Mapping interconnection networks into VEDIC networks

VIPIN CHAUDHARY, BIKASH SABATA, and J. K. AGGARWAL

Department of ECE
Wayne State University
Detroit, MI 48202

Department of ECE
The University of Texas at Austin
Austin, TX 78712-1084

Abstract

We show the universality of the VEDIC *network in simulating other well known interconnection networks by generating the parameters of the* VEDIC *network automatically. Algorithms are given to represent chordal rings, toroidal meshes, binary hypercubes, k-ary n-cubes, and Cayley graphs - star graph and pancake graph, as* VEDIC *networks. Using these parameters the* VEDIC *network can be used as a tool for generating currently known and new interconnection networks.*

1 Introduction

One of the problems impeding the parallelism attainable in a MIMD multicomputer is interprocessor communication. The interconnection network determines several characteristics of the multiprocessor system, such as performance, expansibility, fault tolerance, etc. In general, the interconnection networks have been classified into dynamic and static networks [1]. This distinction comes from the type of computation performed by the node: dynamic network nodes perform only the routing whereas static network nodes perform both computation and routing. The network topologies are benchmarked by the following features: diameter of the network, degree of the network, efficient and distributed routing strategies, expansibility, and fault tolerance.

Several interconnection network topologies have been suggested in the literature which address one or more of the above features [2, 3, 4, 5, 6, 7, 8, 9]. Since there is no single measure to compare these networks, each of the above examples has been justified for some application. For each of these networks, the diameter, degree, expansibility, fault tolerance, routing strategies, etc. need to be evaluated.

This paper deals with the automatic mapping of the commonly known networks into VEDIC networks [10, 11]. The VEDIC network is described by eight topological parameters; varying the parameters generates different families of networks. By a suitable assignment of the parameters, most commonly known networks are realizable [2, 3, 4, 5, 6, 7, 8]. The network's size, diameter, degree, and number of links are evaluated in terms of the network's parameters. This paper presents algorithms to automatically generate the parameters of the VEDIC network for a commonly known network.

The advantages of such an interconnection network are numerous. First, the VEDIC network can be used as a tool to generate new interconnection networks which are application specific. The desired features of the networks can be obtained by manipulating the parameters. Secondly, the features need to be evaluated only once for the VEDIC network. Substituting the values of the parameters determines the features of the particular network. The unicast and multicast routing strategies for the VEDIC network also hold for all the networks generated. The properties of the algorithms (such as deadlock-free) are also inherited by the networks generated. Deadlock-free multicast wormhole routing strategies have been suggested only for hypercubes and mesh connected multicomputers [12, 13] yet. We have suggested deadlock-free wormhole unicast, single multicast and multiple multicast algorithms for VEDIC networks [14]. Finally, the VEDIC networks provide a common framework for different types of interconnection networks, this can be used to study the interrelationships between the various families of networks.

The rest of the paper is organized as follows. Section 2 discusses the concept of the VEDIC network briefly. Section 3, 4, 5, 6, and 7 give algorithms to automatically generate parameters of VEDIC network for chordal rings, toroidal meshes, binary hypercubes, k-ary n-cubes, and Cayley graphs, respectively. Finally, the concluding section suggests directions of ongoing and future research on the VEDIC network.

2 The VEDIC network

This section describes the families of VEDIC interconnection networks. They are regular or irregular hierarchical networks formed by interconnecting rings of various sizes. The lowest level, i.e., level 0, of the hierarchy is a ring consisting of n nodes. The next level consists of rings, each with an equal or lower number of nodes than the lower ring. Each of the higher level rings necessarily has at least one node in common with a ring in the level immediately below that level. These rings formed at the higher level can also have subsets of other rings (at the same level) in common. The next level is formed by constructing rings using subsets of rings at the immediately lower level (not using any subset of the rings at levels lower than the immediately lower level).

