DUAL-CUBES: A NEW INTERCONNECTION NETWORK FOR HIGH-PERFORMANCE COMPUTER CLUSTERS

Yamin Li and Shietung Peng
Faculty of Computer and Information Sciences
Hosei University
Tokyo 184-8584 Japan
E-mail: {speng,yamin}@k.hosei.ac.jp
TEL: +81-42-387-4544
FAX: +81-42-387-4560

ABSTRACT

The binary hypercube, or n-cube, has been widely used as the interconnection network in parallel computers. However, the major drawback of the hypercube is the increase in the number of communication links for each node with the increase in the total number of nodes in the system. This paper introduces a new interconnection network for large-scale distributed memory multiprocessors called dual-cube. This network mitigates the problem of increasing number of links in the large-scale hypercube network while keeps most of the topological properties of the hypercube network. We investigate the topological properties of the dual-cube, compare them with other hypercube-like networks, and establish the basic routing and broadcasting algorithms for dual-cubes.

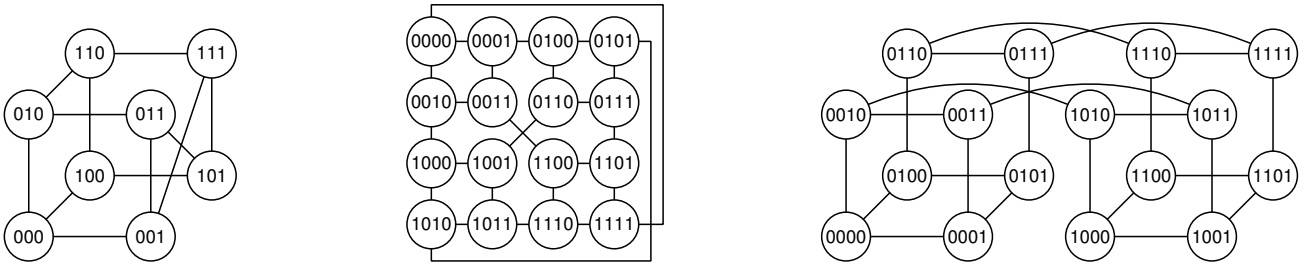
1. INTRODUCTION

The binary hypercube has been widely used as the interconnection network in a wide variety of parallel systems such as Intel iPSC, the nCUBE [4], the Connection Machine CM-2 [7], and SGI Origin 2000 [6]. A hypercube network of dimension n contains up to 2^n nodes and has n edges per node. If unique n-bit binary addresses are assigned to the nodes of hypercube, then an edge connects two nodes if and only if their binary addresses differ in a single bit. Because of its elegant topological properties and the ability to emulate a wide variety of other frequently used networks, the hypercube has been one of the most popular interconnection networks for parallel computer/communication systems.

However, the conventional hypercube has a major shortage, that is, the number of links per node in a system increases logarithmically as the total number of nodes in the system increases. Since the number of links is limited to eight per node with current IC technology, the total number of nodes in a hypercube parallel computer is restricted to several hundreds. Therefore, it is interesting to develop an interconnection network which keeps most of topological properties of hypercubes, and increase the total number of nodes in the system with a fixed amount of links per node.

In this paper, we propose a new interconnection network, called *dual-cube*. The dual-cube shares the desired properties of the hypercube, and increases tremendously the total number of nodes in the system with limited links per node. Especially, the following key property of the hypercube is also true in the dual-cube: each node can be represented by a unique binary number such that two nodes are connected by an edge if and only if the two binary numbers differ in one bit only. However, the size of the dual-cube can be as large as eight thousands with up to eight links per node.

Several variations of the hypercube have been proposed in the literature. Some variations focused on the reduction of diameter of the hypercube, such as folded hypercube [1] and crossed cube [2]; some focused on



(a) Crossed cube, CQ(3) **(b) Hierarchical cubic network, HCN(2,2)** **(c) Reduced hypercube, RH(2,1)**

Figure 1. Three hypercube variations (# edges = 3)

the reduction of the number of edges of the hypercube, such as cube-connected cycles [5] and reduced hypercube [8]; and some focused on the both, like hierarchical cubic network [3].

