Computer Communications 24 (2001) 1060–1079
computer communications
www.elsevier.com/locate/comcom

Hypercube connected rings: a scalable and fault-tolerant logical topology for optical networks

S. Banerjee[a,1], D. Sarkar[b,*]

[a]Accordion Networks, Inc., 39899 Balentine Drive Suite 335, Newark, CA 94560, USA
[b]Department of Computer Science, University of Miami, Coral Gables, FL 33124, USA

Received 3 November 1999; revised 7 September 2000; accepted 11 September 2000

Abstract

A new, fault-tolerant, scalable, and modular virtual topology for lightwave networks employing wavelength division multiplexing is proposed. The proposed architecture is based on a hypercube connected ring structure that enjoys the rich topological properties of a hypercube, but it also overcomes one of its drawbacks. In a hypercube, the nodal degree increases with the number of nodes. Hence, the per-node cost of the network increases as the network size grows. However, in a hypercube connected ring network (HCRNet) the nodal degree is small and it remains constant, independent of the network population. A HCRNet, like a hypercube, is perfectly symmetric in the sense that the average internodal distance in an N-node HCRNet is the same from any source node. Its average internodal distance is in the order of $\log N$ and it is comparable to other regular structures such as the Trous and ShuffleNet. The HCRNet is based on the Cube Connected Cycle (CCC) interconnection pattern proposed for multiprocessor architectures. However, the HCRNet improves on CCC by rearranging its hypercube links, which results in a significantly lower average internodal distance. In this paper we present the structural properties of HCRNet, and address the issues of scalability, and fast routing in complete as well as incomplete HCRNet. © 2001 Elsevier Science B.V. All rights reserved.

Keywords: Virtual topology; Wavelength division multiplexing; Regular multihop networks; Scalability; Routing; Hypercube; Interconnection network; Cube connected cycles

1. Introduction

Wavelength division multiplexing (WDM) is a proven way of realizing a high level of concurrency in a single optical fiber [1]. However, the use of parallel WDM channels in a traditional *single-hop* fashion suffers from two major drawbacks: (a) requirement of expensive wavelength-agile transceivers and (b) requirement of pretransmission coordination between the prospective communicators. On the other hand, a multihop network, in which a packet from one node to another may be routed via intermediate nodes, does not need fast-tuning transceivers or any pretransmission coordination. Recently, several *multihop* network topologies have been proposed including *ShuffleNet* [2], *Two-Dimensional Torus* (Manhattan Street Network (MSN) [3,4]), *de Bruijn* graph [5], *GEMNet* [6,7], *Hypercube* [8], and *TreeNet* [9] (Fig. 1).

The design of a multihop network architecture should meet the following important requirements: (a) *Average internodal distance should be small* as it is inversely proportional to the utilization of a multihop network. (b) Each node in the network should employ only a *small number* of transceivers so that it is economically attractive. Also, the number of transceivers required at a node should not depend on the network population. (c) Physical embedding of the logical network topology should require only a *small set of distinct wavelengths*. (d) It should be possible to *add or remove nodes* from the network, one at a time, with minimal impact on network configuration and performance. (e) *Routing procedure should be simple* as each node may be required to process hundreds of thousands of packets per second. *Regular* multihop networks are often preferred to irregular or random ones as they usually provide simple routing schemes.

We propose a new regular multihop network architecture, dubbed hypercube connected ring network, or HCRNet, which satisfies the above requirements. This network

* Corresponding author. Tel.: 1 + 305-284-2256; fax: +1-305-284-2864.
E-mail address: sarkar@cs.miami.edu (D. Sarkar).

[1] Supported in part by NSF under Grant No. NCR-9496268, and by the Advanced Telecommunications Institute as part of the Navy Contract N001409-94-C-BC03.

PII: S0140-3664(00)00336-4

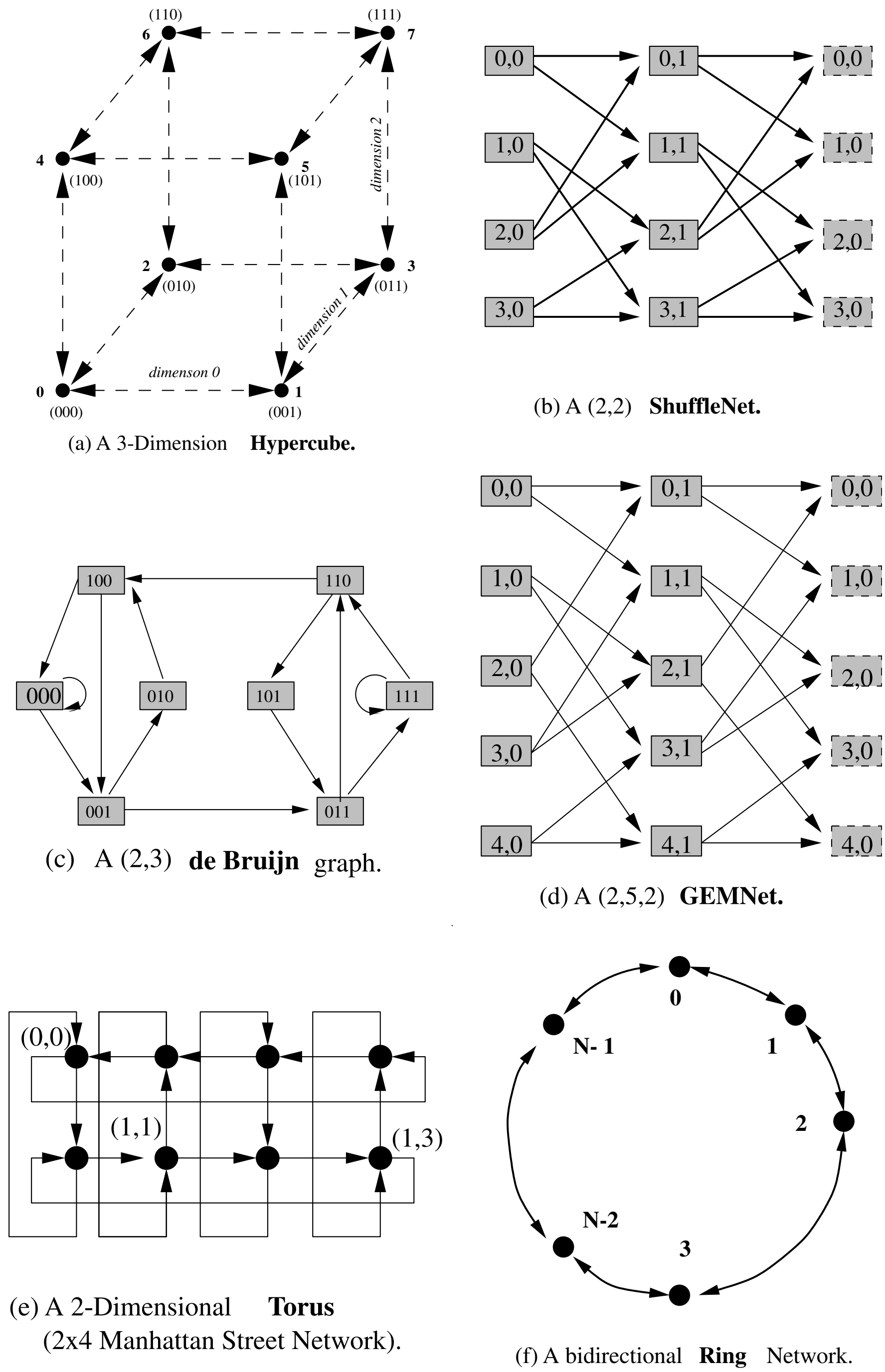


Fig. 1. Various regular multihop virtual network topologies.

topology can be realized as a passive broadcast star network in a local area environment or it can be conceived as a wide area optical network via wavelength routers and converters. As the name suggests, HCRNet is based on the hypercube structure; however, it overcomes a serious limitation of a hypercube as a virtual network topology. Unlike in a hypercube, the nodal degree in HCRNet is independent of the number of nodes in the network. On the other hand, HCRNet enjoys most of the rich structural properties of a hypercube [10]. HCRNet resembles the structure of *Cube Connected Cycles* (CCC) which has been proposed for processor interconnection for parallel computations [11,12]. However, HCRNet improves on CCC by rearranging its hypercube links, which results in a significantly lower average internodal distance. HCRNet's *overall* performance is found to be superior to that of the existing regular structures. It has a better average hop-distance than three-dimensional torus and CCC. Although ShuffleNet and de Bruijn graph have slightly lower average hop-distances than HCRNet, unlike HCRNet they are not scalable and modular. Adding nodes to or deleting nodes from HCRNet involves redeployment of only a small and *constant* number of existing links (i.e. retuning of the existing transceivers).

In the remainder of this section, several regular multihop topologies have been reviewed. A description of HCRNet is provided in Section 2, and its properties are discussed in Section 3. Routing schemes in complete HCRNet are presented in Section 4. Section 5 deals with its scalability issues, and in Section 6 the routing strategy in incomplete HCRNets is addressed.

1.1. Existing regular multihop networks

In this section we briefly review some of the existing regular multihop networks, namely the hypercube, ShuffleNet, de Bruijn graph, GEMNet, Torus, and Ring. Table 1 summarizes some of their properties. A comprehensive survey on regular and multihop network structures can be found in Refs. [13–15,26–28].

1.1.1. Hypercube

An n-dimensional p-ary hypercube consists of p^n nodes. These nodes are labeled in base-p numbers, and two nodes i and j are connected with a bidirectional link if their labels differ in exactly one coordinate position, i.e. Node $(x_{n-1}, x_{n-2}, \ldots, x_i, \ldots, x_0)_{\text{base-}p}$ is connected to Node $(x_{n-1}, x_{n-2}, \ldots, \underline{x}_i, \ldots, x_0)_{\text{base-}p}$ if $(x_i \neq \underline{x}_i)$, $0 \leq i < n$ [10] (Fig. 1a). Thus, the nodal degree of each node is $n(p-1)$. The binary hypercube (i.e. $p = 2$) has been extensively studied as multiprocessor interconnection architecture due to its rich structural properties that naturally supports many parallel algorithms. Several research and commercial hypercube machines were also built (e.g. the cosmic cube, NCUBE, and Intel iPSC). Various fault-tolerant routing schemes have been proposed for *injured* hypercubes for reliability-critical applications [16–20]. The hypercube structure has also been considered as a virtual lightwave network topology. However, it is not regarded as a practical solution since its nodal degree increases with the network size, which necessitates additional expensive nodal interfaces (including transceivers) and a larger set of distinct WDM channels. The average hop-distance[2] in an n-dimensional, p-ary hypercube is given by: $n(1 - p^{-1})$ which reduces to $n/2$ for *binary* hypercubes.