The VEDIC network is represented as $\aleph_0[n, l, m, k, q, w]$, where n is the number of nodes in the ring at level 0, l is the number of levels of the network, m is the maximum difference between the number of nodes at adjacent levels, k is the number of nodes common to two rings at the same level, q is the number of nodes common to two rings at adjacent levels, and w is the distance between two rings at the same level having common nodes.

The distance between rings is defined as the distance between the starting nodes of the two rings. Different variations of m, k, q, and w generate families of networks (details can be found in [10, 11]).

3 Mapping Regular Networks into VEDIC Networks

The VEDIC network can generate other well known networks by fixing some of the parameters. We can automatically generate the parameters of the VEDIC network given the commonly known networks [15]. The VEDIC network in its most general form is a very powerful framework for studying the properties of other networks. These examples show the versatility of the network in modeling a general network [11, 15]. Also, since all the networks are studied in the same framework, the interrelationships between the networks becomes apparent.

3.1 Chordal rings

The chordal rings, defined by Arden and Lee [2], are a family of degree three graphs. The graph is regular and has a very simple representation. The simplicity of the network makes it possible to evaluate different properties of the network and have efficient distributed routing schemes. The network is generated by adding to each node of a ring an additional link, called a *chord*, to some other node across the network. For the chordal ring the number of nodes in the ring n is even, and the distance between the ends of the chord, (the ***chord length***) w_{chord}, is kept constant. Therefore, every odd numbered node i is connected to the $(i + w_{chord}) \bmod n$ node on the ring. The ***chord length*** is assumed to be positive odd. For a ring of size n different chordal rings can be obtained by varying the chord length w_{chord}. The chordal rings structure is also incrementally extensible by adding pairs of nodes to the original network. The figure 1 shows a chordal network of size 16 and chord length 3.

The chordal ring maps into the VEDIC network by mapping the ring to the *level 0* ring and the chords form the *level 1* rings. More specifically, in the VEDIC network, we set n to be even, $m = n - q$, $q = w_{chord} + 1$, $w = 2$, and $k = w_{chord} - 1$. As figure 1 illustrates, by fixing $m = n - q$ the maximum level of the network becomes 1. The figure shows the example for $n = 16$, $w = 2$, $q = w_{chord} + 1 = 4$, and $k = 2$. The indexing of the nodes reduces to just the node number on the ring. Further generalizations of the chordal ring [16] can easily be incorporated into the VEDIC network.

3.2 Toroidal Meshes

Toroidal meshes are mesh connected networks with end-around connections. The end-around connections make the network regular. The torus connected mesh has two parameters, the width W and the height H. In the case when $W = H$ the mesh is a special case of a k-ary n-cube ($k = W$ and $n = 2$). It can be shown that the torus connected mesh always has a Hamiltonian circuit. Using that the VEDIC network parameters can be computed.

Two cases can be distinguished. The first case where the parameter H is even and the second case when it is odd. Figure 2 illustrates the two cases. The Hamiltonian circuit is not unique so the equivalent VEDIC network parameters also vary with the algorithm to compute the Hamiltonian. However, some of the parameters remain independent of the Hamiltonian. For eg., the number of nodes in the level zero ring n remains constant and is $H \times W$; the number of levels l is 2. Figure 2 gives an example of a torus connected mesh with the equivalent VEDIC network.

3.3 Hypercube network

Binary hypercube topology is based on the n-dimensional cube. The binary hypercube is a very regular interconnection, where each node has a degree equal to the dimension of the cube [3]. In a cube of dimension n_{cube}, a node is represented as n_k where k is an n_{cube}-digit binary number. There are n_{cube} neighbors of each node, one corresponding to each dimension. Two nodes n_i and n_j are connected if and only if i and j differ in exactly one digit.

