CHAPTER-V

Studies on Load Balanced Hybrid Parallel Interconnection Network Topology
5.0 Introduction

Parallel processing has assumed a crucial role in the field of supercomputing. It has overcome the various technological barriers and achieved high levels of performance. A great deal of research in parallel and distributed processing has dealt with methods of interconnecting processors or nodes to achieve goals such as low latency, high bandwidth, efficient communication, fault tolerance and reliability. All these goals amount to high performance of the system which is responsible for pushing the development of the powerful interconnecting networks [43]. Most of the performance gains are due to the architectural rather than the technological improvements [62, 72]. The advance research has resulted in modified, augmented, generalized, hierarchical and/or hybrid networks [49, 173, 178]. As already discussed in the previous chapters, in case of augmented networks the performance gain is either due to addition of extra edges or nodes. With addition of extra edges called complementary edges message latency is reduced resulting in efficient communication. Next with addition to nodes called the network controllers, the processor utilization is improved as the NC's are solely responsible for communication.

In recent days, the researchers have started devoting their attention to development of hybrid and hierarchical networks. The hybrid networks also called as product networks allow to synthesize networks of virtually any desired size [49, 169]. Motivations for designing hierarchical interconnection networks include obtaining greater modularity, lower cost, finer scalability and better fault tolerance [173].

With increased scalability, the probability of system failure increases. Irrespective of the network type with increase in number of processors, the system reliability is also expected to decrease. It is expected that the system may suffer from node or link failure and as a result there may be a need for reconfiguration with least recovery time and cost. Initially in order to make the system fault tolerant spare
processors or links were introduced. But this technique was not effective because the cost of communication in reconfiguration process could be high due to large number of message communication. Again with adding spare nodes the original topological structure of the system gets changed. Another alternate approach is to keep some processors in the system reserved and they can be helpful in case of failures. They can replace the failed nodes as and when necessary. But the cube based networks cannot support this type of reconfiguration especially in case of multiple faults [90]. Thus the fault tolerance aspects need to be very effectively addressed. For this reason alternate fault tolerant features need to be introduced in the network.

In such situations, the load balanced graphs offer a better solution. The load balancing problem is a very important area for research with vast area of application. In load balanced graphs, each active node has a passive counterpart and they have same adjacent nodes. Next, a task in load balanced graph is scheduled in such a manner that it is first executed by the active processor and if that node fails then the passive counterpart takes on the job. This results in faster recovery. The Balanced hypercube a variant of the hypercube is a load balanced graph which supports consistently recoverable embedding. The fault tolerance aspect of the Balanced hypercube (BH) is proved to be better than Hypercube [91,157,204]. Each processor in BH has a backup processor that is having the same set of neighbouring nodes. The Balanced hypercube is beneficial for parallel processing in terms of reduced diameter only when the dimension is odd.

However, the performance parameters such as reliability, fault tolerance, cost effectiveness and the time-cost effectiveness are some of the important aspects that need to be addressed while designing any large scale parallel system [16,197,212,213,214]. For this reason there has always been raising demands for design of a versatile interconnection network with load balancing, better reliability, improved fault tolerance and reduced cost.

In the present chapter, considerable effort has been made to meet the above demands and propose a new network topology called the Balanced Varietal Hypercube (BVH). The proposed topology is a load balanced structure. The BVH is built on the basic structure of the Varietal hypercube. It inherits the merits of fault tolerance from the Balanced hypercube (BH). In addition, the BVH has got a reduced diameter and optimal average distance with less cost.
The remainder of this chapter is organised as follows. Section 5.1 presents the background. The proposed architecture and its details are presented in section 5.2. Topological properties of the proposed Balanced Varietal hypercube are presented and discussed in Section 5.3. The BVH is proved to be a load balanced graph in Section 5.4. The routing and broadcasting aspects are discussed in Section 5.5. The performance comparison is carried out in Section 5.6. The Section 5.7 presents the results obtained and their related discussions. At last the Section 5.8 concludes the Chapter.

5.1 Background

This section describes the topological features of the Varietal hypercube and the Balanced hypercube. The above said interconnection network topologies are described using graph theoretical terminologies and notations [46].

5.1.1 Notation

- \( D(G) \): Diameter of network \( G \)
- \( \bar{d} \): Average inter node distance
- \( d_g \): Node degree
- \( E \): Set of edges
- \( R_p \): Reliability of processor
- \( R_l \): Reliability of link
- \( TR \) or \( R(t) \): Terminal reliability
- \( V \): Set of nodes
- \( \eta \): Message traffic density
- \( \lambda_l \): Link failure rate
- \( \lambda_p \): Processor failure rate
- \( \xi \): Cost of the network
- \( \rho \): Ratio of Link cost to processor cost
- \( \sigma \): The ratio of cost of penalty to cost of processor

5.1.2 Balanced Hypercube

The Balanced hypercube proposed by Huang [90], forms a hypercube variant that gives better performance with the same number of edges and vertices. The Balanced hypercubes are superior to hypercube in terms of smaller diameter, support for efficient
reconfiguration without changing the adjacency relationship among tasks. [157]. It's various properties are studied in [91,243,245].