The *diameter* of a network is defined as the maximum of the shortest distances for all pairs of nodes. The diameter of the conventional hypercube is n for the n-dimensional hypercube. This is smaller than that of many other networks with the same number of nodes. Generally, the variations of the hypercube that reduce the diameter, e.g. crossed cube and hierarchical cube, will not satisfy the key property in the hypercube mentioned above. This key property is at the core of many algorithmic designs for efficient routing and communication.

A folded hypercube [1] is constructed from a conventional hypercube by connecting each node to the unique node that is farthest from it. The folded hypercube may perform better than the corresponding conventional hypercube because of its smaller diameter, which is $\lceil n/2 \rceil$ for a network with 2^n nodes. Because the folded hypercube needs an extra link for each node, it results in higher VLSI complexity.

The crossed cube [2] CQ(n) uses the same amount of resources as the conventional hypercube. Its diameter is about half of the diameter of the hypercube, or more precisely, it is $\lceil (n+1)/2 \rceil$ for a network containing 2^n nodes. The crossed cube is constructed by repositioning some of edges in hypercube. Fig. 1(a) shows a crossed cube with 3 edges, or CQ(3).

The hierarchical cubic network [3] HCN(n,n) has 2^n clusters, where each cluster is a n-cube. Each node in the HCN(n,n) has $n+1$ links connected to it. Of these, n links are used inside the cluster. The additional link is used to connect nodes among clusters. Fig. 1(b) shows a hierarchical cubic network with 3 edges, or HCN(2,2). The advantages of the hierarchical cubic network are that the number of links required is reduced approximately to half as many links per node and the diameter is reduced to about three-fourth of a corresponding hypercube.

The cube-connected cycles [5] CCC(n) is constructed from the n-dimensional hypercube by replacing each node in hypercube with a ring containing n nodes. Each node in a ring then connects to a distinct node in one of the n dimensions. The advantage of the cube-connected cycles is that the node's degree is always 3, independent of the value of n.

The reduced hypercube [8] RH(k,m) is obtained from the n-dimensional hypercube by reducing node edges in hypercube by following rules where $k+2^m = n$. There are 2^{2^m} clusters and each cluster is a conventional k-dimensional hypercube. Of the higher $n-k=2^m$ dimensions, a node has only one direct connection to another node. This dimension of the connection is decided by the leftmost m bits in the k-bit field, i.e., the (2^i+k)dimension, where i is the value of the m-bit binary number. Fig. 1(c) shows a reduced hypercube with 3 edges, or RH(2,1). The number of edges of RH(k,m) is reduced to $k+1$.

Origin2000 [6] is constructed with conventional hypercube or folded cube when the number of processors is not so large (less than or equal to 64 processors for instance). Origin2000 reduces the number of links required when the system scale increases by introducing CrayRouters used by the system among hypercubes

(called fat hypercube). CrayRouter is the high level router that does not connect processors directly. The processors are attached to the regular routers within the hypercubes. Each router has six links (one of these is called CrayLink), two links are used to connect two nodes, where each node contains two processors. Therefore, a router has four processors, three local links used for hypercube, and a CrayLink used to connect to a CrayRouter. When the number of processor increases, the number of CrayRouters and the dimension of each CrayRouter will also increase. The largest Origin2000 system can connect up to $2^5 \times 2^3 \times 4$, or 1024, processors. It will use eight 5-dimensional CrayRouters and 256 regular routers.

It is practically important to refine the hypercube networks such that the size of the network can be increased while the number of the links per node is limited by the technology. If the number of links per node is n, the conventional hypercube can connect up to 2^n node; the hierarchical cubic network can connect 2^{2n-2} nodes; while the dual-cube can connect 2^{2n-1} nodes.