1.1.2. GEMNet, ShuffleNet, and de Bruijn Graph

GEMNet is a recently proposed class of network architectures that includes ShuffleNet (Fig. 1b) and de Bruijn graph (Fig. 1c) as its special members [6,7]. The modularity of GEMNet is one, and it generally has comparable or better properties than de Bruijn graph, ShuffleNet, and MSN. In a (K, M, P) GEMNet, the $N(= K{\cdot}M)$ nodes, each with nodal degree P, are arranged in K columns ($K \geq 1$) and M rows ($M \geq P$) (Fig. 1d). Unlike in ShuffleNet, de Bruijn or Shuffle exchange networks, the number of nodes, M, in a column is not restricted to be of the form P^K. Also, for a given N, there exist as many different GEMNet configurations as there are distinct ways of factoring N into two ordered integers, e.g., for $N = 12$ and $P = 2$, we can have GEMNets with one, three, or four columns and 12, four, or three rows, respectively. GEMNet reduces to a (P,K)-ShuffleNet when $M = P^K$. GEMNet also reduces to a de Bruijn graph of diameter D when $M = P^D$, and $K = 1$. Hence, GEMNet, which is based on a generalized shuffle exchange connectivity pattern, has a much more flexible structure than that of ShuffleNet or de Bruijn graph since the number of nodes is not restricted to any form.

1.1.3. Torus (Manhattan Street Network)

The Manhattan Street Network (MSN), proposed by Maxemchuck, is a regular, two-connected network, and is logically equivalent to a two-dimensional torus [3,4]. The arrangement of its links is similar to the one-way, alternating-direction streets and avenues in Manhattan. In an $N_1 \times N_2$ MSN, $N(= N_1 \times N_2)$ nodes are arranged in N_1 rows and N_2 columns. A 2×4 MSN is shown in Fig. 1e. A considerable amount of research has been conducted on this structure resulting in adaptive deflection routing schemes, schemes for adding nodes to complete and incomplete MSN, and routing in incomplete MSN. However, one drawback of MSN is that its average hop-distance increases at the rate of $\sqrt{N}$, which is much faster compared to the corresponding $\log_2 N$ rate for ShuffleNet-like structures. The average internodal distance in an $N_1 \times N_2$ MSN is given as $(N_1 + N_2 + 4)/4 - 4/N$ for even N_1 and N_2 [21]. The MSN can be easily generalized to higher dimensional tori, an example of a three-dimensional torus is shown in Fig. 2. Recently, it has also been shown that the average internodal distance in a three-dimensional $N(= N_1 \times N_2 \times N_3)$-node torus is given by $(N_1 + N_2 + N_3 + 4)/4 - 4/N$, for N_i $1 \leq i \leq 3$, divisible by 4 [22].

1.1.4. Ring

Ring topologies are attractive for their simple interfaces and control. Token Ring and Fiber Distributed Data Interface (FDDI) are two of the popular networks employing ring topology. A uni-directional ring is minimal in the sense that it uses minimum number of links to achieve full connectivity. A survey of bidirectional and other multiconnected ring topologies can be found in Ref. [23]. In a class of two-connected rings, dubbed *forward loop backward hop* (FLBH), each node has a forward link connecting to its neighbor and a backward link connecting to a node at some *skip* distance a. Thus, in an (N,a)-FLBH network, Node i is connected to Node $(i+1)_{[N]}(x_{[k]} = x \bmod k)$ via the forward link and to Node $(N + i - a)_{[N]}$ via the

[2] Average hop-distance is defined as: $\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} h_{ij}$ where h_{ij} is the minimum number of hops required to reach Node j from Node i.

Table 1
Properties of some regular multihop networks

Properties	r-ary Hypercube	MSN (2D torus)	GEM*Net*	Shuffle-net	de Bruijn	Optimal FLBH ring	HCRNet
Number of transmitters (= no. of receivers) per node (d)	$(r-1)\log_r N$	2	≥ 2	r	r	2	3
Number of nodes, N	r^d	$N_1 \times N_2$ (no restriction)	$M \times K$ (no restriction)	kr^k	r^n	No restriction	nr^n
Order of average hop-distance	$\log_r N$	$\sqrt{N}$ (for $d=2$)	$\log_d N$	$\log_d N$	$\log_d N$	$\sqrt{N}$	$\log_d N$
Diameter, D	$\log_r N$	$\sqrt{N}$	$\lceil \log_d M \rceil + K - 1$	$2k-1$	$\log_d N$	$2(\sqrt{N}-1)$	$n + \lceil n/2 \rceil$
Scalability	Poor	Good	Fair[a]	Poor	Poor	Good	Good
Routing complexity	Very low	Low	Low	Very low	Very low	Low	Low
Symmetricalness[b]	Perfectly symmetric	Good[c]	Good	Perfectly symmetric	Good[d]	Perfectly symmetric	Perfectly symmetric

[a] Requires $O(N)$ wavelength retuning, however, there exists a GEMNet for every N.
[b] Defined as standard deviation of average hop-distance from each node in the network to the other $N-1$ nodes.
[c] Perfectly symmetric if both N_1 and N_2 are even.
[d] Not perfectly symmetric due to the r self-loops.

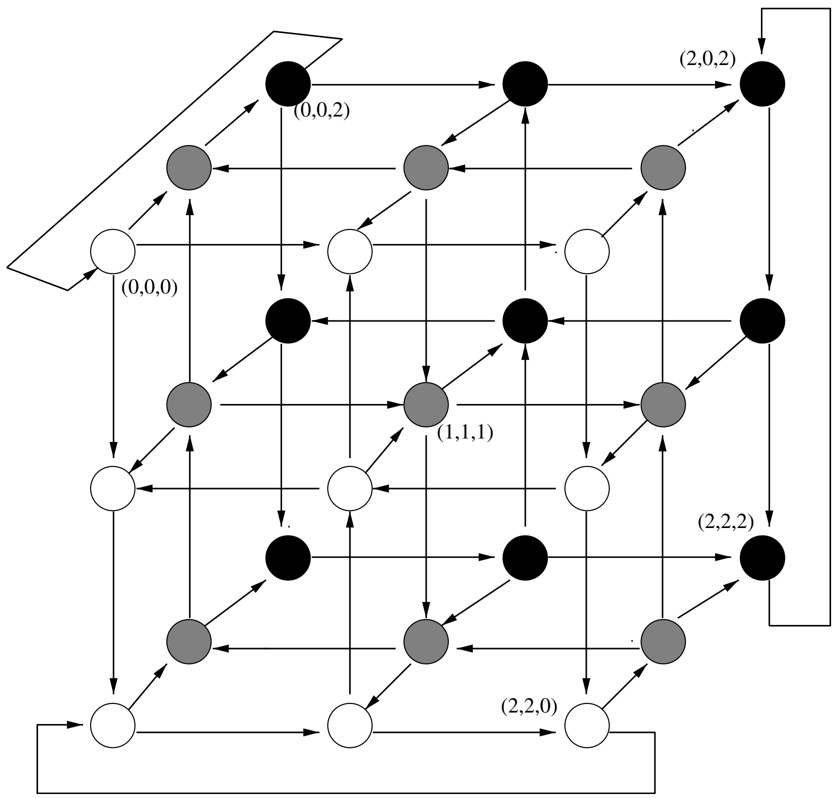


Fig. 2. A three-dimensional $3 \times 3 \times 3$ torous. (Note: some wraparound connections are not shown for clarity.)

backward link. A special member of FLBH class of networks is the bi-directional ring network (Fig. 1f) in which a is unity. However, an N-node FLBH is optimal, in terms of average hop-distance and diameter, when a is set to $\lfloor\sqrt{N}\rfloor$ [24]. For the special case of $N = a^2$, the shortest-path route from Node x to Node y in an N-node FLBH network is specified as two numbers f and b, where $f = \lceil (N + x - y)_{[N]}/a \rceil_{[a]}$ is the number of forward hops (of length 1) and $b = (a^2 + y - x)_{[a]}$ is the number of backward hops (of length a).[3] The forward and backward hops can be taken in any order, and thus, there are ${}^{f+b}C_f$ alternate shortest paths from Node x to Node y. The diameter of a (N,a)-FLBH network, $0 \le a \le \sqrt{N}$, is given by $\lfloor N/(a+1) \rfloor + a - 1$.

2. Description of HCRNet

Consider an n-dimensional binary hypercube with 2^n nodes. Let its nodes be labeled with binary representations of decimal numbers from 0 to 2^{n-1} such that the labels of any two adjacent nodes differ in exactly one coordinate position (Fig. 1a). Let $B_x = (x_{n-1}, x_{n-2}, \ldots, x_0)$ be the binary representation of x. Also, let coordinate i of a node's label denote dimension i, $0 \le i < n$. Note that, if (x mod a) is denoted as $x_{[a]}$, then x_i can be written as: $x_i = (x_{[2^{i+1}]} - x_{[2^i]})/2^i$. Now, the link i of Node x (i.e., the link corresponding to the dimension i), connects to Node $(x + (1 - 2x_i)2^i)$ ($= x \oplus 2^i$, where $\oplus$ is the bitwise *exclusive or* operator).

A Hypercube Connected Ring (HCR) network is obtained by replacing each of the nodes of an n-dimensional hypercube by a ring of n nodes. Thus, the network consists of $N = n2^n$ nodes in 2^n rings each of size n. These nodes are now labeled as (x_h, x_r), where x_h $0 \le x_h < 2^n$, is the label of a ring which is obtained from the corresponding node's label in the original hypercube, and x_r, $0 \le x_r < n$, is the x_r^{th} node in Ring x_h consisting of n nodes. Now, Node (x_h, x_r) is connected to Node $(x_h \oplus 2_r^x, x_r)$ via the x_r-dimension of the hypercube (Fig. 3). Thus, the degree of each node is 3, and even for higher dimensional HCRNets the nodal degree remains unchanged. Note that, since the hypercube links are bidirectional, moving from one dimension to its next one in HCRNet would always involve an extra hop within a ring. For example, Node (000,0) transmits on dimension 0 to Node (001,0), which again transmits on dimension 0 to Node (000,0). Now, for transmitting a packet in dimension 0 followed by a transmission on dimension 1, a hop *within* Ring 001 is needed to reach Node (001,1), which transmits on dimension 1. It is interesting to note that the performance of HCRNet can be significantly improved by staggering the hypercube links as follows: let Node (000,0) transmit on dimension 0 to Node (001,1) and *not* to Node (001,0). Then Node (001,1) transmits on dimension 1 to

[3] For the more general case, $0 \le a \le \sqrt{N}$, the shortest path from Node x to Node y can be computed in constant time as follows. Let $b_{\max} = \lfloor N/(a+1) \rfloor$. Then, b is given as $\lceil (N + x - y)_{[N]}/a \rceil$ if $b \le b_{\max}$, else b is set to 0. Also, f can be computed as $(N + y - (N + x - ba)_{[N]})_{[N]}$.

