is r^3, and the density of nodes is 2 per unit volume, yielding $2r^3$ nodes in a sphere of radius r.

For other topologies, the volume growth functions are computed by curve-fitting experimentally-derived results. Figure 8 shows the volume growths and maximum latencies of each topology for large radii.

Topology	# neighbors	Volume growth	Max latency
A	6	$1.33r^3$	$1.82N^{\frac{1}{3}}$
B	8	$2.00r^3$	$1.58N^{\frac{1}{3}}$
C	4	$1.33r^3$	$1.82N^{\frac{1}{3}}$
D	4	$0.83r^3$	$2.13N^{\frac{1}{3}}$
E	3	$0.40r^3$	$2.71N^{\frac{1}{3}}$
F	3	$1.06r^3$	$1.96N^{\frac{1}{3}}$

Figure 9: Topology volume growth

From Figure 9, it is clear that topologies with lower degrees possess higher latencies. However, maintaining constant pin-count also allows these topologies to possess wider data paths which can reduce the latency for large messages. For a message of length L bits, the maximum number of cycles required to send the message across an unloaded network using wormhole routing can be defined as:

$$\frac{L(k+1)}{P} + 2V^{-1}(N)$$

The graphs in Figure 10 illustrate the latencies for different sizes of networks when sending small and large messages. Note that higher-degree topologies require significantly more time to send long messages across small networks. Also note that for topologies with the same degree, Topology C outperforms D, and Topology F outperforms E.

4.2 Bisection bandwidth

The maximum latency results give an indication of the performance of a topology when the network is lightly loaded. However, a fair evaluation also requires a measure of the capacity of a topology to handle a larger density of message transmissions. The bisection bandwidth is an attempt to analytically estimate this capacity by measuring the lowest number of separated links when a network is divided into two equal halves. In this section, we represent the bisection bandwidth for each topology as a function of the form Br^2, computed as the number of links crossed when a sphere of radius r is split into two equal halves.