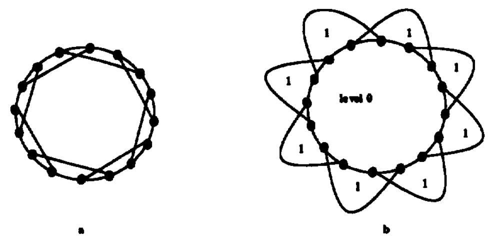

Figure 1: *Example of a Chordal ring and the equivalent* VEDIC *network*

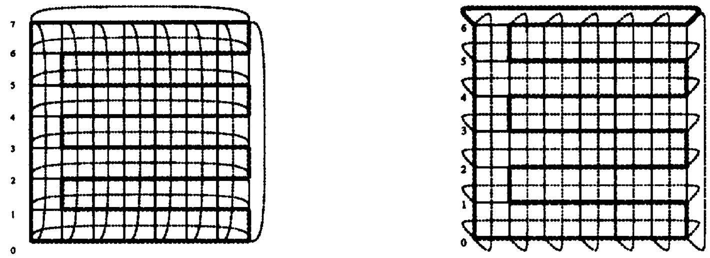

Figure 2: *Example of toroidal meshes. (a) H even (b) H odd; and the equivalent* VEDIC *network.*

The hypercube (or n-cube) has some interesting properties which make it a very useful network. By using these properties and translating them into restrictions in the parameters of the VEDIC network, the hypercube can be generated from the VEDIC network. The significant property here is that the n-cube always has a Hamiltonian circuit. The Hamiltonian circuit is generated by using the binary reflected Gray codes [17]. A simple algorithm to generate the circuit is:

1. Start from node n_0.
2. From n_i go to node n_j such that j is the next Gray code after i.

If the network is traversed by the Hamiltonian circuit, then all the nodes together form a level zero ring. The other interconnections can be presented as chords of this basic ring. The parameters depend on the dimensions of the n-cube. The number of nodes in the level zero ring n is equal to $2^{n_{cube}}$. The number of levels l is 2. The difference between number of nodes in adjacent levels m is $2^{n_{cube}}$. The number of nodes common to rings in the same level k is 0. The distance between start and end of ring q is equal to $\{2^{n_{cube}-1}, 2^{n_{cube}-2} \dots 2^2\}$. The number of rings starting from each node y is $n_{cube} - 2$. Figure 3 gives an example of a 4-cube and the equivalent VEDIC network. Note that this is only one of the ways the network can be generated from the VEDIC network; by considering a different ring as the basic level 0 ring, another equivalent network can be designed. This can also be generalized to the case of cube connected cycles.

3.4 k-ary n-Cube network

k-ary n-cubes are generalization of the binary n-cube where the cube is of dimension n_{cube} and there are k nodes in each dimension. The graph of the network is defined as [18] $G = (V, E)$, where

$$V = \{x \mid x \text{ is a } n_{cube}\text{-digit base-}k \text{ integer, i.e., } x = x_n x_{n-1} \cdots x_1, \text{ and } x_i \in < b >\}$$

and

$$E = \{(x, y) \mid x, y \in V, \text{ and there exists } 1 \le j \le n_{cube} \text{ such that } (x_j - y_j) mod_k = 1 \text{ and } x_i = y_i \text{ for all } i \ne j\}$$

Thus, two nodes in G are connected if and only if their labels differ in exactly one base-k digit by one.

The networks are regular and each node has a degree D where

$$D = \begin{cases} n_{cube} & \text{if } k = 2 \\ 2n_{cube} & \text{else} \end{cases}$$

Like the binary n-cubes the k-ary n-cubes also are assured to have a Hamiltonian circuit. The Hamiltonian is generated by using the generalized cyclic Gray codes