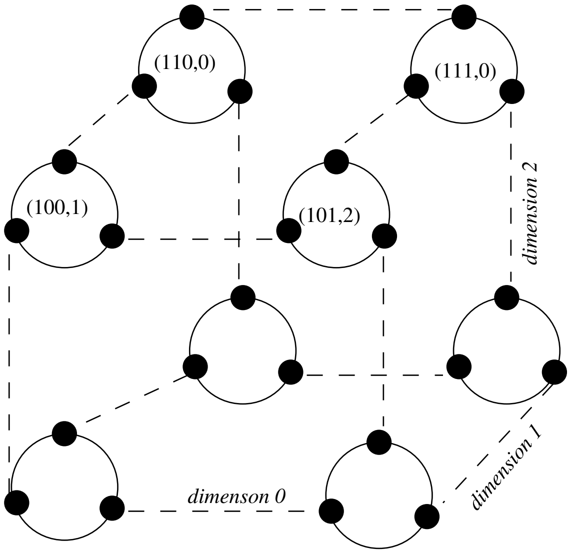


Fig. 3. A 24-node hypercube connected ring network with all bidirectional links.

node (011,2) and so on (see Figs. 4 and 5). This connectivity pattern will eliminate the need of an extra hop for moving from one dimension to *its* next one. Note that, an extra hop will still be required for *skipping* a dimension. In the remainder of this paper, HCRNet will refer to this modified hypercube connected ring network whose connectivity pattern is formally defined as follows:

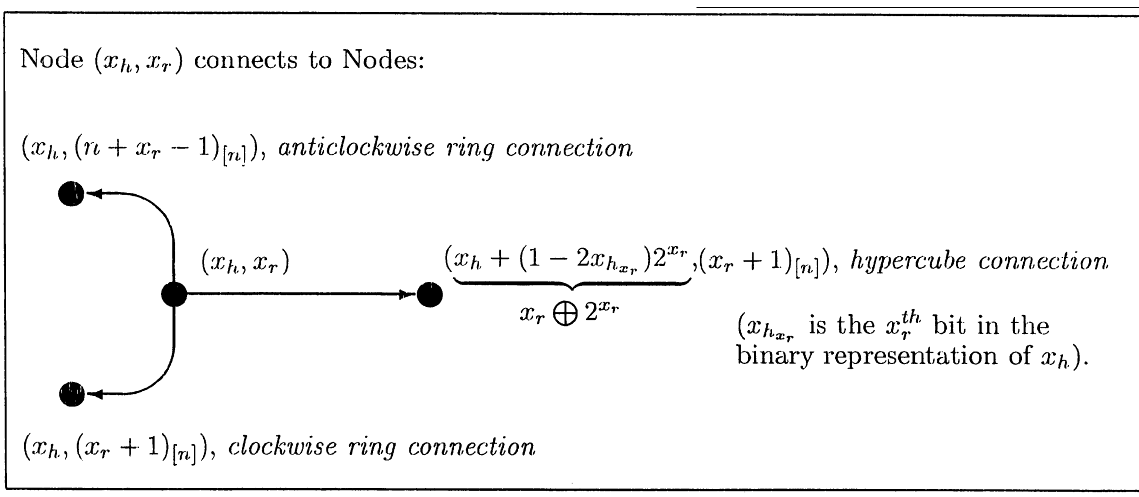
expense of more complicated routing schemes. Since, n increases in the logarithmic scale of the total number of nodes N, n is expected to be small even for large networks, in which case a bidirectional ring structure will be a reasonable choice.

3. Properties of HCRNet

In this section some of the structural properties of HCRNet are discussed. First, the diameter, D, of an n-dimensional HCRNet is given by:

$$D_{\mathrm{HCR}} = n + \left\lfloor \tfrac{1}{2} n \right\rfloor. \quad (1)$$

This can be shown as follows. Since the network is symmetric with respect to every node, without loss of generality, consider Node (0,0) as the source node. Let (d_h,d_r) be a destination node, and let $B_x = (x_{n-1}, x_{n-2}, \ldots, x_1, x_0)$ be the binary representation of d_h. Also, let x_k be the leftmost '1' in B_x, i.e. $k = \max\{i | x_i = 1;\ 0 \le i < n\}$. Then, Ring d_h, which differs in $k+1$ coordinates with respect to the source ring, can be reached[4] in $k+1$ hops from Node 0, since, for $0 \le j \le k+1$, $x_j = 0$ corresponds to traversal of a ring's link and $x_j = 1$ corresponds to the traversal of a hypercube's link. For example, a node in Ring 001010 can be reached from Node (000000,0) in four hops via the following path: (000000,0) (000000,1) (000010,2) (000010,3) (001010,4). Thus, Ring $(2^n - 1)$ is one of the farthest from Node 0, and it can be



A formal definition of HCRNet can be stated as follows: a binary n-dimensional HCRNet is a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ such that $\mathcal{V} = \{(i,j) | i \in [0, 1, \ldots, 2^n - 1]; j \in [0, 1, \ldots, n-1]\}$, $|\mathcal{V}| = N$, and $\mathcal{E} = \{((i,j),(k,l)) | (i \oplus k = 0 \text{ and } |j - l| \in [1, n-1]) \text{ or } (i \oplus k = 2^j \text{ and } l = (j+1)_{[n]})\}$. A *directed hypercube link* in dimension j connects Node (i,j) to Node $(i \oplus 2^j, (j+1)_{[n]})$.

Note that, the bidirectional rings can be replaced by more efficient structures such as the optimal FLBH ring or GEMNet. However, this can be achieved only at the

reached in n hops. It can be easily verified that starting from any given node, any ring in the network can be reached in n hops. Now, reaching the farthest node in the destination ring will take additional $\lfloor n/2 \rfloor$ hops. Hence, the diameter of the network is given by $n + \lfloor n/2 \rfloor$.

Note that, routing from a source Node (s_h,s_r) to a destination Node (d_h,d_r) can be broken-up into three parts: (i) routing within the ring of the source Node s to Node a

[4] Reaching a ring means reaching any one of its n nodes.

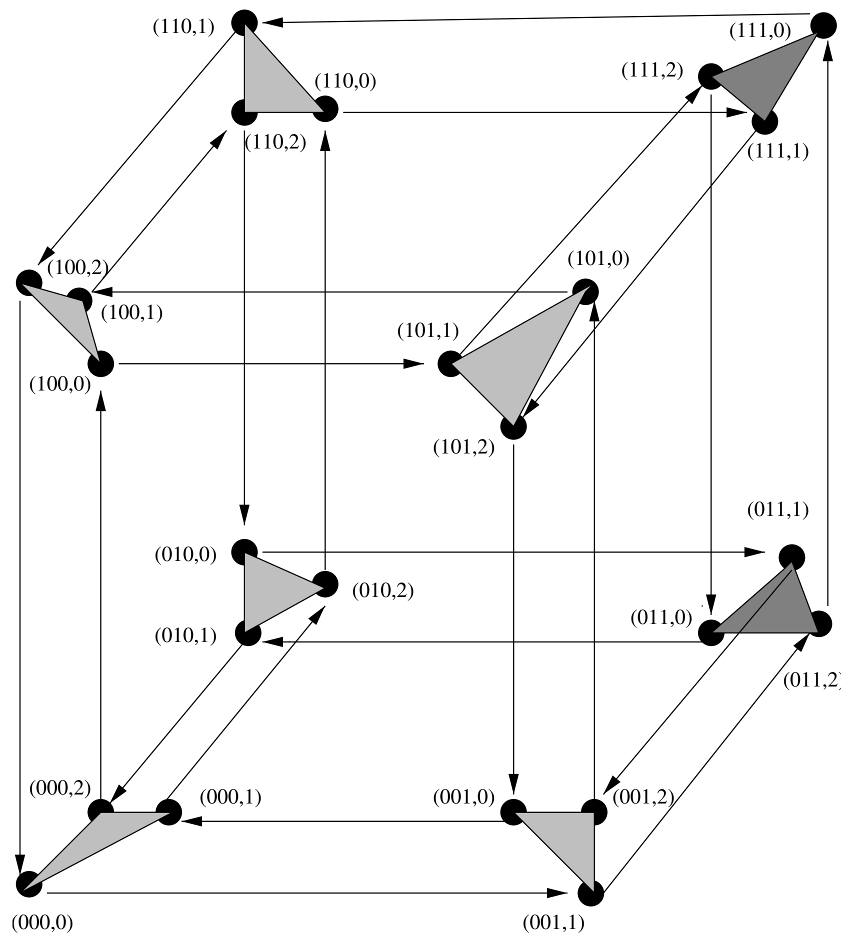


Fig. 4. A 24-node modified HCRNet with unidirectional hypercube links. (Note: all ring-links are bidirectional.)

(say); (ii) routing from Node *a* to a node in the destination node's ring (say Node *b*) via nodes in intermediate rings; and finally (iii) routing within the destination ring from Node *b* to the destination node. Let $C_h = B_{s_h} \oplus B_{d_h}$. Then C_h represents the dimensions of the hypercube in which the source and the destination nodes differ. Also, let $C_h = (c_{n-1}, c_{n-2}, \ldots, c_1, c_0)$, and let $|C_h|$ be the number of 1's in C_h (i.e. C_h is the *Hamming distance* between s_h and d_h). Then, the minimum number of hops required to reach Ring d_h from Ring s_h is $|C_h|$ plus the number of hops in the initial and intermediate rings to *skip* dimensions, if any. For example, let $s = (00000, 0)$, and $d = (10011, 4)$. Then, if routing on the hypercube starts from dimension 0 at the source node, the destination *ring* will be reached in five hops: (00000,0) (00001,1) (00011,2) (00011,3) (00011,4) (10011,0); and the destination node will be reached in one more hop: (10011,0) (10011,4). On the other hand, if the routing on the hypercube starts from dimension 4 at the source ring then the destination ring can be reached in four hops: (00000,0) (00000,4) (10000,0) (10001,1) (10011,2); and the destination node can be reached in two more hops: (10011,2) (10011,3) (10011,4). Thus, in the first case, two hops were 'wasted' for skipping dimensions 2 and 3 in ring 00011, whereas in the second case one extra hop was required in each of the initial and final rings. Although, in both the cases the destination node is reached in six hops, this example shows the trade-offs in choosing the dimension on which the source ring is exited. Note, however, once the source ring is left on the '*optimum*' dimension, the rest of the route becomes fixed.[5]

There exists a *Hamiltonian circuit*[6] in every HCRNet and it can be constructed in a hierarchical way as follows. Since HCRNet is perfectly symmetric, without loss of generality let us consider Node (0,0) as the starting node. Note that, an *n*-dimensional HCRNet can be partitioned, along any one of its *n* dimensions into two identical parts. Each of these parts resembles an $n-1$-dimensional HCRNet, only with the exception of having *n* nodes in each of its ring instead of $n-1$ nodes. We now show that there exists a Hamiltonian

[5] Except for the case when the destination node can be reached in $\lfloor n/2 \rfloor$ hops both in clockwise and anticlockwise directions from the first node reached in the destination ring. (Clockwise direction is defined as $(0\ 1\ 2 \cdots n\ 0)$).

[6] A Hamiltonian circuit in a connected graph $\mathcal{G}$ is a closed path that transverses *every vertex* of $\mathcal{G}$ exactly once, except the starting vertex which is visited twice.