The Balanced Hypercube network (B\textsuperscript{H}_n) of dimension n is a load balanced graph having 2\textsuperscript{2n} nodes. Each vertex of B\textsuperscript{H}_n has a unique n-component vector on \{0, 1, 2, and 3\} for its label such as \((a_0, a_1, ..., a_{n-1})\). A vertex u having label \((a_0a_1...a_{n-1})\) is adjacent to the following 2n vertices for 1 ≤ i ≤ n - 1,

- \(((a_0 + 1)\text{mod } 4, a_1, ..., a_{i-1}, a_i, a_{i+1}, ... a_{n-1})\),
- \(((a_0 - 1)\text{mod } 4, a_1, ..., a_{i-1}, a_i, a_{i+1}, ... a_{n-1})\),
- \(((a_0 + 1)\text{mod } 4, a_1, ..., a_{i-1}, a_i + (-1)^{a_0}\text{mod } 4, a_{i+1}, ... a_{n-1})\) and
- \(((a_0 - 1)\text{mod } 4, a_1, ..., a_{i-1}, a_i + (-1)^{a_0}\text{mod } 4, a_{i+1}, ... a_{n-1})\).

The Balanced Hypercube of dimension 3 is shown in Fig. 5.1. The B\textsuperscript{H}_n can be constructed from four copies of B\textsuperscript{H}_{n-1} by adding a new edge in the nth dimension of every vertex in B\textsuperscript{H}_{n-1}. B\textsuperscript{H}_1 is a cycle with vertex set \{0,1,2,3\}.

![Figure 5.1: Balanced Hypercube of dimension 3](image)

5.1.2 Varietal Hypercube

The Varietal Hypercube is a variation of Hypercube with reduced diameter and average distance [40]. An n-dimensional Varietal Hypercube (VQ\textsuperscript{n}) is constructed from two numbers of (n-1) dimensional Varietal Hypercubes in a way similar to that of the
Hypercube with some modifications in connections. The connections are as follows: $VQ_1$ is a complete graph of two vertices with address 0 and 1.

For $n > 1$, $VQ_n$ is constructed from $VQ_{n-1}^0$ and $VQ_{n-1}^1$ according to the rule: a vertex $u$ with node address $(0, u_{n-1} u_{n-2} u_{n-3} \ldots u_1)$ from $VQ_{n-1}^0$ and a vertex $v$ with node address $(1, v_{n-1} v_{n-2} v_{n-3} \ldots v_1)$ from $VQ_{n-1}^1$ are adjacent in $VQ_n$ if and only if

1) $u_{n-1} u_{n-2} u_{n-3} \ldots u_1 = v_{n-1} v_{n-2} v_{n-3} \ldots v_1$, if $n = 3k$ or

2) $u_{n-3} \ldots u_1 = v_{n-3} \ldots v_1$ and $(u_{n-1} u_{n-2} , v_{n-1} v_{n-2}) \in \{(00,00),(01,01),(10,11),(11,10)\}$, if $n = 3k$. The Varietal Hypercube of dimension 3 is shown in Fig. 5.2.

![Figure 5.2: Varietal Hypercube of dimension 3 (VQ3)](image)

5.2 Proposed Topology: Balanced Varietal Hypercube (BVH)

The present section is devoted towards providing the topological details of the proposed topology Balanced varietal hypercube (BVH).

5.2.1 Construction

Let $G = \{V, E\}$ be a finite, undirected graph with set of nodes $V$ and set of edges $E$. A node in $V$ represents a processor and an edge in $E$ represents a communication link between two processors. If an edge $e = (u, v) \in E$, then the nodes $u$ and $v$ are adjacent. For each node $v$ there exists another node $v'$ such that $v$ and $v'$ have same adjacent nodes. The pair $v$ and $v'$ are called matching pair. A task can be scheduled to both $v$
and \( v' \) in such a way that one copy is active and the other one is passive. If node \( v \) fails, its task can simply be shifted to node \( v' \) by activating copies of these tasks in \( v' \). All the other tasks running on other nodes need not be reassigned to keep the adjacency property, that is two tasks those are adjacent are still adjacent after the reconfiguration. It is possible to have an active task running on node \( v \) with its backup in \( v' \), while having another active task on \( v' \) and its backup on node \( v \). The degree \( d(v) \) of node \( v \) is equal to the number of edges in \( G \) which are incident on \( v \). The diameter of \( G \) is the maximum distance between two nodes in \( G \) over all pairs of nodes. The Balanced varietal hypercube of different dimensions are shown in Fig. 5.3.