A dual-cube uses binary hypercubes as basic components. Each such hypercube component is referred to as a *cluster*. Assume that the number of nodes in a cluster is 2^m. In a dual-cube, there are two *classes* with each class consisting of 2^m clusters. The total number of nodes is $2^m \times 2^m \times 2$, or 2^{2m+1}. Therefore, the node address has $2m+1$ bits. The leftmost bit is used to indicate the type of the class (class 0 and class 1). For the class 0, the rightmost m bits are used as the node ID within the cluster and the middle m bits are used as the cluster ID. For the class 1, the rightmost m bits are used as the cluster ID and the middle m bits are used as the node ID within the cluster. Each node in a cluster of class 0 has one and only one extra connection to a node in a cluster of class 1. These two node addresses differ only in the leftmost bit position.

In a r-connected dual-cube, $r-1$ edges are used within cluster to construct an $(r-1)$-cube and a single edge is used to connect a node in a cluster of another class. There is no edge between the clusters of the same class. If two nodes are in one cluster, or in two clusters of distinct classes, the distance between the two nodes is equal to its Hamming distance, the number of bits where the two nodes have distinct values. Otherwise, it is equal to the Hamming distance plus two: one for entering a cluster of another class and one for leaving.

The rest of this paper is organized as follows. Section 2 describes the dual-cube structure in details. Section 3 discusses the topological properties of the dual-cube. Section 4 gives the routing and broadcasting algorithms. Section 5 concludes the paper and presents some future research directions.

2. DUAL-CUBE INTERCONNECTION NETWORK

A r-connected dual-cube F_r is a undirected graph on the node set $\{0,1\}^{2r-1}$ such that there is an edge between two nodes $u = (u_{2r-1} \dots u_1)$ and $v = (v_{2r-1} \dots v_1)$ in F_r if and only if the following conditions are satisfied:

(1) u and v differ exactly in one bit position i.

(2) if $1 \le i \le r-1$ then $u_{2r-1} = v_{2r-1} = 0$.

(3) if $r \le i \le 2r-2$ then $u_{2r-1} = v_{2r-1} = 1$.

Intuitively, the set of the nodes u of form $(0u_{2r-2} \dots u_r * \dots *)$, where $*$ means "don't care", constitutes a $(r-1)$-dimensional hypercube. We call these hypercubes *clusters* of class 0. Similarly, the set of the nodes u of form $(1 * \dots * u_{r-1} \dots u_1)$ constitutes a $(r-1)$-dimensional hypercube, and we call them clusters of class 1. The edge connects two nodes in two clusters of distinct class is called *cross-edge*. In other word, $\langle u, v \rangle$ is a cross-edge if and only if u and v differ at the leftmost bit position only.

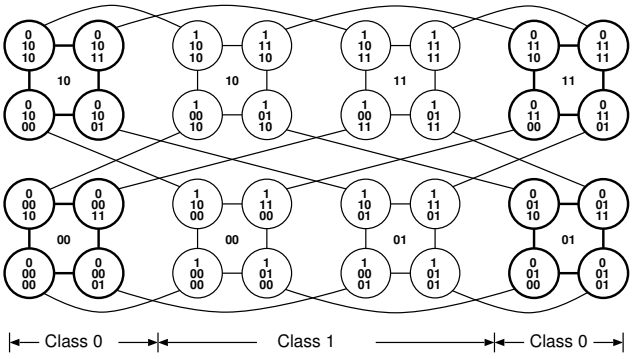


Figure 2. The dual-cubes F_3

We divide the binary representation of a node into three parts: Part I is the rightmost $r-1$ bits, part II is the next $r-1$ bits, and part III is the leftmost bit. For