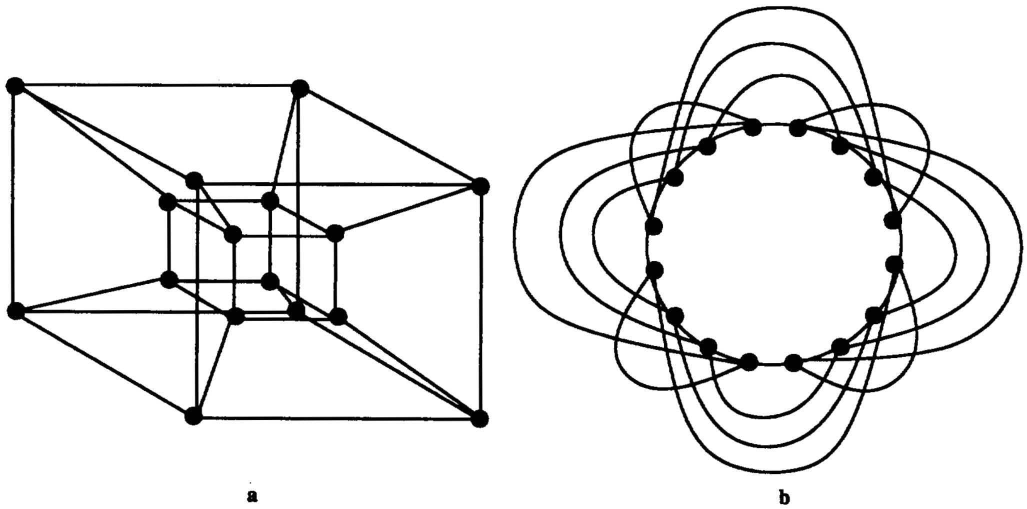

Figure 3: *Example of 4-cube and the equivalent* VEDIC *network.*

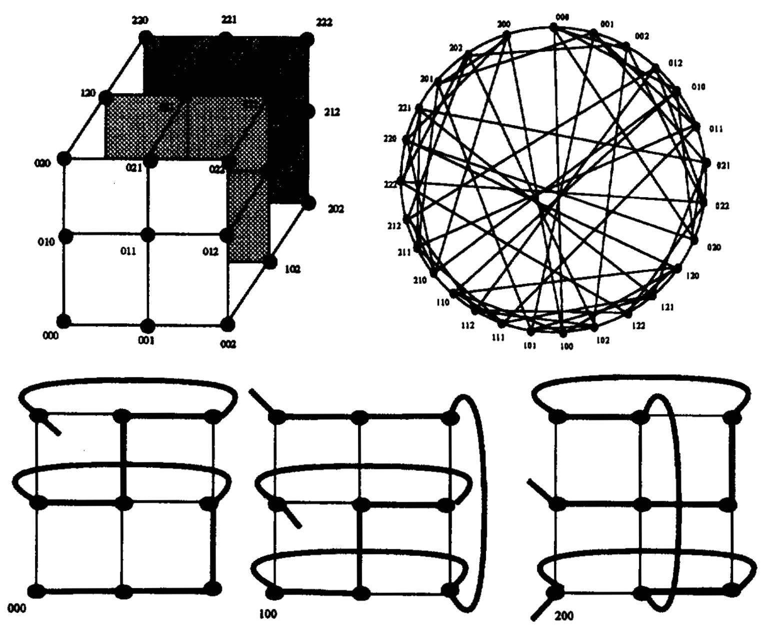

Figure 4: *Example of 3-ary 3-cube, its Hamiltonian cycle and the equivalent* VEDIC *network.*

[18]. There are numerous algorithms to generate these cyclic Gray codes and depending on the algorithm the VEDIC parameters can be evaluated. However, in each case the basic level 0 ring is the Hamiltonian circuit and the other interconnects are the level 1 rings. The level 0 ring size n is $k^{n_{cube}}$. The number of levels l is 2. The difference between number of nodes in adjacent levels m is $k^{n_{cube}}$. The distance between start and end of ring q depends on the Gray coding algorithm used. Figure 4 gives an example of a 3-ary 3-cube and the equivalent VEDIC network.