An \( n \)-dimensional Balanced Varietal Hypercube (BVH\(_n\)) consists of \( 2^{2n} \) nodes each of which is represented by the address \((a_0, a_1, a_2, ..., a_{n-1})\) where \( a_i \in \{0,1,2,3\} \) and \( 0 \leq i \leq n - 1 \). Every node \((a_0, a_1, a_2, ..., a_{n-1})\) connects the following \( 2n \) nodes, which are divided into two categories: (i) inner nodes and (ii) outer nodes. In an \( n \) dimensional Balanced Varietal hypercube BVH\(_n\) each unit is connected to others through hyperlinks.

![Diagram of Balanced Varietal Hypercube](image)

(i) Inner nodes:

Case I: When \( a_0 \) is even,

(i) \( <(a_0+1)mod\ 4, a_1, a_2, ..., a_{n-1}> \)

(ii) \( <(a_0-2)mod\ 4, a_1, a_2, a_3, ..., a_{n-1}> \)

Case II: When \( a_0 \) is odd,

(i) \( <(a_0-1)mod\ 4, a_1, a_2, ..., a_{n-1}> \)
(ii) \( (a_0+2) \mod 4, a_1, a_2, \ldots, a_{n-1} \) 

(b) dimension 3, BVH_3

Figure 5.3: Balanced Varietal Hypercube of dimension (a) dimension 2, BVH_2
(b) dimension 3, BVH_3

(ii) **Outer nodes:**

Case I: When \( a_0 = 0,3 \);

(i) For \( 'a_l' = 0 \)
\[ <(a_0+1) \mod 4, a_1, \ldots, (a_l+1) \mod 4, a_3, \ldots, a_{n-1} > \]
\[ <(a_l-1) \mod 4, a_1, \ldots, (a_l+1) \mod 4, a_3, \ldots, a_{n-1} > \]

(ii) For \( 'a_l' = 3 \)
\[ <(a_0+1) \mod 4, a_1, \ldots, (a_l-1) \mod 4, \ldots, a_{n-1} > \]
Chapter V Some Studies on Parallel Computer Interconnection Network Topology

\[ <(a_0 - 1) \mod 4, a_1, \ldots, (a_{i-1}) \mod 4, \ldots, a_n, > \]

Case II: When \( a_0 = 1, 2 \) and \( a_i = 0, 3 \)
\[ <(a_0 + 1) \mod 4, a_1, \ldots, (a_{i+2}) \mod 4, \ldots, a_n, > \]
\[ <(a_0 - 1) \mod 4, a_1, \ldots, (a_{i+2}) \mod 4, \ldots, a_n, > \]

Case III: When \( a_0 = 0, 1 \)
(i) For \( a_i = 1 \)
\[ <(a_0 + 1) \mod 4, a_1, \ldots, (a_{i+2}) \mod 4, \ldots, a_n, > \]
\[ <(a_0 - 1) \mod 4, a_1, \ldots, (a_{i+2}) \mod 4, \ldots, a_n, > \]
(ii) For \( a_i = 2 \)
\[ <(a_0 + 1) \mod 4, a_1, \ldots, (a_{i+2}) \mod 4, \ldots, a_n, > \]
\[ <(a_0 - 1) \mod 4, a_1, \ldots, (a_{i+2}) \mod 4, \ldots, a_n, > \]

Case IV: When \( a_0 = 2, 3 \)
(i) For \( a_i = 1 \)
\[ <(a_0 + 1) \mod 4, a_1, \ldots, (a_{i+1}) \mod 4, \ldots, a_n, > \]
\[ <(a_0 - 1) \mod 4, a_1, \ldots, (a_{i+1}) \mod 4, \ldots, a_n, > \]
(ii) For \( a_i = 2 \)
\[ <(a_0 + 1) \mod 4, a_1, \ldots, (a_{i+1}) \mod 4, \ldots, a_n, > \]
\[ <(a_0 - 1) \mod 4, a_1, \ldots, (a_{i+1}) \mod 4, \ldots, a_n, > \]

In the next section the details of the topological properties of the proposed BVH network are discussed.

5.3 Topological Properties of BVH

For a multicomputer system, parameters such as diameter, average internode distance and message traffic density are crucial and determine the performance of the network. This section proceeds to evaluate such parameters.

**Degree:** The degree of a node in a graph is defined as the total number of edges connected to that node. Similarly the degree of a network is defined as the largest degree of all the vertices in its graph representation.