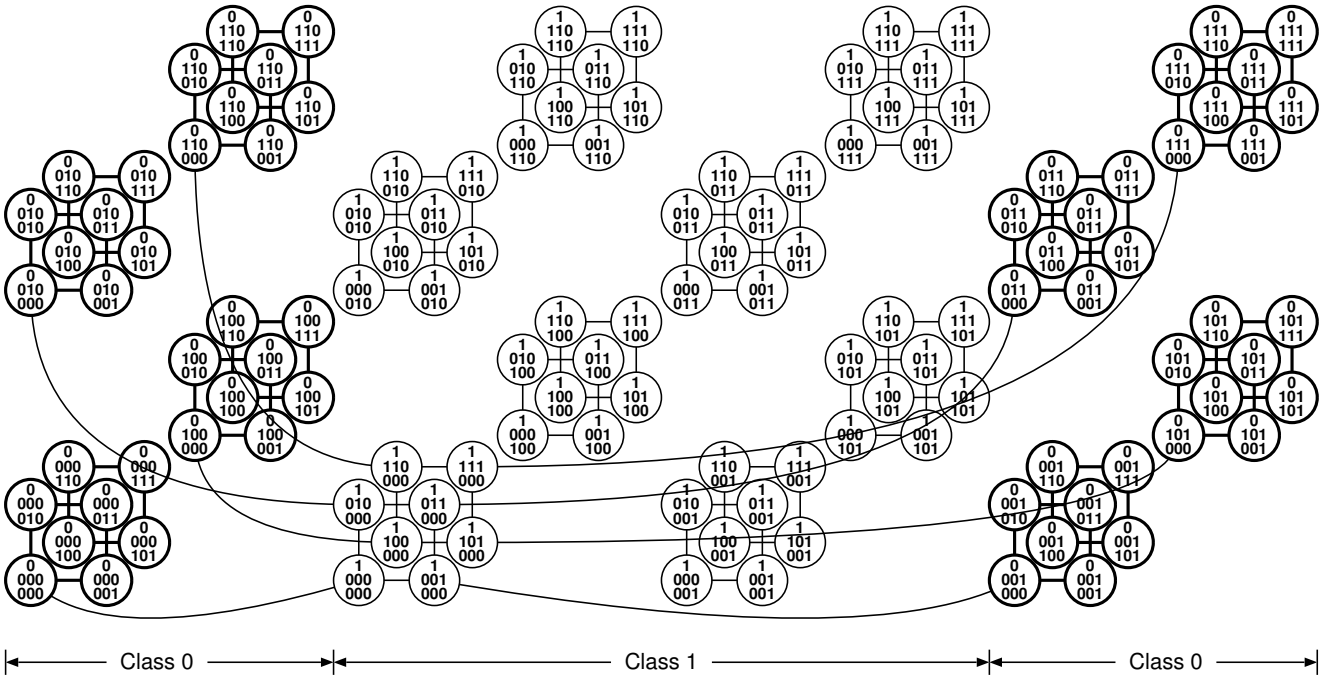


Figure 3. The dual-cubes F_4

the nodes in a cluster of class 0 (class 1), part I (part II) is called *node ID* and part II (part I) is called *cluster ID*. Part III is a *class indicator*. The cluster contains node u is denoted as C_u. For any two nodes u and v in F_r, $C_u = C_v$ if and only if u and v are in the same cluster.

Fig. 2 depicts an F_3 network. The class indicator is shown at the top position in the node address. For the nodes of class 0 (class 1), the node ID (cluster ID) is shown at the bottom, and the cluster ID (node ID) is shown at the middle. Fig. 3 shows only those edges connecting to cluster 0 of class 1.

3. TOPOLOGICAL PROPERTIES

Through this paper, we use the following terminologies. Let G be a undirected graph. A path from node s to node t in G is denoted by $s \rightarrow t$. The length of the path is the number of edges in the path. For any two nodes s and t in G, we denote $d(s,t)$ as the length of a shortest path connecting s and t. The diameter of G is defined as $d(G) = \max\{d(s,t)|s,t \in G\}$. The connectivity of G is defined to be the minimum number of nodes whose removal disconnects G or reduces it to a single node. G is k-connected if its connectivity is k.