3.5 Cayley Graphs

Cayley graphs are group theoretic models for designing and analyzing symmetric interconnection networks. Given a set of generators for a finite group G, a *Cayley Graph* is generated by making a graph where the vertices correspond to the elements of the group G and the edges correspond to the action of the generators [9]. Cayley graphs are vertex symmetric graphs and it has been conjectured that there exists a Hamiltonian cycle for all Cayley graphs. For specific graphs the Hamiltonian property can be demonstrated. In this paper we discuss the mapping of two specific examples of Cayley graphs into VEDIC networks.

3.5.1 Pancake Graphs

Pancake graphs are Cayley graphs where the generators correspond to pancake flips. For a size n permutation the flipping of the top i ***pancakes with a spatula*** gives the i^{th} generator. Thus there are $(n-1)$ generators and the graph has $n!$ vertices each with degree $(n-1)$. The Pancake graphs have a Hamiltonian circuit. Suppose the i^{th} generator is denoted by g_i then a edge connected to a vertex can be represented by the corresponding generator and a path between two vertices can be represented by a sequence of generators corresponding to the sequence of edges belonging to the path. The Hamiltonian cycle can be represented as a sequence of $n!$ generators. Consider the following sequence of generators:

(1.0) Take a sequence of $n!$ g_1s.
(2.0) for $i = 2$ to $i = n$
    (2.1) replace every $(i!)^{th}$ symbol with g_i.

It can be shown that the resultant path is a Hamiltonian circuit. Given the Hamiltonian cycle it is easy to evaluate the VEDIC network parameters. The network has only two levels and the base level 0 has $n!$ nodes and the number of level 1 rings from each node is equal to $n-3$. The span of the level 1 rings varies with the node, however it follows a sequence, for eg. in case of $n = 4$ the sequence is $\{6, 13, 8, 18, 13, 20\}$. Similar expressions can be obtained for other n-Pancake graphs. Figure 5 shows a 4-Pancake graph and the equivalent VEDIC network.

3.5.2 Star Graphs

Consider a graph whose vertices are labeled as the permutations of 1 through n. Also, two permutations are connected if by interchanging the first symbol with another symbol in the first permutation results in the second permutation. The resultant graph is the ***star graph*** [9]. Star graphs are attractive alternative to the n-cube because the topological properties are better or comparable to the n-cube. The degree of the graph is $n-1$ and it interconnects $n!$ vertices in the graph while the n-cube interconnects 2^n vertices with degree n.

It has been shown that star graphs have a Hamiltonian cycle [19]. Once the Hamiltonian has been obtained it is easy to construct the equivalent VEDIC network. The parameters of the network are obtained from the topological description of the star graph and its Hamiltonian circuit. Figure 6 illustrates the case of $n = 4$. The level 0 ring has $n! = 24$ vertices. Each vertex has $n - 3 = 1$ level 1 ring originating from it. The span of the level 1 rings varies with the vertex but follows a sequence; in this case $\{6, 16, 8, 18, 10, 20\}$.

4 Conclusion

The main conclusion of this paper is that we have convincingly shown that VEDIC networks are universal and there exist simple algorithms to evaluate the paramaters of the VEDIC networks for most commonly known networks. If the networks to be modeled do not have hamiltonian circuits then the evaluation of the VEDIC parameters is more involved. But, as we have noticed, most commonly used interconnection networks do have hamiltonian circuits.

The VEDIC network as a tool not only models the existing interconnection networks, but also offers a fertile source for generating new network topologies. The universality of the VEDIC network enables us to present the generated networks in a uniform and comparable framework.

One of the immediate advantages of the VEDIC network is to generate new interconnection networks which are application specific. The desired features of the networks can be obtained by manipulating the parameters of the VEDIC network. The features need to be evaluated only once for the VEDIC network. Substituting the values of the parameters determines the features of the particular network.