**Theorem 5.1:** The degree of any node in the Balanced varietal hypercube of dimension \( n \) is given by
\[ d_g = 2n. \] (5.1)
Proof: From the Definition 3.1 it is clear that BVH₁ is constructed from four nodes and the number of edges connected to each node is 2. A Balanced varietal hypercube of any dimension BVHₙ is constructed from four BVHₙ-1s with each node having two extra connections as shown in Fig.5.3 and 5.4. So, when the dimension is increased by one, the number of extra connections made to each node is increased by 2. Hence, the theorem is proved.

Number of Nodes:

In a finite undirected graph \( G = (V, E) \), \( V \) represents the node set and \( E \) represents the edge set. Normally a node in \( V \) represents a processor and an edge in \( E \) corresponds to a communication link connecting two numbers of processors.

**Theorem 5.2:** The total number of nodes in an \( n \)-dimensional Balanced varietal hypercube is given by

\[
p = 2^{2^n}
\]

(5.2)

Proof: From the construction of Balanced varietal hypercube it is clear that, for every node in BVH there exist another node such that these two nodes are having same adjacent nodes. Hence an \( n \)-dimensional Balanced varietal hypercube is very much similar to a varietal hypercube of dimension \( 2n \), and the number of nodes is same as that of \( n \)-cube [252].

**Lemma 5.1:** A graph \( G = (V, E) \) is an \( n \)-cube if and only if

a) \( V \) has \( 2^n \) vertices.

b) Every vertex has degree \( n \).

c) \( G \) is connected.

d) Any two adjacent nodes A and B are such that the nodes adjacent to A and those adjacent to B are linked in a one-to-one fashion.

For a one dimensional Balanced varietal hypercube, the number of nodes is equal to \( 2^{2^1} = 4 \) nodes. For a two dimensional Balanced varietal hypercube shown in Fig. 5.3, the total number of nodes are equal to \( 2^{2^2} = 16 \). Similarly, for a three dimensional balanced varietal hypercube the total number of nodes is equal to \( 2^{2^3} = 64 \).

Hence, by induction it can be proved that the \( n \)-dimensional BVHₙ has \( 2^{2^n} \) nodes.

Number of Edges: An edge represents a communication link between two processors in a network. If an edge \( e = (u,v) \in E \), then the nodes \( u \) and \( v \) are adjacent.
Chapter V

Some Studies on Parallel Computer Interconnection Network Topology

Theorem 5.3: In Balanced varietal hypercube, the total number of edges is given by

\[ E = n \times 2^{2n} \]  

(5.3)

Proof: From Theorem 5.2, an n-dimensional Balanced varietal hypercube has \(2^{2n}\) nodes. According to Theorem 3.1 the degree of any node in an n-dimensional BVH is 2n. But a link is shared by two nodes as shown in Fig. 5.3. Therefore the total number of links or edges for BVH\(_n\) is 

\[ E = 2n \times 2^{2n}/2 = n2^{2n}. \]

Diameter: The diameter is considered to be the most important parameter of any network. The distance \(d(u,v)\) between two distinct vertices is the length of the shortest path between these vertices. The diameter of \(G\), denoted as \(D(G)\) is defined to be the maximum of these distances. Since the diameter is the worst case distance in a graph, it reflects how long it would take for a node to broadcast message to all other nodes.

Theorem 5.4: The diameter of an n-dimensional Balanced varietal hypercube is

\[ D(G) = 2n \text{ for } n=1 \text{ and } \left\lceil n + \frac{n}{2} \right\rceil \text{ for } n > 1. \]  

(5.4)

Proof: The method of mathematical induction on \(n \geq 1\) is used to prove the theorem. The theorem is true for \(n=1\). For \(n=1\), BVH is a cycle consisting of nodes \(\{0,1,3,2\}\). So, it is clear that the diameter of BVH\(_1\) is 2.

For \(n=2\), as shown in Fig. 5.3, the maximum of the shortest distance between two nodes is \(2 + \frac{2}{2} = 3\). The distance of each node is calculated from every other node.

For BVH\(_3\), the distance is \(3 + \frac{3}{2} = 4\).

Let \(u=(a_0,a_1,a_2,...,a_{n-1})\) and \(v=(b_0,b_1,b_2,...,b_{n-1})\) be two nodes in an n-dimensional balanced varietal hypercube. When \(a_{n-1} \neq b_{n-1}\) it can be considered that \(u\) and \(v\) are on two adjacent BVHs of dimension \(n-1\). Hence, the distance between them is \(n + \frac{n}{2}\) as \(n > 1\). Hence the result is true for \(n\).

When \(a_{n-1} = b_{n-1}\), then the nodes are on the same BVH\(_n\). Hence the distance between them is less than \(n + \frac{n}{2}\) and equal to \(\left\lceil \frac{3n-3}{2} \right\rceil\).