A topology is evaluated in terms of a number of parameters such as degree, diameter, bisection width, cost (defined as the product of the degree and diameter), average distance for any two nodes, regularity, symmetry etc. We should consider the network cost as the main parameter for measuring and comparing of the different topologies. Other important measures for networks include the existence of simple routing and communication algorithms. The dual-cube networks have a binary presentation of nodes in which two nodes are connected by an edge if and only if they differ in one bit position, just as in hypercubes. This features is the key for designing efficient routing and communication algorithms on dual-cubes. Another important features of the dual-cubes is that, within the given bound on the number of links per node, say r, the network can have up to 2^{2r-1} nodes, more than the hypercubes or the hierarchical cubes can have with the same bound on the node degree.

Table 1 summarizes the degree, diameter, cost, average node distance, and bisection width of the hypercube and the dual-cube networks, assuming that the two networks have the same number of nodes which is 2^n, where n is an odd integer. The dual-cube shows a

significant gain in the cost of the network. Since the number of nodes in F_r is 2^{2r-1}, we have $r = (n+1)/2$. The diameter and the average node distance of F_r will be shown in the next sections. The bisection width of F_r is obtained by letting F_r^1 = half of the clusters of class 0 + half of the clusters of class 1, and $F_r^2 = F_r - F_r^1$. It is easy to see that the removal of $2^n/4$ edges will disconnect F_r^1 and F_r^2, and this number is the minimum for bisecting F_r.

Next, we consider the problem of recursive construction of F_r from F_{r-1}. Since the number of nodes in F_r is four times the number of nodes in F_{r-1}, we need four F_{r-1} in order to construct a F_r. First, we introduce the following notation: $S_a^{b_2b_1}$ (a, b_1, and b_2 are single bits) is defined as the set of clusters of class a in the ith F_{r-1}, where $i = b_2b_1$ ($0 \le i \le 3$).

The node address in F_r is assigned as follows. Suppose the node address in F_{r-1} has the format of $(ax_{r-2}\dots x_1y_{r-2}\dots y_1)$, where $x_{r-2}\dots x_1$ is a cluster (node) ID and $y_{r-2}\dots y_1$ is a node (cluster) ID for $a = 0$ ($a = 1$), then the node address in F_r has the format of $(ab_2x_{r-2}\dots x_1b_1y_{r-2}\dots y_1)$.

The F_r is constructed as follows: 2^{r-1} clusters of class 0 in F_r are formed by pairwisely connecting nodes in S_0^{00} with nodes in S_0^{01}, and nodes in S_0^{10} with nodes in S_0^{11}; similarly, 2^{r-1} clusters of class 1 in F_r are formed by pairwisely connecting nodes in S_1^{00} with nodes in S_1^{10}, and nodes in S_1^{01} with nodes in S_1^{11}. By following this rule, it is easy to see that if two nodes are connected, their addresses differ in only one bit position: the addresses of two nodes in class 0 (1) differ in b_1 (b_2), and the addresses of two nodes in distinct classes differ in the class indicator bit.

Fig. 4 shows the recursive construction of F_2 and F_3 from F_1 and F_2, respectively. The recursive connections are marked with bold lines and curves. F_1 in Fig. 4(a) is a K_2 that has only two nodes, one for each class. Fig. 4(c) is the same as Fig. 4(b), and Fig. 4(d) is the same as Fig. 2. The b_2b_1 is written in the left side in Fig. 4(b) and Fig. 4(d). It is easy to check that the cross-edges are correctly placed in this construction.

4. ROUTING AND BROADCASTING

The problem of finding a path from a source s to destination t and forwarding a message along the path is known as the routing problem. The broadcasting task is to send a message from a source to all other nodes. Routing and broadcasting are the basic communication problems for interconnection networks. In this section, we will describe routing and broadcasting algorithms for the dual-cube networks.