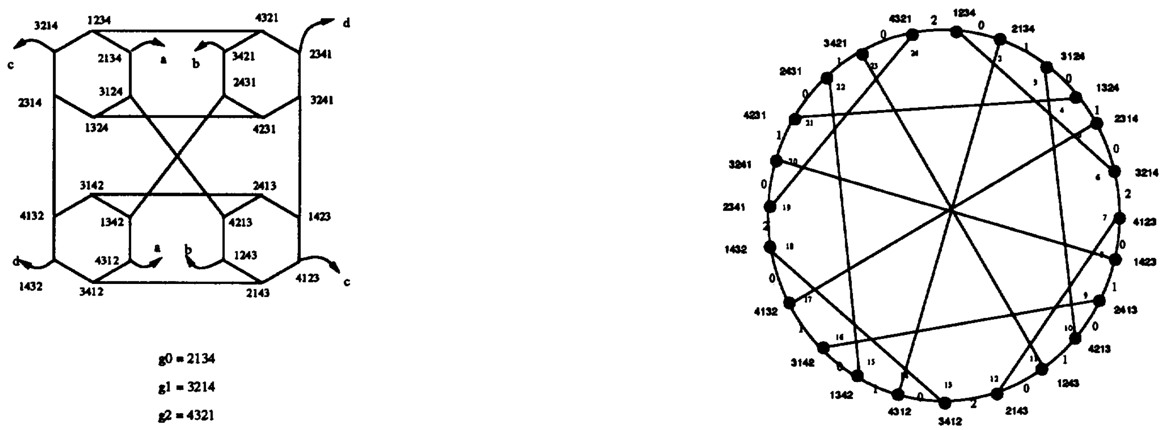

Figure 5: *Pancake graph with n = 4 and the equivalent VEDIC network*

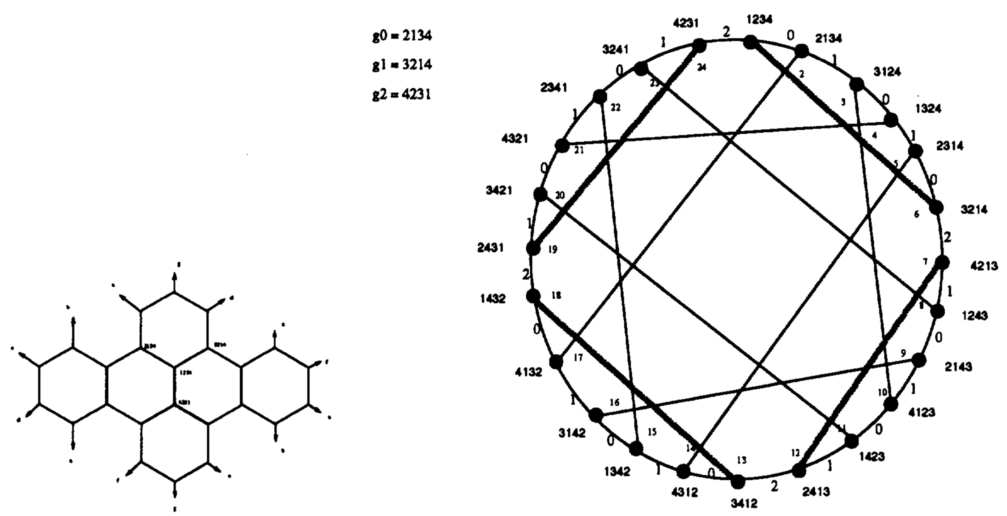

Figure 6: *Star graph with n = 4 and the equivalent VEDIC network*

We have shown the efficacy of the VEDIC network by suggesting deadlock-free wormhole unicast, multicast, and multiple multicast routing algorithms for the VEDIC networks. Since most of the known networks are special cases of the VEDIC network, the proposed unicast and multicast routing algorithms are applicable to all these networks. Thus, we rid ourselves of proposing different multicast routing algorithms for each network.