Average Distance: In a loosely coupled distributed system, while executing any parallel algorithm message traffic between processors takes on a distribution fairly close to uniform distribution. The average distance conveys the actual performance of the network better in practice. The summation of distance of all nodes from a given node over the total number of nodes determines the average distance of the network [23,96].
Theorem 5.5: In the Balanced varietal hypercube the average distance $\bar{d}(BVH_n)$ is given by,

$$\bar{d}(BVH_n) = \frac{1}{2^n} \sum d[(0,0,0), k] ; \text{all nodes in BVH}_n \quad (5.5)$$

Proof: The total number of nodes in BVH$\_n$ is $2^{2n}$. The average distance is the ratio of sum of distances of all nodes from a given node to the total number of nodes.

**Message Traffic Density:** The performance of a network in handling the message traffic can be analyzed by assuming that each node is sending a message to a node at distance $d$ on the average. An efficient network should have a wide enough bandwidth to handle the resulting traffic so that the message traffic density is the minimum.

Theorem 5.6: The message traffic density for an n-dimensional Balanced varietal hypercube is

$$\eta = \frac{\bar{d}(BVH_n)2^{2n}}{n2^{2n}} \quad (5.6)$$

Proof: As discussed earlier, the message traffic density can be calculated if we know the average distance, the total number of nodes and the total number of edges. From Theorem 5.2, the number of nodes in a BVH of dimension $n$ is $2^{2n}$. From Theorem 5.3, the number of edges in BHV$\_n$ is $n \times 2^{2n}$. Using the average distance of a n-dimensional BVH, $\eta$ can be calculated. Hence Message traffic density is given by

$$\eta = \frac{\text{Avg.Distance} \times \text{No.of nodes}}{\text{No.of links}}$$

$$= \frac{\bar{d}(BVH_n)2^{2n}}{n2^{2n}}$$

**Cost:** The cost is an important factor as far as an interconnection network topology is concerned. The topology which possesses minimum cost is treated as the best candidate. The cost factor of a network is the product of degree and diameter.

Theorem 5.7: The cost of an n-dimensional Balanced varietal hypercube is given by

$$\xi = 2n \times \left[n + \frac{n}{2}\right] \quad (5.7)$$

Proof: The degree of an n-dimensional BVH is 2n. The diameter is $\left[n + \frac{n}{2}\right]$. Since the cost is product of degree and diameter, hence for a BVH$\_n$

Cost = degree $\times$ diameter $= 2n \times \left[n + \frac{n}{2}\right]$ for $n > 1$.

**Node-disjoint Path:** The Node-disjoint path defines in how many ways two nodes can be linked without any common node. The Node-disjoint paths are to be considered quite important while designing an interconnection network.
Theorem 5.8: For any pair of nodes in an n-dimensional Balanced varietal hypercube, there exists $2^n$ node disjoint paths between them.

**Proof:** For one dimensional BVH, the Node-disjoint paths between any two nodes are equal to $2 \times 1 = 2$. For BVH$_1$, considering nodes 0 and 3

Path 1: 0-1-3
Path 2: 0-2-3, as it is a cycle of nodes (0, 1, 3, 2).

In two dimensional BVH, the Node-disjoint paths will be $2 \times 2 = 4$. For example, from node (0,0) and (3,3) the different paths are

Path 1: (0,0)-(1,1)-(3,1)-(1,3)-(3,3)
Path 2: (0,0)-(1,0)-(0,2)-(2,2)-(3,3)
Path 3: (0,0)-(2,0)-(3,2)-(1,2)-(3,3)
Path 4: (0,0)-(2,1)-(0,1)-(2,3)-(3,3)

Similarly from Fig.5.4, the Node-disjoint paths between (0, 0, 0) and (3, 3, 0) are

Path 1: (0,0,0)-(1,1,0)-(3,1,0)-(1,3,0)-(3,3,0)
Path 2: (0,0,0)-(1,0,0)-(0,2,0)-(2,2,0)-(3,3,0)
Path 3: (0,0,0)-(2,0,0)-(3,2,0)-(1,2,0)-(3,3,0)
Path 4: (0,0,0)-(2,1,0)-(0,1,0)-(2,3,0)-(3,3,0)
Path 5: (0,0,0)-(1,0,1)-(0,0,1)-(1,1,1)-(0,1,1)-(3,1)
Path 6: (0,0,0)-(2,0,1)-(3,0,1)-(2,1,1)-(311)-(231)-(3,3,0)

So, there are $2 \times 3 = 6$ different paths for a 3-dimensional BVH. By induction it can be proved that for an n-dimensional BVH there will be $2^n$ paths.