In order to describe the routing algorithm, we need some notation for the neighbor nodes of a node $s \in F_r$. The r neighbor nodes of s, $s^{(i)}$, $1 \le i \le r$, are denoted as follows: Assume s is of class 0 and $s = (0a_{2r-2}\dots a_1)$, then $s^{(i)} = (0a_{2r-2}\dots a_{i+1}\bar{a}_ia_{i-1}\dots a_1)$, where $1 \le i \le r-1$, and $s^{(r)} = (1a_{2r-2}\dots a_1)$. Assume s is of class 1 and $s = (1a_{2r-2}\dots a_1)$, then, $s^{(i)} = (1a_{2r-2}\dots a_{j+1}\bar{a}_ja_{j-1}\dots a_1)$, where $r \le j \le 2r-2$ and $i = j-(r-1)$, and $s^{(r)} = (0a_{2r-2}\dots a_1)$.

The routing algorithm is as follows. If $C_s = C_t$ then it is the routing in hypercubes. Assume that $C_s \ne C_t$. If C_s and C_t are of different classes, say C_s of class 0 and C_t of class 1, then we first routing s to s' in C_s such that the node ID of s' is the same as the cluster ID of t. Then, routing s' to $t' \in C_t$ through a cross-edge. Finally, t' can be routed to t in C_t. Next, assume that C_s and C_t are of the same class, say C_s and C_t are of class 0. We first routing s to s' in C_s such that the node IDs of s' and t are the same. Then, we route s' to s'' through a cross-edge (the cluster ID of s'' is equal to the node ID of t). Next, we route s'' to t' in $C_{s''}$ (of class 1) such that the node ID of t' is the same as the cluster ID of t. Finally, route t' to t in one step through a cross-edge.

From the above routing algorithm, assuming that s and t are different in $k \le 2r-1$ bits, the length of the routing path is k if $C_s = C_t$ or C_s and C_t are of different classes, otherwise, it is $k+2$. Notice that if C_s and C_t are of the same class (say, class 0)), then the path connecting s and t should pass through a node in class 1. This implies that the shortest path connecting s and t is of length at least $k+2$. Therefore, Theorem 1 is true.

Theorem 1 *Assume that nodes s and t in F_r differ in k bit-positions. The distance between sand t, $d(s,t) = k+2$ if s and t are in the different clusters of the same class; otherwise $d(s,t) = k$. If s and t are of the same class and $k = 2r-2$ then the distance is the same as the diameter of F_r, $d(F_r) = d(s,t) = 2r$.*

In an n-dimensional hypercube, the average distance between any two nodes is $(\sum_{i=0}^{n} C(n,i) \times i)/2^n = n/2$. To calculate the average node distance of F_{2r-1},

Table 1. Hypercube v.s. Dual-cube

Network	Degree	Diam.	Cost	Avg. distance	Bisec. width	# Links
Hypercube	n	n	n^2	$n/2$	$2^n/2$	$2^n n$
Dual-cube	$(n+1)/2$	$n+1$	$(n+1)^2/2$	$n/2+1-1/2^{(n-1)/2}$	$2^n/4$	$2^n(n+1)/2$