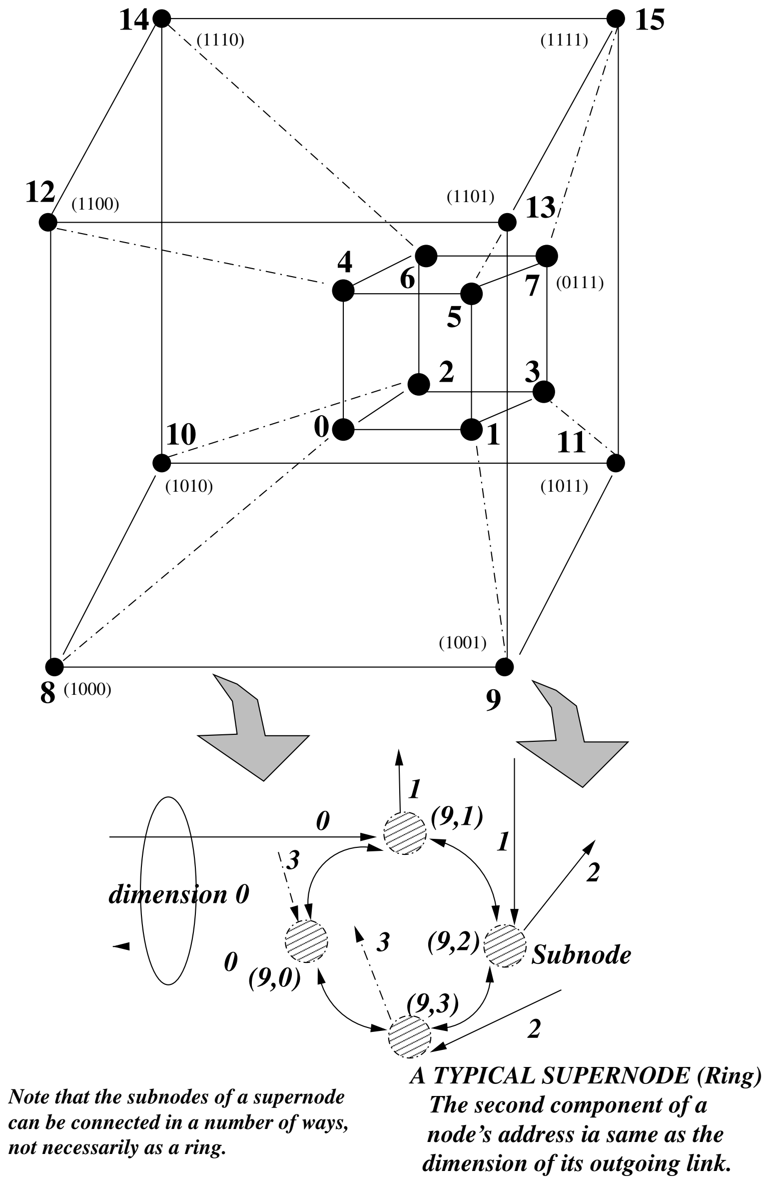


Fig. 5. A 64-node HCRNet.

circuit in each of these two parts, which can be merged to form a Hamiltonian circuit for the entire n-dimensional HCRNet. We start by constructing the Hamiltonian circuit in two dimensions as follows. Let Node ($C_3$000,0) be one of the starting node where C_3 represents the coordinates of dimensions 3 to $(n-1)$. The Hamiltonian path for two-dimensional HCRNet will cover all of its rings which differ only in the first two dimensions with respect to Node ($C_3$000,0) (Fig. 6). This Hamiltonian Circuit can be constructed as follows:

1. Starting from Node ($C_3$000,0) traverse the rings ($C_3 0_{c_1 c_0}$) in the 2-bit gray code[7] order till the last ring (Ring ($C_3$010)) is reached. This may require traversal within some of the intermediate rings which is to be performed in clockwise order.
2. Retrace path to the starting node by traversing the rings in the reverse order, again taking clockwise link(s) within intermediate rings whenever necessary.
3. Similarly, create a Hamiltonian circuit starting from Node ($C_3$100,0).
4. Merge the two Hamiltonian circuits as follows: removing links ($C_3$000,2) ($C_3$000,$3_{[n]}$) and ($C_3$100,2) ($C_3$100,$3_{[n]}$) from the two Hamiltonian circuits; add links ($C_3$000,2) ($C_3$100,$3_{[n]}$) and ($C_3$100,2) ($C_3$000,$3_{[n]}$).

It can be easily verified that the above procedure constructs a Hamiltonian circuit covering all the nodes of all the eight rings that differ in the lower three

[7] Gray code is a special member of cyclic codes in which successive code words differ in exactly one coordinate, e.g. 000, 001, 011, 010, 110, 111, 101, 100.

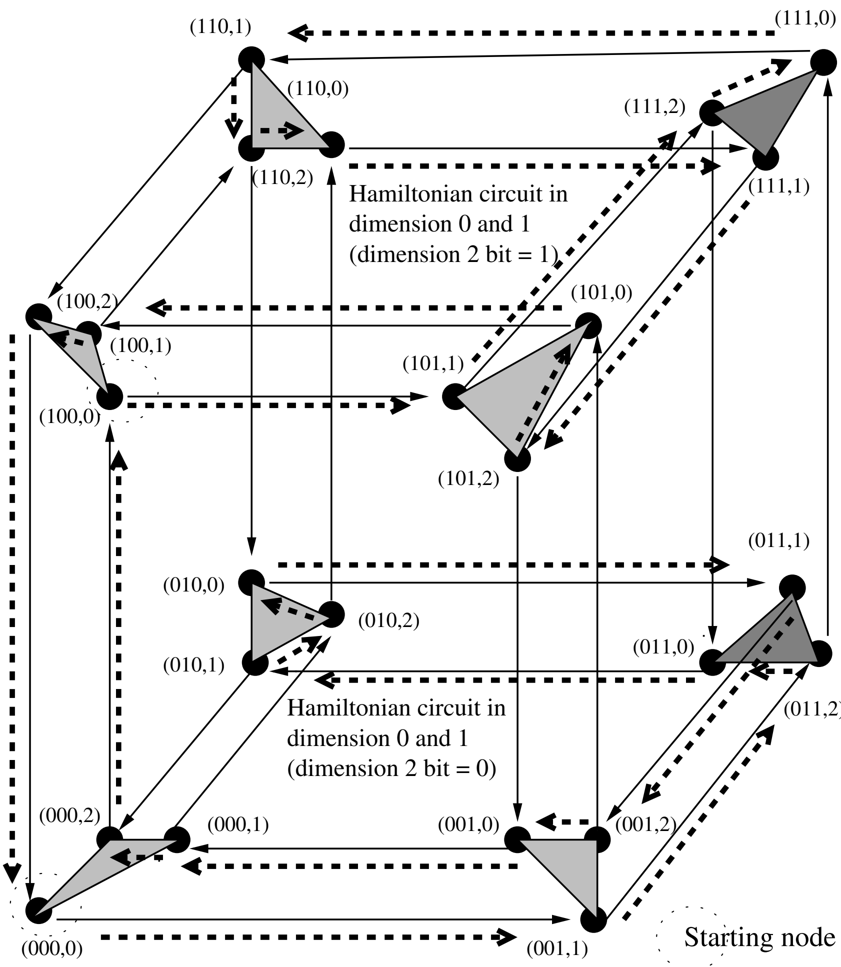


Fig. 6. A Hamiltonian path in a three-dimensional HCRNet.

dimensions with higher $n-3$ dimensions remaining fixed. Next, two Hamiltonian circuits can be obtained in lower *three* dimensions, one for $c_3 = 0$, and the other one for $c_3 = 1$ with the higher $n-4$ dimensions remaining unchanged. In general, two Hamiltonian paths starting at Node $(C_i 0 c_{i-2\cdots c_0}, 0)$ and at Node $(C_i 1 c_{i-2\cdots c_0}, 0)$, each of them covering $n2^{i-1}$ nodes of the 2^{i-1} rings for a given C_i, can be merged by (a) removing the two links $(C_i 00\cdots 0, i-1)$ $(C_i 00\cdots 0, i)$ and $(C_i 10\cdots 0, i-1)$ $(C_i 10\cdots 0, i)$, and (b) adding the two links $(C_i 00\cdots 0, i-1)$ $(C_i 10\cdots 0, i)$ and $(C_i 10\cdots 0, i-1)$ $(C_i 00\cdots 0, i)$. Thus, a Hamiltonian circuit for an n-dimensional HCRNet can be constructed by merging two Hamiltonian circuits obtained for $(n-1)$-dimensional HCRNet.

3.1. An upper bound on average hop-distance

Although the HCR network is symmetric with respect to each node, so far we are unable to obtain a closed-form expression for its exact average hop-distance. However, we present an approximate analysis of its hop-distance, which is later shown to provide a good estimate of the actual average hop-distance. Also, as n increases, the estimated average hop-distance is found to closely approximate the actual average hop-distance.

In obtaining an approximation of average hop-distance we assume that traversal within the source and intermediate rings is allowed only in *clockwise* direction, i.e., $(.,0) \rightarrow (.,1) \rightarrow \cdots (\cdot, n)\ (\cdot, 0)$. This approximation is due to the fact that some of the destination rings can be reached in fewer hops if the shortest path from the source node to the node that is transmitting in the *optimum* dimension in the source ring, is via the anticlockwise direction. Now, since all the nodes will have the same average hop-distance, we assume that Node (0,0) is the source node. Then, assuming only clockwise traversal in the source and intermediate rings, a ring whose address is of the form $(0001x_{k-2}\ x_{k-3} \ldots x_0)$ will be reached in k hops irrespective of the values of x_i, $0 \le i \le k-2$; since $x_i = 0$ would imply traversal within a ring whereas $x_i = 1$ would imply traversal of a hypercube-link. Thus, the number of rings that can be reached in k hops is 2^{k-1} which is the number of combinations of x_i, $0 \le i \le k-2$. Hence, the average hop-distance of a *ring* from the source

Node (0,0) is given by:

$$\bar{R} = \frac{1}{2^n}\sum_{k=1}^{n} k2^{k-1} = n - 1 + 2^{-n}. \quad (2)$$

Now, consider a ring that is R hops away from the source node. Then, all the n nodes in this ring can be reached in an average of

$$\frac{1}{n}\left[R + 2(R+1)\cdots 2\left(R + \frac{n-1}{2}\right)\right] \quad \text{or} \quad R + \frac{n^2-1}{4n} \quad (3)$$

hops if n is odd or in

$$\frac{1}{n}\left[R + 2(R+1)\cdots 2\left(R + \frac{n-1}{2}\right) + \left(R + \frac{n}{2}\right)\right] \quad \text{or} \quad R + \frac{n}{4} \quad (4)$$

hops if n is even. Combining (2)–(4) we get the following expression for the approximate average hop-distance in HCRNet:

$$\bar{H}_{\mathrm{HCR_{approx}}} = \bar{R} + \frac{1}{n}\left\lfloor\frac{n}{2}\right\rfloor\left\lceil\frac{n}{2}\right\rceil = n + \frac{1}{n}\left\lfloor\frac{n}{2}\right\rfloor\left\lceil\frac{n}{2}\right\rceil + 2^{-n} - 1. \quad (5)$$

Note that, for large n, (5) reduces to

$$\lim_{n\to\infty} \bar{H}_{\mathrm{HCR_{approx}}} = \tfrac{5}{4}n - 1 = O(\log N). \quad (6)$$

Also, for a cube connected cycle structure with *bidirectional* hypercube links (see Fig. 2), the approximate average hop-distance can be similarly obtained as:

$$\bar{H}_{\mathrm{HCR_{Bidirectional_{approx}}}} = \frac{3}{2}n + \frac{1}{n}\left\lfloor\frac{n}{2}\right\rfloor\left\lceil\frac{n}{2}\right\rceil + 2^{(1-n)} - 2. \quad (7)$$

For large n, (7) reduces to

$$\lim_{n\to\infty} \bar{H}_{\mathrm{HCR_{Bidirectional_{approx}}}} = \frac{7}{4}n - 2. \quad (8)$$

The performance of HCRNet is compared with some of the other trivalent regular structures, including a three-dimensional torus (Fig. 2), in Fig. 7.