### 5.4 Load Balancing in BVH

A completely connected graph $G$ is said to be load balanced if and only if for every node in $G$ there exist another node matching to it. These two nodes have same set of adjacent nodes. Also the graph should have even number of nodes [204].

Theorem 5.9: The BVH$_n$ is a load balanced graph.

Following Lemmas are proposed in support of the proof of the Theorem 5.9.

**Lemma 5.2:** The BVH contains even number of nodes.

**Proof:** The BVH is constructed from $2^n$-dimensional Varietal hypercubes. Each BVH$_n$ is derived from four copies of BVH$_{n-1}$. Each node of these individual sub modules are
connected by adding two numbers of extra edges. Thus the node degree becomes 2n. Hence, the BVH contains even number of nodes.

**Lemma 5.3:** The BVH\(_n\) is a completely connected graph.

**Proof:** From the construction it is clear that the n-dimensional BHV is a connected graph as it is extended from two (n-1) dimensional varietal hypercube. Thus BVH\(_{n+1}\) is derived from four copies of BVH\(_n\). The nodes of the two sub graphs are connected by adding two extra connections.

If \(v = (a_0, a_1, a_2, ... a_{k-1}, i)\) is a node in BVH\(_{k+l}\) for \(0 < i < 3\), then the above said edges connect to following nodes:

i) \(((a_0+1), a_1, a_2, ... a_{k, i+1})\) and \(((a_0-1), a_1, a_2, ... a_{k-1}, i+1)\) in BVH\(_{k+l+1}\) if \(a_0\) is even and

ii) \(((a_0+1), a_1, a_2, ... a_{k, i-1})\) and \(((a_0-1), a_1, a_2, ... a_{k-1}, i-1)\) in BVH\(_{k+l-1}\) if \(a_0\) is odd.

Hence, BVH is a completely connected graph.

**Lemma 5.4:** The Balanced varietal hypercube is a bipartite graph.

**Proof:** The Balanced varietal hypercube of dimension \(n\) can be partitioned into a set of matching pairs \(v = (a_0, a_1, ... a_{n-1})\) and \(v' = (a_0 + 3, a_1, ... a_{n-1})\) as they have same adjacent node set. Hence, the node set of BVH can be divided into two disjoint subsets. Then any edge must link two nodes from different subsets. In order to divide the node set of the network into two parts the extra connections used to link the sub modules (as discussed in Lemma 5.3) need to be removed. Hence, the BVH is a bipartite graph.

**Lemma 5.5:** The BVH supports consistently recoverable embedding.

**Proof:** In this case embedding of ring network into the BVH is considered with unit dilation and unit congestion. In BVH the basic building block is a varietal hypercube and it is a variant of the Hypercube. Embedding of rings in to HC and VQ's are already reported in Literature [199,216, 40]. Thus, a ring can be easily embedded in BVH. But the objective is to prove in the presence of ‘k’ faults a ring can be successfully embedded in \(k\) reconfiguration steps.

In BVH each node has a matching pair which has same adjacent nodes. If any node \(v = (a_0, a_1, ... a_{n-1})\) fails at any point of time, then it is replaced by its matching
pair \((a_0 + 3, a_1, ... a_{n-1})\) in one step. The embedding process utilizes a mapping 
\((\phi)\) on the node set \(V\), of BVH defined as follows:

\[
\phi(v) = v', \text{ where } v' \in V. \text{ The node } v' \text{ can be any one of the following:}
\]

i) \(\phi(0, 3, 3, ... 0, a_{i+2}, a_{i+3}, ... a_{n-1}) = (1, 3, 3, ... 0, a_{i+2} + 1, a_{i+3}, ... a_{n-1})\),

where \(0 \leq i < n - 1\).

ii) \(\phi(0, a_1, a_2, ... a_{n-1}) = (1, a_1, a_2, ... a_{n-1})\) if node \((0, a_1, a_2, ... a_{n-1})\) is
different from the node in (i).

iii) \(\phi(1, a_1, a_2, ... a_{n-1}) = (0, a_1 - 1, a_2, ... a_{n-1}).\)

Now the set of edges, that connects to the above nodes that is \(v\) and \(\phi(v)\) will
form a cycle with \(2 \times 4^{n-1}\) nodes. If fault occurs at any of these nodes then the corresponding matching node will be inserted in the sequence. Thus, the mapping \(\phi\) provides a consistent recoverable embedding of rings in the Balanced varietal hypercube.

**Proof of Theorem 5.9:** For a network to be load balanced, the network should be
connected, with even number of nodes and bipartite. It has been already proved in
Lemma 5.2 and 5.3, that the BVH network is a connected graph containing even number
of nodes. In Lemma 5.4 it has been proved that the BVH to be a bipartite graph. Next
according to Lemma 5.5, it supports recoverable embedding. Hence, the BVH network
is a load balanced graph.