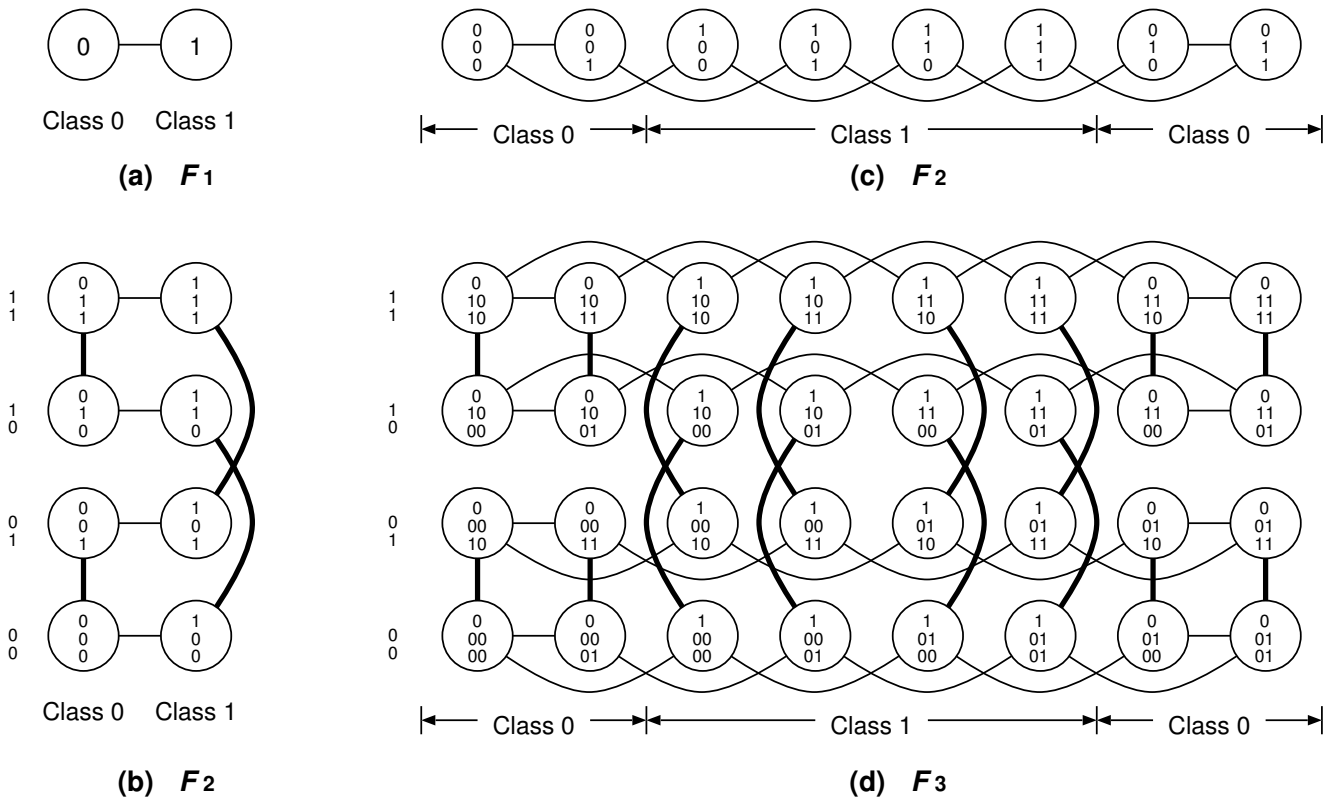


Figure 4. The construction of F_2 and F_3 from F_1 and F_2 respectively

assume that node s is in the cluster of class 0 in F_{2r-1}. Then the average distance between s and nodes u in the clusters of class 1, avg$(d(s,u))$, can be calculated as follows: First, we compute the average distance between s and the 2^{r-1} clusters of class 1. It is $(1/2^{r-1})\sum_{i=0}^{r-1} C(r-1,i)\times(i+1) = 1+(r-1)/2$. Therefore, we get avg$(d(s,u)) = 1+(r-1)/2+(r-1)/2 = r$. Similarly, the average distance between s and node v in the clusters of class 0, avg$(d(s,v))$ can be derived as follows: avg$(d(s,v)) = (2+(r-1)/2-2/2^{r-1})+(r-1)/2 = r+1-1/2^{r-2}$, where the first term, representing the average distance between s and the 2^{r-1} clusters of class 0, comes from the fact that the shortest paths from s to all clusters of class 0 except the cluster containing s, require two cross edges. From the above two formula, we conclude that the average node distance of F_{2r-1} is $r+1/2-1/2^{r-1}$.

The broadcasting process should satisfy some desirable properties: (1) A node should not send (receive) the message simultaneously to (from) more than one of its neighbors; (2) A node receives the message exactly once for the whole duration of the broadcasting process. We show an optimal broadcasting algorithm which completes broadcasting in optimal time (i.e., the

diameter of the dual-cube) under the two restrictions listed above.