4. Routing in complete HCR network

In this section two routing methods are described, first, a shortest path scheme which takes $O(n)$ time and then a near-optimal and faster routing scheme (see Table 2).

4.1. Shortest path routing

Let R_{ij} denote the shortest distance *from* Node i to Node j in an n-Node bidirectional ring, i.e. $R_{ij} = \min\{|j-i|, n - |j-i|\}$. Consider a source node (s_h,s_r), and a destination node (d_h,d_r). Let $C_h = (c_{n-1}c_{n-1}\cdots c_1c_0) = B_{s_h} \oplus B_{d_h}$, and let $|C_h| = m$, i.e., the source and destination rings differ in m dimensions. Let $A = (a_{m-1},\ldots,a_1,a_0)$ such that $a_i > a_j$ if $i > j$, and $a_i = k$ if c_k is the ith, '1' in C_h; i.e., A is the ordered set of the dimensions in which the source and the destination rings differ. For example, let $s_h = (10100)$, $d_h = (11001)$, then $C_h = (01101)$, $m = 3$, and $A = (3,2,0)$. Now, if the routing starts by first correcting dimension a_i at the source ring, then the last dimension corrected will be the predecessor of a_i, i.e., $l_i = a_{(m+i-1)_{[m]}}$, and hence the destination ring will be first reached at Node $(d_h,(l_i+1)_{[n]})$. Thus, the total number of hops from the source to the destination node is given by:

$$\mathscr{H}^{(sd)}_{a_i} = R_{s_r a_i} + (n - a_i - l_i)_{[n]} + R_{(l_i+1)_{[N]} d_r} \quad (9)$$

Note that, the first term in (9) denotes the number of hops to reach node (s_h,a_i) from the source node, the second term refers to the number of hops required to reach the destination ring from node (s_h,a_i) via intermediate rings and the last term denotes the number of hops required to reach the destination node from the initial node reached in the destination ring. Obviously, the optimum dimension, a_{opt}, for leaving the source ring satisfies the expression: $\mathscr{H}^{(sd)}_{a_{\mathrm{opt}}} = \min\{\mathscr{H}^{(sd)}_{a_i} | 0 \le i < n\}$. Thus, finding the optimum dimension, a_{opt}, takes $O(n)$ time. However, once a_{opt} has been found, computing the shortest route is straightforward and can be obtained in constant time. The source node will generate a binary routing string, $\mathscr{R}$, of length $(R_{s_r a_{opt}} + n)$-bit as follows. Rightmost $R_{s_r a_{opt}}$ bits in $\mathscr{R}$ are set to zeroes. The remaining n bits are obtained by cyclically shifting the bits of C_h such that the a_{opt} bit becomes the rightmost bit. The source node then forwards the packet towards Node (s_h,a_{opt}) via the shortest route (i.e. in clockwise or anticlockwise direction) along with the route information to reach the destination ring. Once the message reaches the destination ring at node (d_h,l_{opt}), it will again be forwarded to the destination node via the shortest path which is to be decided at Node (d_h,l_{opt}). The routing string $\mathscr{R}$ will be interpreted as follows by all the intermediate nodes:

Until (*destination ring is reached*) **do**
If (*message comes on a ring-link from the previous node*)
  if (*next routing bit is 0*) /* starting from the rightmost bit in $\mathscr{R}$ */
    Forward the message to the next downstream node in the same direction
  else /* i.e., next routing bit is 1 */
    Forward the message onto the outgoing hypercube-link
else /* i.e., message came on a hypercube-link*/
  if (*next routing bit is 0*)
    Forward the message to the next node in the same ring in the *clockwise* direction
  else /* i.e., next routing bit is 1 */
    Forward the message onto the outgoing hypercube-link

Example. Let $s = (s_h, s_r) = (010110, 3)$, and let $d =$

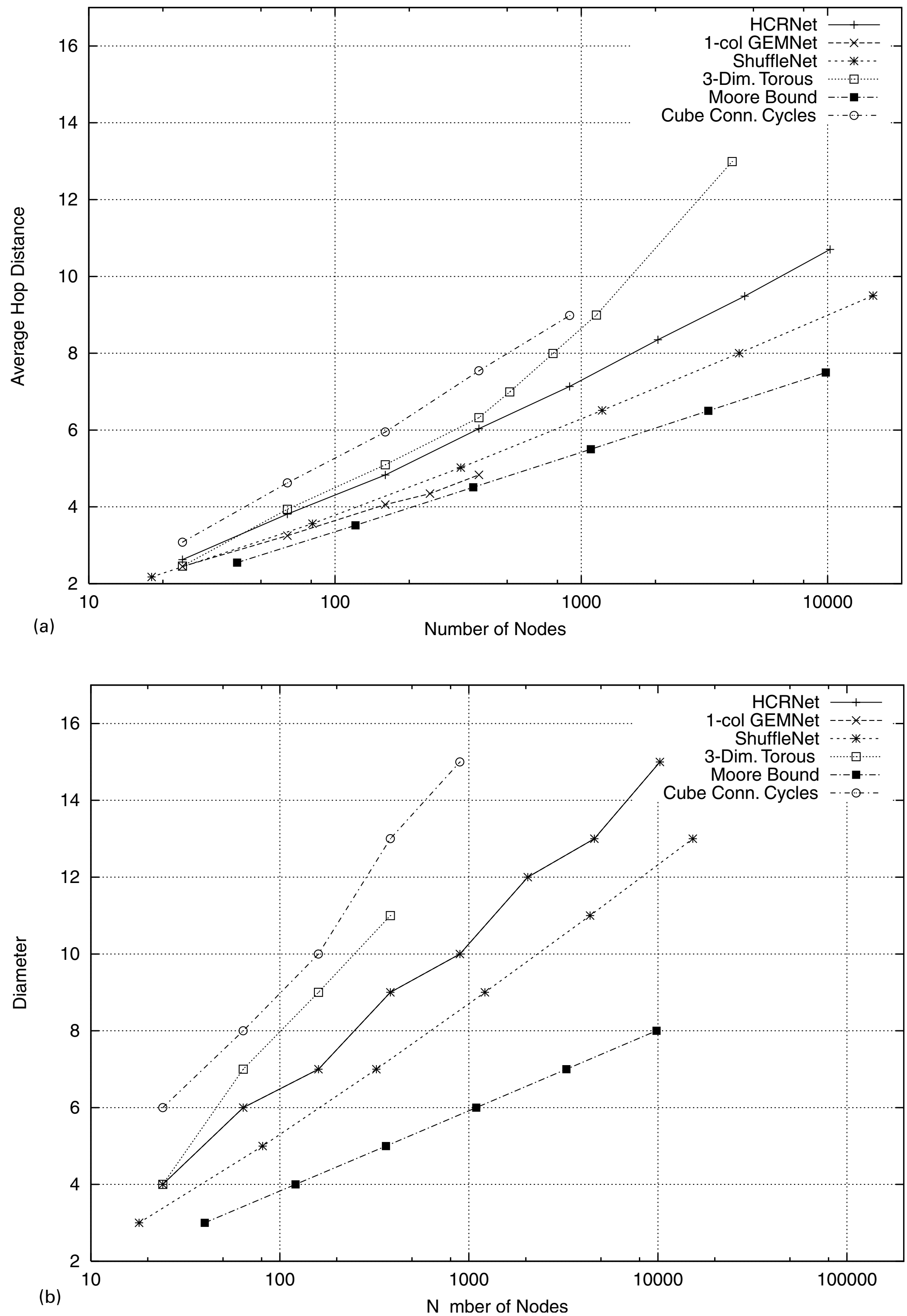


Fig. 7. (a) Comparison of *average hop-distances* in various trivalent regular networks. (b) Comparison of *diameters* in various trivalent regular networks.

$(d_h, d_r) = (110011, 2)$. Then, $C_h = (100101)$, $A = (5, 2, 0)$, and $a_{\text{opt}} = 5$. Then, source Node $(s_h, 3)$ will forward the message to Node (s_h,4) with the routing code $\mathcal{R} = (0010110)$. Node (s_h,4), in response to the rightmost '0' in $\mathcal{R}$, will forward the message to Node (s_h,5). Since the next (i.e., second) routing bit is '1', Node (s_h,5), will forward the message on its outgoing hypercube-link to Node (110110,0). The third routing bit is also '1', hence, the message will be next routed to Node (110111,1). Now, the fourth routing bit is '0', therefore the message will be routed

Table 2
Comparison of the routing schemes

n	3	4	5	6	7	8	9	10	11	12	13	14
N	24	64	160	384	896	2048	4608	10,240	22,528	49,152	106,496	229,376
Optimal (shortest path)	2.625	3.812	4.831	6.036	7.136	8.351	9.487	10.703	11.860	13.076	14.245	15.463
Simple (approx./upper bound)	2.792	4.063	5.231	6.516	7.722	9.004	10.224	11.501	12.728	14.002	15.231	16.500
Efficiency (%)	94.02	93.85	92.36	92.65	92.41	92.74	92.79	93.06	93.18	93.37	93.53	93.67

Table 3
Procedure for scaling-up an n-dimensional HCRNet to an $(n+1)$-dimensional HCRNet by adding one node at a time