The VEDIC network in its most general form is a very powerful framework for studying the properties of other networks and their interrelationships. The families of networks generated by this network on varying certain parameters can be investigated in detail. We are currently investigating the generation of interconnection networks for specific applications like image processing and computer vision. We are also mapping the proposed routing algorithms onto the more common networks and optimizing them.

We believe that we have hardly touched upon the possible network topologies that can be generated by the VEDIC network. The areas of future research on VEDIC network would be to quantize the correlation between the various parameters and their physical significance. One could then automatically generate networks for a specific application.

Acknowledgements

This research was suppported in part by IBM.

References

[1] T. Feng, "A survey of interconnection networks," *Computer*, pp. 12–27, Dec. 1981.

[2] B. W. Arden and H. Lee, "Analysis of chordal ring network," *IEEE Trans. on Computers*, pp. 291–295, Apr. 1981.

[3] F. P. Preparata and J. Vuillemin, "The cube-connected cycles: a versatile network for parallel computation," *Commun. of the ACM*, pp. 300–309, May 1981.

[4] R. A. Finkel and M. H. Solomon, "The lens interconnection strategy," *IEEE Trans. on Computers*, pp. 960–965, Dec. 1981.

[5] J. R. Goodman and C. H. Sequin, "Hypertree: a multiprocessor interconnection topology," *IEEE Trans. on Computers*, pp. 923–933, Dec. 1981.

[6] B. W. Arden and H. Lee, "A regular network for multicomputer systems," *IEEE Trans. on Computers*, pp. 60–69, Jan. 1982.

[7] F. T. Leighton, *Complexity issues in VLSI: Optimal layouts for shuffle exchange graphs and other networks.* Cambridge, MA: MIT Press, 1983.

[8] W. J. Dally, "The j-machine: Sysem support for actors," in *Actors: Knowledge Based Concurrent Computing* (Hewitt and Agha, eds.), MIT Press, 1989.

[9] S. B. Akers and B. Krishnamurthy, "A group-theoretic model for symmetric interconnection networks," *IEEE Trans. on Computers*, vol. C-38, pp. 555–566, Apr. 1989.

[10] V. Chaudhary, B. Sabata, and J. K. Aggarwal, "The vedic network for multicomputers," in *Proc. of Int. Conf. on Parallel Processing*, pp. 686 – 687, 1991.

[11] V. Chaudhary, B. Sabata, and J. K. Aggarwal, "The vedic network for multicomputers," Tech. Rep. TR-91-7-71, Computer and Vision Reseach Center, The University of Texas at Austin, 1991.

[12] X. Lin and L. M. Ni, "Deadlock-free multicast wormhole routing in multicomputer networks," in *Proc. Int. Symp. on Computer Architecture*, pp. 116–125, 1991.

[13] X. Lin and L. M. Ni, "Performance evaluation of multicast wormhole routing in 2d-mesh multicomputers," in *Proc. Int. Conf. on Parallel Processing*, 1991.

[14] V. Chaudhary, B. Sabata, and J. K. Aggarwal, "Multicast routing in vedic networks." Submitted for publication.

[15] V. Chaudhary, B. Sabata, and J. K. Aggarwal, "Mapping interconnection networks into vedic networks." Submitted to International Parallel Processing Symposium.

[16] K. W. Doty, "New designs for dense processor interconnection networks," *IEEE Trans. on Computers*, pp. 447–450, May 1984.

[17] N. Deo, *Graph Theory with applications to Engineering and Computer Science.* Prentice Hall, 1974.

[18] S. Lakshmivarahan and S. K. Dhall, *Analysis and Design of Parallel Algorithms: Arithmetic and Matrix Problems.* Supercomputing and Parallel Processing, New York: McGraw-Hill, 1990.

[19] M. Nigam, S. Sahani, and B. Krishnamurthy, "Embedding hamiltonaian cycles and hypercubes into star graphs," in *Proc. of Int. Conf. Parallel Processing*, Aug. 1990.