**Illustration:**

In BVH\(_3\), the total number of nodes is 16 as shown in Fig. 5.3. It is a completely
connected graph. The node set \(V\) of BVH can be partitioned into two disjoint subsets
namely \(V_1= \{00, 10, 02, 12, 03, 13, 01, 11\}\) and \(V_2= \{30, 20, 32, 22, 33, 23, 31, 21\}\). The set
\(V_2\) serves as the spare node set. If fault occurs at any node of set \(V_1\), then matching node
of \(V_2\) will be used as replacement. The initial ring embedding is shown with arrow heads
depicting its path in Fig. 5.4 (a). An eight node ring is embedded into the BVH\(_3\). The
node sequence is \((00, 10, 02, 12, 03, 13, 01, 11, 00)\) as shown in the Figure.

In case fault occurs for example, if fault occurs at node \((1 2)\) then its
spare node \((2 2)\) will be used for replacement. In case of more than one faults, like
suppose the node \((0 1)\) becomes faulty then \((3 1)\) will be used as shown in Fig. 5.4 (b).
The dilation of this embedding is one. The two faults require only two reconfiguration
steps. Thus the BVH network supports recoverable embedding. Hence it is a load balanced graph.

Figure 5.4: Embedding of ring in BVH$_3$ (a) Regular network, (b) Faulty network

In the following section optimal routing and broadcasting algorithms are developed for the proposed Balanced varietal hypercube.

### 5.5 Routing and Broadcasting in BVH

In parallel computer networks, communication is an important issue regarding how the processor can exchange message efficiently and reliably. An optimal routing algorithm aims to find the shortest path between two nodes communicating with each other.

**Routing:**

In the routing process, each processor along the path considers itself as the source and forwards the message to a neighboring node one step closer to the destination. The algorithm consists of a left to right scan of source and destination address. Let $r$ be the right most differing bit (quaternary) position. The numbers to the right of $u_r$ is not to be considered as they lie on the same BVH$_r$. Since the diameter of BVH$_1$ is 2 there is at least one vertex which is a common neighbor of $u_r$ and $v_r$. If $d$ is an element such that $d$ neighbor of $u_r$ is also a neighbor of $v_r$. Then $d$ is chosen such that $u_r=v_r$. Then in the next step $d_{i-1}$ is chosen such that $u_{r-i}=v_{r-i}$. This process continues until $u=v$.

Algorithm : Procedure Route($u, v$)

begin
Chapter V Some Studies on Parallel Computer Interconnection Network Topology

\( r: \) right most differing bit position
\( d: \) choice such that \( d_{u_r} = v_r \)
route to \( d \)-neighbour else
route to \( r \)-neighbour (\( u \) and \( v \) are adjacent)
if (\( u \) and \( v \) are not adjacent) then
\( d_j = \) choice that \( d_{u_r,j} = d_{v_r,j} \)
route to \( d_j \) neighbour
end

This process continues till \( u_0, u_1, u_2, \ldots, u_{r-1}, u_r = v_0, v_1, v_2, \ldots, v_{r-1}, v_r \).

Finally, \( u = v \) that is source = destination.

**Broadcasting:**

Broadcasting is the process of information dissemination in a communication network by which a message originated at a node is transmitted to all other nodes in the network. The broadcast primitive finds wide application in the control of distributed systems and in parallel computing. For instance, in computer networks, there are many tasks, such as scheduling and updating other processors in order to continue the processing.

![Diagram](image)

Figure 5.5: Broadcasting in (a) BVH₁ (b) BVH₂
An optimal one-to-all broadcast algorithm is presented for BVH\(_n\) assuming that concurrent communication through all ports of each processor is possible. It consists of \((n+1)\) steps.

**Lemma 5.6:**
The oriented versions of the trees \(ST_i\) obtained by directing arcs from parent to child for \(i=0, 1, \ldots, d-1\) are pair wise arc disjoint.

**Procedure** \(Broadcast(u, n):\)

**Step 1:** send message to \(2n\) neighbours of \(u\)

**Step 2:** one of \(2n\) nodes sends message to its \(2n-1\) neighbours. Then \(n\) nodes from the rest nodes send message to their \((2n-2)\) neighbours.

**Step 3:** continue step 2 till all the nodes get the message.

**Step 4:** end

The broadcast procedure has been illustrated in Fig. 5.5 (a) and (b) for one dimensional and two dimensional BHV respectively.

### 5.6 Performance Analysis of BVH

In this section various performance parameters are evaluated for the proposed BVH\(_n\) topology.