Phase	Location of the $(i\cdot 2^n + k)$th new node $(0 \le k < 2^n)$		Retuning operations/new connections (new connections are established as soon as the nodes involved are added to the network)
	i	Location	
1	0	Add the new node to Ring k between Nodes $(k,0)$ and $(k,n-1)$. Label the new node as (k,n). (Choose k in the order of $0,1,2,3,\ldots,2^n-1$)	Node $(k,n-1)$ transmits to and receives from Node (k,n) instead of Node $(k,0)$. Node $(k,n-1)$ transmits to Node $(k \oplus 2^{n-1}, n)$. New connections: Node (k,n) transmits to and receives from Node $(k,0)$ on the wavelengths on which Node $(k,0)$ was transmitting to and receiving from Node $(k,n-1)$. Node (k,n) transmits to Node $(k \oplus 2^0, 0)$. Node (k,n) receives from Node $(k \oplus 2^{n-1}, n-1)$.
2	1	Add the new node the Ring k between Nodes $(k,0)$ and (k,n) Label the new node as $(k,n+1)$ (Choose k in the order of $0,2,1,3,4,6,\ldots$, i.e., in the order of $x_0, x_0+2^i, x_1, x_1+2^i, \ldots$ is the smallest ring number not yet visited in this phase)	Node (k,n) transmits to and receives from Node $(k,n+1)$ instead of Node $(k,0)$. Node (k,n) transmits to Node $(k \oplus 2^0, n+1)$ New connections: Node $(k,n+1)$ transmits to and receives from Node $(k,0)$ on the wavelengths on which Node $(k,0)$ was receiving from and transmitting to Node (k,n). Node $(k,n+1)$ transmits to Node $(k \oplus 2^1, 0)$. Node $(k,n+1)$ receives from Node $(k \oplus 2^0, n)$.
3	2	Create a new ring with the following 2 nodes and label it as $(k+2^n)$. Move Node $(k,n+1)$ to location $(k+2^n,n)$. Add the new node at $(k+2^n,0)$. Choose k in the order of $x_0, x_0+2^{i-2}, x_1, x_1+2^{i-2}, \ldots$ where x_j is the smallest ring number not yet visited in this phase)	Node (k,n) transmits to and receives from Node $(k,0)$ instead of Node $(k,n+1)$ on the wavelengths on which Node $(k,0)$ was transmitting to and receiving from Node $(k,n+1)$. New connections: Bi-directional ring connections involving Nodes $(k+2^n,0)$ and $(k+2^n,n)$. Node (k,n) transmits to Node $(k+2^n,0)$. Node $(k+2^n,n)$ receives from Node $(k \oplus 2^{n \oplus} 2^0, 0)$. Node $(k,2^n,0)$ transmits to Node $(k \oplus 2^{n \oplus} 2^0, n)$.
4	$3 \le i \le n+1$ (in increasing order of i)	Add the new node at $(k+2^n,i-2)$ between Nodes $(k+2^n,i-3)$ and $(k+2^n,n)$ (Choose k in the order of $x_0, x_0+2^{i-2}, x_1, x_1+2^{i-2}, \ldots$ where x_j is the smallest ring number not yet visited in this phase)	Node $(k+2^n,i-3)$ transmits to and receives from Node $(k+2^n,i-2)$ instead of Node $(k+2^n,n)$. Node $(k \oplus 2^n, i-3)$ transmits to Node $(k \oplus 2^n \oplus 2^{i-3}, i-2)$. New connections: Node $(k+2^n,i-2)$ transmits to and receives from Node $(k+2^n,n)$ on the wavelengths on which Node $(k+2^n,n)$ was receiving from and transmitting to Node $(k+2^n,i-3)$. Node $(k+2^n,i-2)$ transmits to Node $(k+2^n+2^{i-3},i-3)$.

within the present ring (110111) in clockwise direction to the next node (110111,2). Finally, since the last routing bit is '1', the message will be routed via hypercube-link to Node (110011,3). Now, Node (110011,3) will recognize that the message has reached the destination *ring* and it will route the message to the destination node via the shortest path.

4.2. Simple and faster routing

In this section we present an $O(1)$ routing scheme. Although this scheme does not always choose the shortest path, its average performance is within 90% of that of the optimal $O(n)$ algorithm described above. Essentially, this scheme is based on the way (5) was derived for approximate average hop-distance. In this scheme, instead of computing the optimum dimension, the source node s either sends the message to the next node in *clockwise* direction or it transmits the message on its hypercube-link if the $s_r \in A$. Now, the routing code is simply given by $\mathscr{R} = (c_{(n+s_r-1)_{[n]}} \cdots c_{(s_r+1)_{[n]}} c_{s_r})$ which can be obtained by cyclically shifting the bits of C_h such that the s_r-bit of C_h becomes the rightmost bit in $\mathscr{R}$. Note that, once the message reaches the destination ring, any superfluous leading '0's in $\mathscr{R}$ will be discarded. This routing strategy can be summarized as follows:

until (*destination ring is reached*) **do**
if (*next routing bit is '1'*) /* starting from the rightmost bit in $\mathscr{R}$ */
  Forward the message on the hypercube-link
else /* i.e., next routing bit is '0' */
  Forward the message on the clockwise-ring-link

4.3. Broadcasting in HCRNet

In this section we describe an efficient broadcasting scheme in HCRNet. The scheme is optimal in the sense that all the rings are reached in shortest possible hops and that the broadcast message reached every node only once. We assume that the message contains a field that identifies it as a broadcast message. We also assume that there exists a simple mechanism to broadcast a message within a ring.

A node $(x_1x_2\ldots x_{n-1}, k)$ initiating a broadcasting in an n-dimensional HCRNet sets routing string, $\mathscr{R}$ as all zero and broadcasts the message within its ring $x_1x_2\ldots x_{n-1}$. For broadcast messages the routing string indicates the dimensions traveled by the message so far. Now *any* node, (Y,q), receiving the broadcast message checks if the message has already traveled any dimension *greater* than or equal to q. If not, it sets the $(q+1)$th bit in $\mathscr{R}$ to 1 and sends it along *its hypercube* link (in dimension q). Note that each node has another responsibility of taking part in broadcasting the message within its local ring. When a node forwards the broadcast message to the next node in the local ring the routing string remains unchanged. The following *example* shows a broadcast from Node (0110,0) in a four-dimensional HCRNet. A detailed study of broadcasting in hypercubes can be found in Ref. [17].

4.4. Edge loading under the routing schemes

It is interesting to note that, under uniform traffic pattern, both the routing schemes result in a perfectly balanced distribution of loads among the hypercube edges. Assuming a traffic of unity between every source–destination pairs, x and y, $x \neq y$, the load on any hypercube link is given by $n^2 2^{n-1}$ for both simple and shortest path routing. This is a property of HCRNet, which it has inherited from hypercube. However, the ring-edges carry a lower load and the excess capacity in the these links can be utilized to bring heavily communicating nodes 'closer' by logically placing them in the same ring [25].

5. Scalability of HCRNet

In this section we describe how an n-dimensional HCRNet can be systematically grown to an $(n+1)$-dimensional HCRNet. A procedure for adding nodes, one at a time, to an existing HCRNet is presented in this section. A routing scheme in such *incomplete* HCRNets is presented in the following section.

5.1. Scaling up an n-dimensional HCRNet to an (n + 1)-dimensional HCRNet

A systematic way of adding nodes in a complete HCRNet, one at a time, can be exploited for efficient routing in the resulting *incomplete* HCRNet. The sequence of adding nodes is also important in reducing the number of retunings required at the existing nodes. Note that, to form an $(n+1)$-dimensional HCRNet, the total number of nodes to be added to an n-dimensional HCRNet is given by: $[(n+1)^{2n+1} - n2^n] = (n+2)\cdot 2^n$. Given below is the outline of the sequence in which these nodes are added followed by a more formal description.

The first 2^n nodes are added to the existing 2^n rings in the order of increasing ring numbers. The new node in Ring x, $0 \le x < 2^n$, placed between Nodes $(x,0)$ and $(x, n-1)$ and is labeled as (x,n). Each ring now consists of $(n+1)$ nodes. Also, the wavelength assignments at the transceivers of Nodes $(x, n-1)$, (x,n), $(x \oplus 2^{n-1}, n)$, and $(x \oplus 2^0, 0)$ are set such that Node $(x, n-1)$ transmits to Node $(x \oplus 2^{n-1}, n)$ and Node (x,n) transmits to Node $(x \oplus 2^0, 0)$ (Fig. 8). The next 2^n nodes are also added to the existing 2^n rings, in the increasing order of ring numbers. Now, in Ring x, this new node is placed between Nodes $(x,0)$ and (x,n) and is labeled as $(x,n+1)$. Again, transceivers at some of the nodes are retuned such that Node (x,n) transmits to Node $(x \oplus 2^0, n+1)$ and Node $(x,n+1)$ transmits to Node $(x \oplus 2^1, 0)$. Now, each ring consists of $(n+2)$ nodes, one

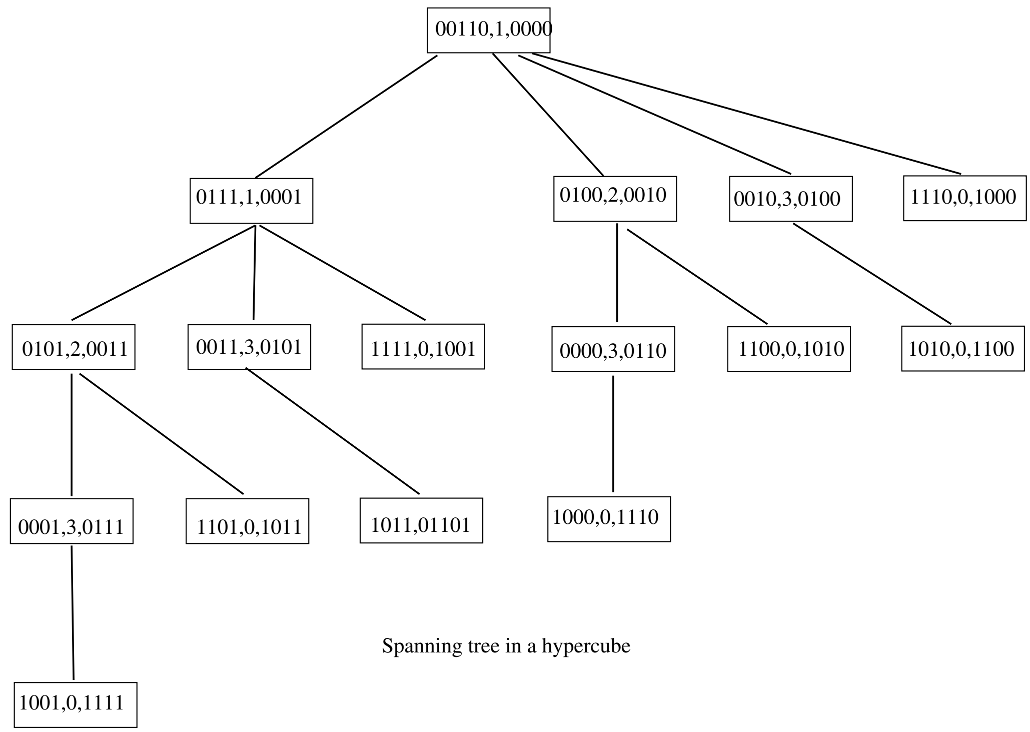


Fig. 8. Example of a broadcast from Node (011,0). Broadcasting within rings are not shown.

more than what it should have in $(n+1)$-dimensional HCRNet. However, while adding the next node (i.e., the $(2{\cdot}2^n+1)$th new node) to Ring 0, a *new ring*,[8] Ring 2^n, is created and Node $(n+1)$ of Ring 0 is moved to Ring 2^n and is labeled as Node $(2^n,n)$. The new node is also added to the new ring and is labeled as Node $(2^n,0)$. Node $(2^n,n)$ is then connected to Node (0,0) (i.e., Node $(2^n \oplus 2^n, 0)$) and Node $(0,n)$ is connected to Node $(2^n,0)$ (Fig. 9). The next new ring is created as Ring $2^n \oplus 2^0$. Now, since Node $(2^n \oplus 2^0, 1)$ is not yet available, Node $(2^n,0)$ transmits to Node $(2^n \oplus 2^0, n)$ instead. Similarly, Node $(2^n \oplus 2^0, 0)$ transmits to Node $(2^n,n)$. (Note that, since at least two nodes are required to form a ring, a new ring is created only when three additional nodes are available at an existing ring.) This process is continued, until 2^n new rings are created, each with two nodes labeled $(x,0)$ and (x,n). The next 2^n nodes are added as Node 1 in the new rings, next 2^n nodes as Node 2, and so on (Fig. 10). The last 2^n nodes, added as Node (x,n), completes the construction of $(n+1)$-dimensional HCRNet. An algorithmic description of the scaling-up procedure is given in Table 3.