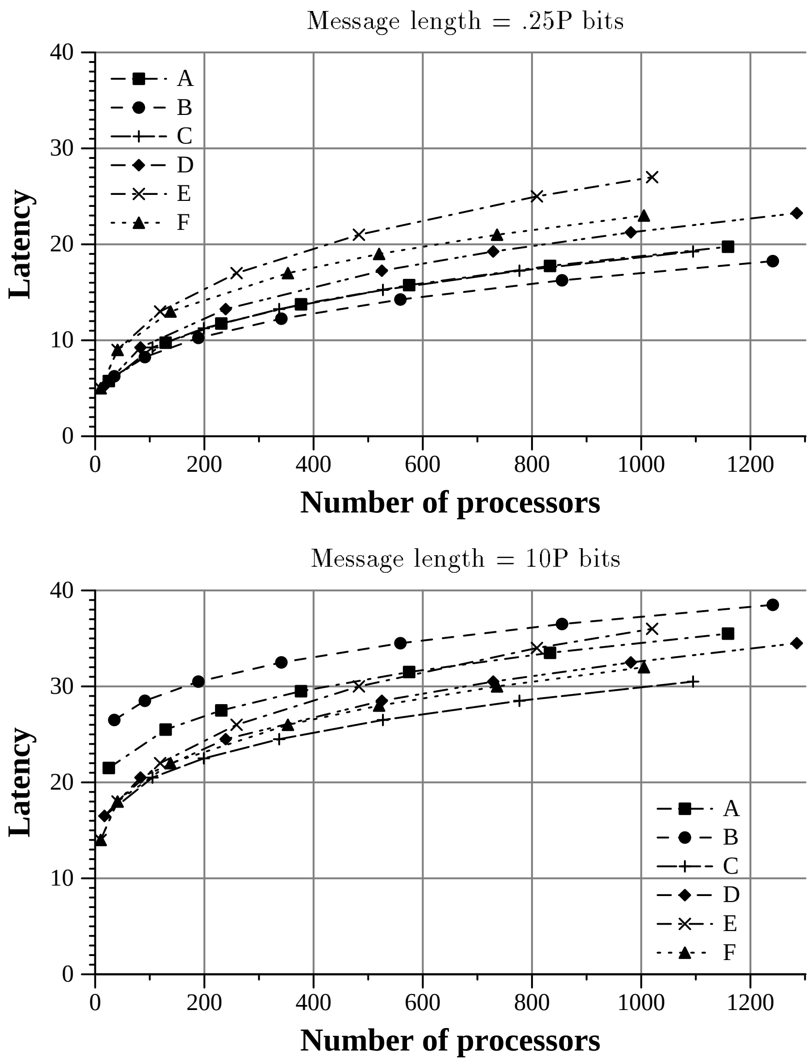


Figure 10: Maximum latency on an unloaded network

The bisection bandwidth of Topology A can be computed by considering the number of links that would be cut by a horizontal plane near $z = 0$. For such a division, the only links that would be cut are in the z direction, and can be computed by considering the number of nodes in the plane $z = 0$ of a radius-r "sphere". Since such a "sphere" takes the shape of an octahedron for Topology A, the cross-section at the plane $z = 0$ takes the shape of a diamond with vertices at $(\pm r, 0, 0)$ and $(0, \pm r, 0)$. The number of nodes in the plane can then be estimated by multiplying the area of the plane by the density of nodes.

The area of the plane is the area of a square with diagonal $2r$, while the density of each node is equal to one per square unit of area. Thus the number of nodes is equal to $2r^2$. The bisection bandwidth of Topology B can be derived with a similar strategy, and is equal to $4r^2$.

For other topologies, the bisection bandwidth is once again derived by curve-fitting computed results. Figure 11 presents bandwidth results as a function of radius and number of nodes, as well as the bandwidth normalized to a constant pin-count P.

Topology	# neighbors	Bandwidth(r)	Bandwidth(N)	Normalized Bandwidth(N)
A	6	$2.00r^2$	$1.65N^{\frac{2}{3}}$	$0.236PN^{\frac{2}{3}}$
B	8	$4.00r^2$	$2.52N^{\frac{2}{3}}$	$0.280PN^{\frac{2}{3}}$
C	4	$1.00r^2$	$0.83N^{\frac{2}{3}}$	$0.166PN^{\frac{2}{3}}$
D	4	$0.75r^2$	$0.85N^{\frac{2}{3}}$	$0.170PN^{\frac{2}{3}}$
E	3	$0.32r^2$	$0.59N^{\frac{2}{3}}$	$0.148PN^{\frac{2}{3}}$
F	3	$0.65r^2$	$0.63N^{\frac{2}{3}}$	$0.156PN^{\frac{2}{3}}$

Figure 12: Bisection bandwidth

The last column of Figure 12 represents the capacity of each topology for a given number of nodes when the channel width is limited by pin-count. This function is further illustrated by the graph in Figure 13, derived from simulation results of bisection bandwidth for particular spheres of each topology. Note that topologies with higher degrees tend to have higher bandwidths, with little difference between topologies of the same degree.

5 Routing simulation

Although the analytical results just presented give an indication of the performance of topologies under some scenarios, their accuracy is constrained by some significant assumptions. The latency measurement only represents delays for the unrealistic case when no contention arises in routing. The bisection bandwidth measurement, on the other hand, assumes the overly-pessimistic situation of total non-locality in communication. Rather than relying on these extremes, we focus instead on a random routing simulation to measure message latencies. Even though this method is not as accurate as a simulation of true program traces, it gives us a more realistic measurement of topology performance.

The results presented in this section are obtained from a uniformly random routing simulation. At every clock tick, each node has a certain probability (the *injection rate*) of injecting a message to a random destination. In order to achieve

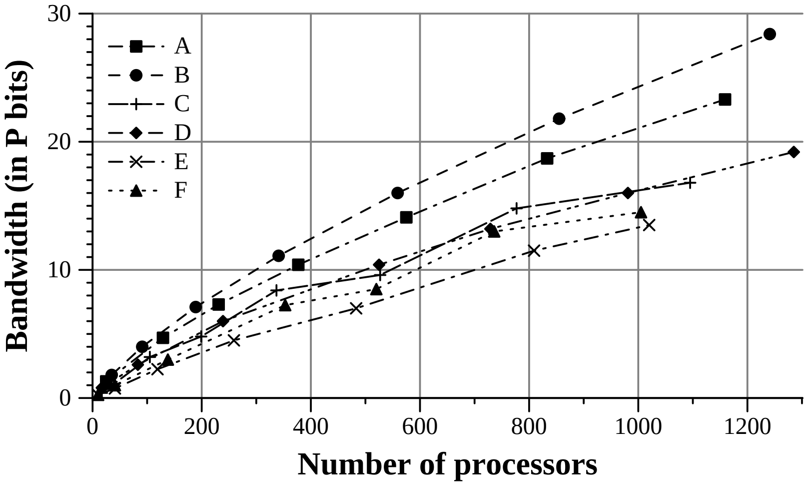


Figure 13: Normalized bisection bandwidths

a minimal and fully adaptive routing algorithm [8], a routing table of size $N \times N$ is precomputed which contains all links that can be taken for any minimal path between each pair of possible source and destination nodes.‡. The table can be computed in time $O(N^3)$ by employing a dynamic programming approach similar to those for computing shortest paths in a graph. Deadlock avoidance is accomplished by imposing no limits on the number of messages that can be placed on a link, thus in effect allowing an infinite number of virtual channels. Although an abort-and-retry approach [5] could potentially be used, we had no easy way of ensuring that no livelocks would arise using such techniques.