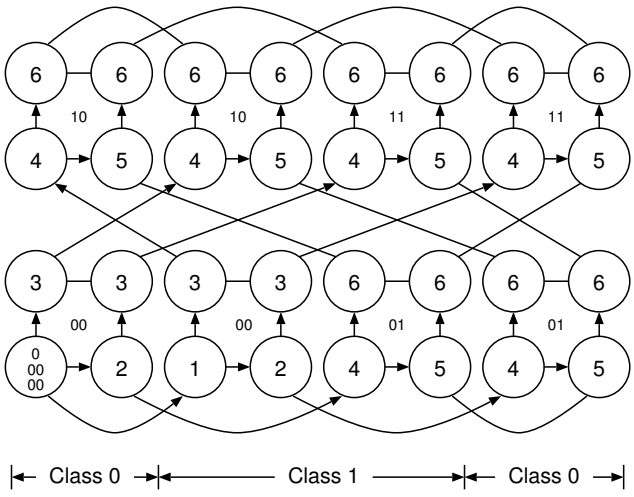


Figure 5. Broadcasting in F_3

The algorithm for broadcasting from a source s works as follows: Assume that C_s is of class 0 (the case that C_s is of class 1 can be done similarly). The source s first send the message to its neighbor $s' = s^{(r)}$ through a cross-edge. Then, s and s' broadcasts simultaneously the message to all nodes in C_s and $C_{s'}$ using binomial trees of C_s and $C_{s'}$ with roots s and s', respectively. Next, every node $u \in C_s \setminus \{s\}$ and every node $u' \in C_{s'} \setminus \{s'\}$ sends the message to its neighbor $v = u^{(r)}$ and $v' = u'^{(r)}$ through a cross-edge, respectively. Finally, every v and v' broadcasts the message to all nodes in C_v and $C_{v'}$.

Fig. 5 shows the broadcasting in F_3. The sending node has the address of 00000. The numbers in the figure are the numbers of steps during the broadcasting. From the above algorithm, the broadcasting is completed in $1 + (r-1) + 1 + (r-1) = 2r$ steps. Therefore, the following theorem is true.

Theorem 2 *Broadcasting in F_r can be done in $d(F_r)$ optimal time under the restricted one-port communication model.*

5. CONCLUSION

In this paper, we proposed a new interconnection network, dual-cube, and showed many attractive properties of the dual-cube including recursive construction, efficient routing and broadcasting. A lot of issues concerning dual-cubes are worth further research. We list some of them below. (1) Investigate the collective communication in dual-cubes. (2) Investigate the embedding of other frequently used topologies into dual-cubes. (3) Develop the techniques for mapping application algorithms onto dual-cubes. (4) Find maximum number of disjoint paths between any two nodes in a dual-cube, or a fault-free path between any two non-faulty nodes in a faulty dual-cube.

References

[1] A. E. Amawy and S. Latifi. Properties and performance of folded hypercubes. *IEEE Transactions on Parallel and Distributed Systems*, 2(1):31–42, 1991.

[2] Kemal Efe. The crossed cube architecture for parallel computation. *IEEE Transactions on Parallel and Distributed Systems*, 3(5):513–524, Sep. 1992.

[3] K. Ghose and K. R. Desai. Hierarchical cubic networks. *IEEE Transactions on Parallel and Distributed Systems*, 6(4):427–435, April 1995.

[4] J. P. Hayes and T. N. Mudge. Hypercube supercomputers. *Proc. IEEE*, 17(12):1829–1841, Dec. 1989.

[5] F. P. Preparata and J. Vuillemin. The cube-connected cycles: a versatile network for parallel computation. *Commun. ACM*, 24:300–309, May 1981.

[6] SGI. *Origin2000 Rackmount Owner's Guide, 007-3456-003*. http://techpubs.sgi.com/, 1997.

[7] L. W. Tucker and G. G. Robertson. Architecture and applications of the connection machine. *IEEE Computer*, 21:26–38, August 1988.

[8] S. G. Ziavras. Rh: a versatile family of reduced hypercube interconnection networks. *IEEE Transactions on Parallel and Distributed Systems*, 5(11):1210–1220, November 1994.