Note that the number of transmitters and receivers to be retuned per additional node is three in Phase 1 and 2, four in Phase 3 and 3 Phase 4 (Table 3, Fig. 11). For example, when Node (x,n) is placed between Nodes $(x,0)$ and $(x,n-1)$, only Node $(x,0)$ needs to retune its transmitters and receiver to receive from the new node instead of from Node $(x,n-1)$. The new node also needs to tune its transceivers to communicate with Node $(x,n-1)$ on the wavelengths on which Node $(x,n-1)$ was communicating with Node $(x,0)$. However, tuning at a new node is not counted as a retuning operation. Thus, adding a new node to HCRNet has a very small effect on the existing links in the network.

[8] An *old* ring refers to one of the 2^n rings in n-dimensional HCRNet. A new ring refers to one of the additional 2^n rings in $(n+1)$-dimensional HCRNet.

6. Fault-tolerant routing in incomplete HCRNet

An HCRNet, constructed according to the scaling procedure described in the previous section, and which has at least one more node than an n-dimensional HCRNet but has less nodes than an $(n+1)$-dimensional HCRNet is an *incomplete* $(n+1)$-dimensional HCRNet. A HCRNet, complete or incomplete, with faulty links or nodes is called an *injured* HCRNet. In an incomplete HCRNet, the connectivity pattern is deterministic, and it can be easily established if the number of additional nodes, added as per the scaling rule, is known. On the other hand, the connectivity pattern in an injured HCRNet is non-deterministic since any of its links and/or nodes can be flawed. In the following subsection, we describe a simple yet efficient routing scheme for incomplete HCRNet. This scheme can then be modified, via a depth-first search approach, for routing in a

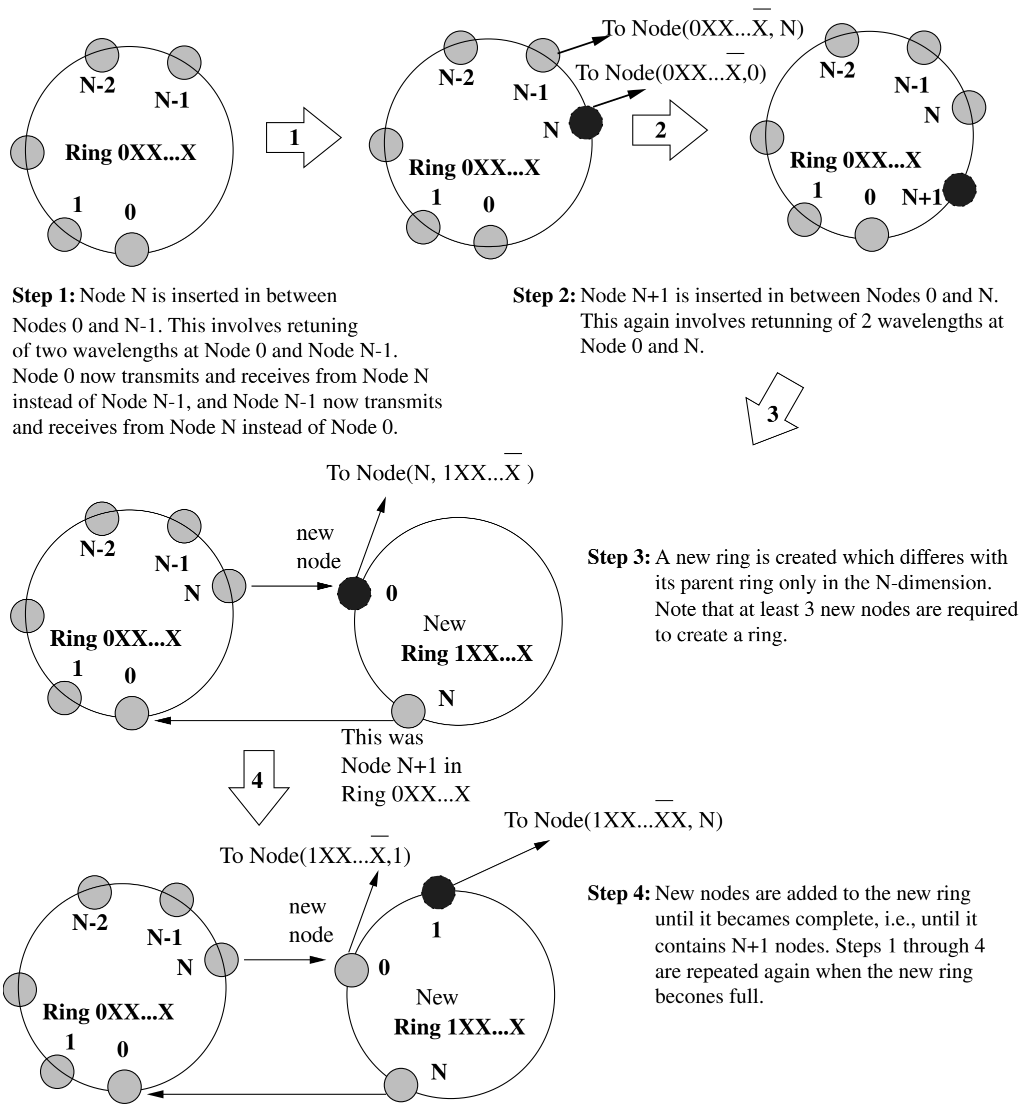


Fig. 9. Steps in adding nodes, one at a time, to an exiting ring.

connected HCRNet with arbitrary number of faulty components [16].

6.1. Routing in incomplete HCRNet

An important property of HCRNet is its hierarchical abstraction. For example, an $(n+1)$-dimensional HCRNet can be viewed as two n-dimensional HCRNets connected only via links in dimension n. Note that, according to our scaling procedure, a ring in an incomplete HCRNet will have an outgoing link in dimension k, only if it has an outgoing link in dimension $(k-1)$. This is because, in a new Ring x new nodes are labeled in increasing order. Thus, if the last node added to a ring is labeled as k, then *all* the rings must have nodes labeled as $k-1$. This implies that there is a hypercube-link in Ring x in dimension $k-1$ from Node $(x, k-1)$ to Node $(x \oplus 2^{k-1}, k)$ if Node $(x \oplus 2^{k-1}, k)$ is already added to the network; otherwise the link in $(k-1)$-dimension from Node $(x, k-1)$ is connected to Node $(x \oplus 2^{k-1}, n)$. Also, if any of the dimension remaining to be corrected in order to reach the destination ring is not available at the source or at an intermediate Ring x, then the packet has to be forwarded to the complete n-dimensional HCRNet via the link in dimension n from Node (x,n). Now, if the source and the destination rings do not differ in dimension n then the packet has to be routed once more through dimension n. In such cases, this can be achieved by simply adding n to the list of dimensions to be corrected. From an incomplete Ring[9] x since only Node (x,n) connects to a ring of the complete n-dimensional HCRNet, once a packet reaches this node it has to determine whether the

[9] In an *incomplete ring* in an incomplete $(n+1)$-dimensional HCRNet at least one link in one of the $n+1$ dimensions is missing.

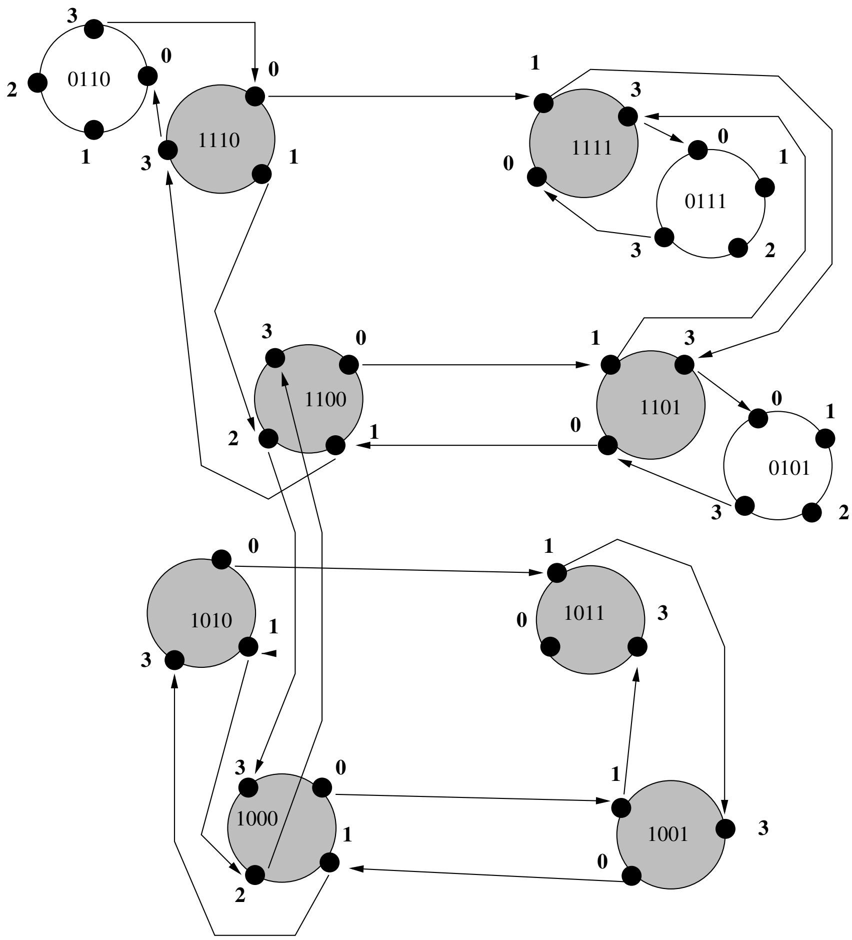


Fig. 10. Incomplete four-dimensional HCRNet (in Phase 4, $i = 4$, see Table 3), shaded rings are the new rings. Connections to parent rings are shown for Rings 1110, 1111, and 1101. Similar connections to parent rings for other new rings are NOT shown. Also, connections among the old rings are NOT shown.

packet needs to be routed out of the present ring to the complete n-dimensional HCRNet.