Figure 14 shows the simulated average latency of short messages (length P bits) using "spheres" of 256 processors. For each topology, the number of pins are held constant, causing each message to be of length $k+1$ flits for a degree-k topology with $k+1$ ports. Observe that latencies for low loads are very similar. For larger loads, Topology B performs best, followed by Topology A, as indicated by the bisection bandwidth measurements. However, Topology D performs significantly better than Topology C, while Topology F is only able to support a much lower load than any other topologies. This discrepancy with the bisection bandwidth results will be discussed later in this section.

In order to observe any difference in latency due to the higher channel widths of lower-degree topologies, results from a simulation on longer messages (length $5P$ bits) is shown in Figure 15. For light loads, a difference in latency exists between the higher-degree topologies A and B and other topologies. As the load increases, the higher bandwidth of some topologies impose lighter contention penalties on the latencies, resulting in various crossover points in the graph.

‡ Minimal routing algorithms that do not rely on tables also exist for topologies A-D[9]

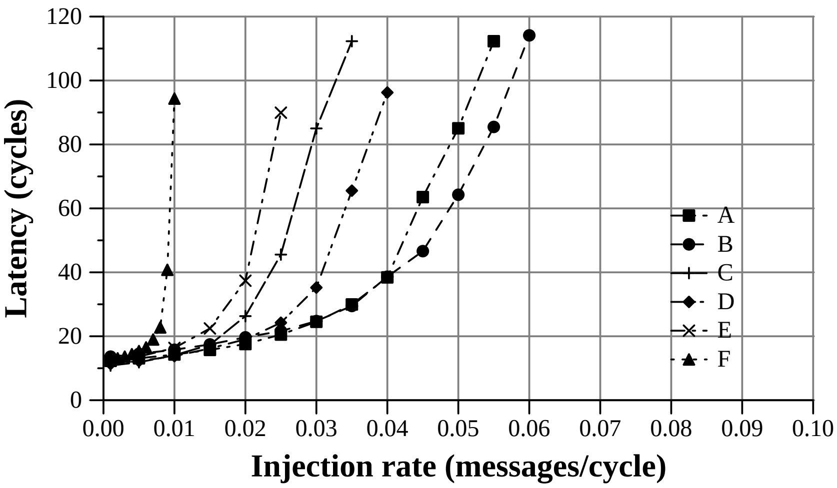


Figure 14: Simulated latencies for messages of length P bits

These crossover points can be used to influence the design decisions for a machine. For example, let us examine the crossover point between topologies A and D at the injection rate of 0.003 messages per cycle. A machine that is optimized for applications with loads lower than an injection rate of 0.003 should employ Topology D, while one that expects much higher loads should be built using Topology A. Note however that the vertical value of the crossover point may also determine its applicability. Although Topology B outperforms Topology A for loads higher than 0.007, one may question whether such a load is relevant since the latencies are already over 140 cycles per message. In order to optimize for such high load demands, it may be more advantageous to employ other factors to improve the speed of the network with respect to the processors.

From the routing simulation results, we see that some topologies do not perform at high loads nearly as well as others with similar bisection bandwidths. This can be partially explained by observing that an adaptive routing scheme works best if a message header has many choices of physical links at each node. With a large number of choices, the header can be assigned to the link with the least contention and improve latency. Even when topologies have similar bisection bandwidths, the average number of choices at each node can different significantly. Figure 16 shows the number of choices of links that a source node has in routing to a destination node, averaged over all sources and destinations of a 256-node "sphere". Note that a number near 1 implies that there are very few routing decisions. This may explain the relatively poor performance of topologies C and F with respect to topologies of the same degree.