In an incomplete HCRNet a packet from Node (s_h,s_r) to Node (d_h,d_r) is routed as follows. First the source node decides, in constant time, if all the dimensions in which the source and the destination rings differ are available in the source ring. If so, the packet is forwarded onto the clockwise direction until it reaches a node transmitting in one of the target dimension (say dimension a); otherwise the packet it sent to Node (s_h,n) via the shortest path. In the first case, a is removed from the set of target dimensions and the packet is simply routed onto dimension a. In the later case, Node (s_h,n) determines, in constant time, if the source and destination rings differ in dimension n and if all the remaining target dimensions are available in the present ring. If a target dimension is missing and if the source and the destination ring differ in dimension n then the packet is simply routed to Node $(s_h \oplus 2^n, 0)$ in the parent[10] ring in the complete n-dimensional HCRNet. However, if a target dimension is missing and if the source and the destination ring to *not* differ in dimension n, then the packet is still routed to Node $(s_h \oplus 2^n, 0)$ and *dimension n is added to the list of target dimensions*. Once a packet reaches a node in the complete n-dimensional HCRNet, it is routed according to the simple routing procedure of a complete HCRNet. On the

Table 4
Efficiency of the routing algorithm in incomplete six-dimensional HCRNet

Phase	Number of nodes sf1	Average hop-distance		Efficiency of simple routing (%)
		Shortest path route	Simple routing	
0	160	4.831	5.231	92.36
1	192	5.495	5.984	91.82
2	224	5.731	6.410	89.41
3	256	6.131	7.034	87.16
4	288	6.226	7.293	85.37
4	320	6.392	7.408	86.29
4	352	6.260	6.939	90.21
4	384	6.036	6.516	92.65

[10] Parent of Ring $00\cdots01a_{i-1}a_{i-2}\cdots a_1a_0$ is the Ring $00\cdots00a_{i-1}a_{i-2}\cdots a_1a_0$.

Let (s_h, s_r) be the source node and let (d_h, d_r) be the destination node.
Let $C_h = d_h \oplus s_h$.
Set present_ring, $r = s_h$; Set present_dimension, $e = s_r$. Set present_node=(r, e)

/* see if $\exists$ a dimension not available in the present ring */
Let (r, n_{-1}) denote the node immediately downstream of Node (r, n) in clockwise direction
/* since Node (r, n) is directly communicating with this node it can easily keep
track of n_{-1}. Note that n_{-1} is not equal to $n-1$ in an incomplete ring
consisting of less than $n+1$ nodes */
Then, dimensions $n_{-1}-1, n_{-1}, n_{-1}+1, \ldots, n-1$ are assumed unavailable in Ring r
/* Note that, according to scaling rules, dimensions $0, 1, \ldots, (n_{-1}-2)$ are
available in Ring r */
Let U be an $n+1$ bit binary number with bits $n_{-1}-1, n_{-1}, n_{-1}+1, \ldots, n-1$ set to 1 and the
remaining bits set to 0

do

if $(C_h \wedge U) = 0$ /* i.e., all the required dimensions are available */
If $((C_h \wedge 2^e) \neq 0)$
Set $C_h = C_h \oplus 2^e$
Set $r = r \oplus 2^e$
Forward the packet on dimension e
Set $e = (e+1)$ if $(n_{-1} < e < n)$ set $e = n$;
else
Set $e = (e+1)$; if $(n_{-1} < e < n)$ set $e = n$;
forward the packet in clockwise direction

else /* i.e., all the dimensions in C_h are not available in Ring r */
forward the packet to Node (r, n) via the shortest path (by setting $C_h = 0$)
/* i.e., either in clockwise or in anticlockwise direction */
When Node (r, n) is reached
Recompute $C_h = d_h \oplus s_h$
Set $C_h = C_h \oplus 2^n$;
Set $r = r \oplus 2^n$
Forward the packet on dimension n
/* i.e., packet sent for routing in complete n-dimensional HCRNet */
Set $U = 0$; /* i.e., all the dimensions are available */
Set $e = 0$

until $(C_h = 0)$

Once the destination Ring, d_h, is reached, the packet is routed to the destination node via the shorter of
clockwise and anticlockwise paths.

Fig. 11. Routing rules in an incomplete $(n+1)$-dimensional HCRNet.

other hand, if all of the target dimensions are available in the present ring then the packet is forwarded onto the clockwise direction. Not that, although a slightly more efficient routing is possible we preferred to keep our scheme simple, *fast*, and easily implementable. It should also be noted that the routing scheme, formally described in Fig. 11, does not use any global information. The performance of this routing method is compared against shortest path routing in Table 4.

6.2. Fault-tolerant routing in injured HCRNet

In this section we will consider routing in an *injured* HCRNet in which one or more of the nodes and/or links are faulty. The routing scheme is required to detect such faults in the network and it should automatically reroute the information via another fault-free route if one exists. Fault-tolerant routing in HCRNet is based on the depth-first search approach proposed for hypercubes in Ref. [16]. In order to conserve space we point how the scheme in Ref. [16] can be modified for HCRNet and we refer the reader to Ref. [16] for the details.

In HCRNet, the failure of the ring-links can be dealt by simply sending the message back in the direction from which it came. Let the message enter the ring at node (X,k) and reach Node (X,q) where the next ring-link has failed. Obviously, the route string does not contain any of the dimensions from k through q. However, as the message travels towards the 'other end' of the ring there might be some desired dimension from $(q+1)_{[n]}$ to $k-1$, in which case the message will continue on an optimal path. Otherwise, a detour has to be taken from the node which does not appear in the path list (see Ref. [16]). Note that the failure of a node (X,k) in HCRNet is equivalent to the failure of the dimension k link from Node X in the corresponding hypercube. Also, the failure of a hypercube link in HCRNet is equivalent to the failure of the corresponding link in the hypercube.

7. Conclusions

A new regular multihop architecture for logical lightwave network topology is studied. The proposed architecture is based on a hypercube connected ring structure that not only enjoys the rich and widely studied topological properties of hypercube, but it also overcomes one of its drawbacks. In a hypercube, the nodal degree increases with the number of nodes, and hence the per-node cost of the network increases as the network size grows. However, in a hypercube connected ring network, the nodal degree is small and it remains constant, independent of the network population. Moreover, the proposed architecture, HCRNet, satisfies the basic requirements of a multihop network topology. HCRNet is *scalable*, *modular*, and, it is *perfectly symmetric*. It provides a simple and *fast* routing mechanism that leads to balanced loads on the hypercube-links under uniform traffic. Its average internodal distance is in the logarithmic order of the network population and it is comparable to other regular structures such as the Trous and ShuffleNet. HCRNet resembles the Cube Connected Cycle interconnection pattern proposed for multiprocessor architectures. However, HCRNet improves on CCC by rearranging its hypercube links, which results in a significantly lower average internodal distance. In this work, the structural properties of HCRNet, its scalability, and fast routing schemes in complete and incomplete HCRNet are studied.

Future work will include the study of fault-tolerant routing schemes that can be based on depth-first search or other techniques that are proposed for a hypercube. Also, it will be interesting to extend this work to other forms of product graphs, e.g. generalized hypercube interconnecting generalized multi-connected rings.

References

[1] C.A. Brackett, Dense wavelength division multiplexing networks: principles and applications, IEEE JASC 8 (6) (1990) 948–964.

[2] M.G. Hluchyj, M.K. Karol, ShuffleNet: an application of generalized perfect shuffles to multihop lightwave network, IEEE Infocom '88, 1988, pp. 379–390.

[3] N.F. Maxemchuck, Regular mesh topologies in local and metropolitan area networks, AT&T Tech. J. 64 (1985) 1659–1686.

[4] N.F. Maxemchuck, Routing in the Manhattan Street network, IEEE Trans. Commun. COM-35 (1987) 503–512.

[5] K. Sivarajan, R. Ramaswami, Multihop lightwave networks based on de Bruijn graphs, IEEE Infocom '91, Bal Harbor, FL, April 1991, pp. 1001–1011.

[6] S. Banerjee, J. Iness, B. Mukherjee, New modular architectures for regular multihop lightwave networks, Proc. Conf. on Optical Fiber Communications/IOOC '93, San Jose, CA, February 1993, pp. 147–148.

[7] J. Iness, S. Banerjee, B. Mukherjee, GEMNET: a new modular architectures for regular multihop lightwave networks, IEEE/ACM Trans. Networking 3 (4) (1995) 470–476.

[8] L.N. Bhuyan, D.P. Agrawal, Generalized hypercube and hypercube structures for a computer network, IEEE Trans. Comput. C-33 (4) (1984) 323–333.

[9] M. Gerla, L. Fratta, Tree structured fiber optics MANs, IEEE J. Selected Areas Commun. 6 (6) (1988) 934–943.

[10] Y. Saad, M.H. Schultz, Topological properties of hypercubes, IEEE Trans. Comput. C-37 (1988) 867–872.

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

[12] A.M. Schwartz, M.C. Loui, Dictionary machines on cube-class networks, IEEE Trans. Comput. C-36 (1) (1987) 100–105.

[13] B. Mukherjee, WDM-based local lightwave networks — Part II: multihop systems, IEEE Network 6 (4) (1992) 20–32.

[14] R. Ramaswamy, Multiwavelength lightwave networks for computer communication, IEEE Commun. 31 (2) (1993) 78–88.

[15] B. Li, A. Ganz, Virtual topological for WDM star LANs — the regular structures approach, IEEE Infocom '92, Florence, Italy, May 1992, pp. 1–8.

[16] M.-S. Chen, K.G. Shin, Depth first search approach for fault tolerant routing in hypercube multicomputers, IEEE Trans. Parallel Distributed Systems 1 (2) (1990) 152–159.

[17] S.L. Johnson, C.T. Ho, Optimum broadcasting and personalized communications in hypercubes, IEEE Trans. Comput. C-38 (9) (1989) 1249–1268.

[18] P. Ramanathan, K.G. Shin, Reliable broadcast in hypercube computers, IEEE Trans. Comput. C-37 (12) (1988) 1654–1656.

[19] P.T. Gaughan, S. Yalamanchilli, Adaptive routing protocols for hypercube interconnection networks, IEEE Computer 26 (5) (1993) 12–24.

[20] H.P. Katseff, Incomplete hypercubes, IEEE Trans. Comput. 37 (5) (1988) 604–608.

[21] C. Rose, Mean internodal distance in regular and random multihop networks, IEEE Trans. Commun. 40 (8) (1992) 1310–1318.

[22] D. Banerjee, B. Mukherjee, R. Suryanarayan, The multidimensional torus: analysis of average hop distance and application as multihop lightwave network, Proc. ICC '94, vol. 3, 1994, pp. 1675–1680.

[23] C.S. Raghavendra, J.A. Silvester, A survey of multi-connected loop topologies for local computer networks, Comput. Networks ISDN Systems 11 (1) (1986) 29–42.

[24] C.S. Raghavendra, M. Gerla, Optimum loop topologies for distributed systems, Proc. Seventh Data Communications Symp., Mexico City, October 1981, pp. 218–223.

[25] S. Banerjee, B. Mukherjee, Algorithms for optimized node arrangements in ShuffleNet based multihop lightwave networks, Proc. IEEE Infocom '93, San Francisco, CA, March–April 1993, pp. 557–564.

[26] B. Mukherjee, Optical Communication Networks, McGraw-Hill, New York, 1997.

[27] S. Banerjee, V. Jain, S. Shah, Regular multihop logical topologies for lightwave networks, IEEE Commun. Surveys 2 (1) (1999). http://www.cosmoc.org/pubs/surveys/banerjee.html

[28] J. Yoo, S. Banerjee, Design, analysis, and implementation of wavelength-routed all-optical networks: routing and wavelength assignment approach, IEEE Communications Surveys, Broadband Networks Area, 1997. http://www.comsoc.org/pubs/surveys/yoo.html