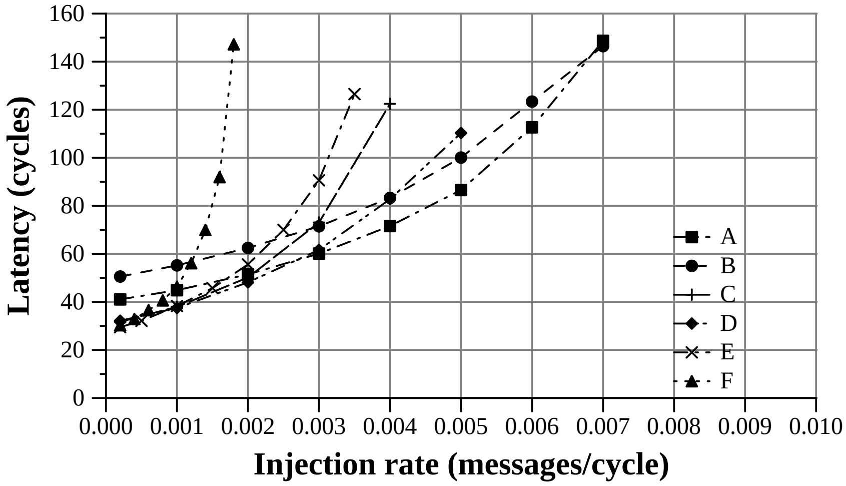


Figure 15: Simulated latencies for messages of length $5P$ bits

Topology	# neighbors	Average choices
A	6	2.19
B	8	2.72
C	4	1.45
D	4	1.53
E	3	1.23
F	3	1.15

Figure 17: Average number of routing choices

6 Conclusion

We have shown that by applying various transformations to the six-neighbor mesh, one can derive new topologies with different characteristics. Despite the existence of many physical representations of individual topologies, a consistent logical definition can be formed based on the connectivity of nodes. This definition can in turn be used to prove isomorphism between different physical representations.

In order to normalize topologies to maintain similar switch complexity, we keep the pin-count constant and vary the channel widths of each topology according to the number of neighbors for each node. Analytical techniques can then be applied to compute maximum latency and bandwidth for comparison. As expected, lower-degree topologies require a larger radius than higher-degree topologies and thus incur higher latencies for very short messages on large networks. However, for long messages, the overhead of sending the message becomes predominant, resulting in superior performance for lower-degree topologies. Simulated bisection bandwidth results predict that higher-degree topologies are able to sustain higher loads even with smaller channels.

We presented the results of a minimal and fully-adaptive routing scheme using unlimited virtual channels to avoid deadlock. On a medium-sized machine, no differences in latency is detectable for small messages on low loads. For larger messages, lower-degree topologies possess lower latencies as predicted. As the message injection rate increases, the average latency of lower-degree topologies increase very quickly, whereas higher-degree topologies can tolerate higher loads, also as predicted by bisection bandwidth. However, some large differences in load tolerance exist between topologies of similar degree and bisection bandwidth. Although there could be many reasons for this, we speculate that one explanation involves the lower number of routing choices of topologies that exhibit poor load tolerance.

The above results are meant as a preliminary comparison of topology performance. As illustrated, the attractiveness of different topologies vary significantly with different message lengths and injection rates. Before any conclusions can be reached about the preferred topology, comparisons using relevant applications and realistic machine models must be performed.

References

[1] Anant Agarwal. Limits on interconnection network performance. *IEEE Transactions on Parallel and Distributed Systems*, 2(4):398–412, 1991.

[2] Thomas H. Cormen, Charles E. Leiserson, and Ronald L. Rivest. *Introduction to Algorithms*. MIT Press, Cambridge, Massachusetts, 1990.

[3] William Dally. Virtual channel flow control. *IEEE Transactions on Parallel and Distributed Systems*, 3(2):194–205, 1992.

[4] William Dally and Charles Seitz. Deadlock-free message routing in multiprocessor interconnection networks. *IEEE Transactions on Computers*, C-36(5), 1987.

[5] Jae H. Kim, Ziqiang Liu, and Andrew A. Chien. Compressionless routing. In *The 21st Annual International Symposium on Computer Architecture*, pages 289–300, 1994.

[6] D. Linder and J. Harden. An adaptive and fault tolerant wormhole routing strategy for k-ary n-cubes. *IEEE Transactions on Computers*, C-40(1):2–12, 1991.

[7] Allen D. Malony. Regular processor arrays. In *The 2nd Symposium on the Frontiers of Massively Parallel Computation*, pages 499–502, 1988.

[8] Lionel M. Ni and Philip K. McKinley. A survey of wormhole routing techniques in direct networks. *Computer*, 26(2):62–76, 1993.

[9] Gill Pratt, Steve Ward, John Nguyen, and Chris Metcalf. The diamond interconnect. In press.

[10] Supercomputing Research Center. Five year review, March 1991.

[11] Steve Ward, et al. A scalable, modular, 3D interconnect. In *1993 International Conference on Supercomputing*, pages 230–239, 1993.