**Cost Effectiveness Factor:**

The total cost of a multicomputer system comprises of the cost of the processors as well as the cost of the communication links. Usually, the number of links is a function of the number of processors. Thus, the earlier methods of performance evaluation by speedup and efficiency are inadequate. The cost effectiveness factor gives more insight to the performance of parallel systems that uses parallel algorithms [197].

Cost effectiveness of the BVH is a product of two terms, one characterises the architecture and the other corresponds to the efficiency of the algorithm. Therefore, the Cost effectiveness factor, \(CEF(p)\) for the proposed system is the ratio of cost effectiveness \(CE(p)\) to the efficiency \(E(p)\) where \(p\) is the total number of processors in the system. Here the number of links is a function of the number of nodes in the system. The CEF of BVH is given by
Chapter V 
Some Studies on Parallel Computer Interconnection Network Topology

\[ \text{CEF}(p) = \frac{CE(p)}{E(p)} = \frac{1}{1 + \rho g(p)} \]  \hspace{1cm} (5.8)

Where \( \rho = \frac{c_l}{c_p} = \frac{\text{cost of a link}}{\text{cost of a processor}} \) and
\[ g(p) = \frac{f(p)}{p} \]

\( f(p) \) gives the number of links as a function of \( p \), the total number of nodes and \( n \), the diameter of the network.

For \( \text{BVH}_n \), \( p = 2^{2n} \). The total number of links is given by

\[ E = n2^{2n} = f(p). \]

Hence,
\[ g(p) = \frac{f(p)}{p} = \frac{n2^{2n}}{2^{2n}} = n. \]  \hspace{1cm} (5.9)

Now using Eq. (5.9) in (5.8), we can have
\[ \text{CEF}(p) = \frac{1}{1 + \rho n} \]  \hspace{1cm} (5.10)

The CEF enables the comparison of different parallel algorithms in different multicomputer architectures to determine the most cost effective combination of algorithm and architecture.

**Time Cost Effectiveness Factor:**

The consideration of time factor is essential in evaluation of performance of a parallel system. The Time cost effectiveness factor (TCEF) takes into account the time factor in addition to the cost effectiveness factor considered in the above paragraph. It considers the situation where a faster solution to a problem is more rewarding than a slower solution [197].

\[ \text{TCEF}(p, T_p) = \frac{1 + \alpha T_i^{p-1}}{1 + \rho g(p) + \frac{T_i^{p-1}}{p} \alpha} \]  \hspace{1cm} (5.11)

where \( T_i \) is the time required to solve the problem by a single processor using the fastest sequential algorithm, \( T_p \) is the time required to solve the problem by a parallel algorithm using a multicomputer system having \( p \) processors and \( \alpha \) is the ratio of cost of penalty with the cost of processors. For linear time penalty in \( T_p, \alpha \) is chosen as 1.

Substituting the value of \( g(p) \) from Eq. (5.9) in Eq.(5.11) the TCEF for \( \text{BVH}_n \) is given by
\[ \text{TCEF}(p, T_p) = \frac{1 + \alpha}{1 + \rho n + \frac{\alpha}{2^{2n}}} \]  \hspace{1cm} (5.12)
Reliability Analysis:

The assessment of reliability is very important for critical systems like the parallel systems. Reliability is the conditional probability that a system will survive in an interval $(0,At)$, given that it was operational at time $t=0$. The reliability of an electronic component $(R_t)$ of the system is given by

$$R_t = e^{-\lambda t}$$

where $\lambda$ is the failure rate of the component and $t$ is the mission time.

Reliability of a network is dependent on the reliability of its components at the hardware level. It decreases in an exponential manner with time. Hence, the reliability is not only dependent on the topology but also on time. From the topological point of view, reliability issues have been addressed by different researchers in the past [212,213,186]. For simplicity, two terminal reliability or simply terminal reliability is considered here.

The terminal reliability is defined as the reliability between any two specified nodes termed as source and destination. The total numbers of node disjoint paths as well as number of links and nodes involved in a particular path are important for evaluation of its reliability. The reliability analysis has been carried out following a method called sum of disjoint products (SDP) [186,204]. Using the said method the probability of each term is found out separately which is then added together to get the exact two-terminal reliability. For calculating the terminal reliability (TR) between two given nodes of a network, the reliability of each node as well as edge is also considered.

Terminal reliability between a pair of nodes is given by

$$TR = 1 - [(1 - R_t^m R_p^n)^k (1 - R_t^{m'} R_p^{n'})^{k'}]$$

where, $R_t$ = Reliability of each link

$R_p$ = Reliability of each processor (node) where there are $k'$ paths with $m'$ number of links and $n'$ number of processors in each path.

Reliability of BVH$_2$

From the Theorem 5.8, for BVH$_2$, considering node (0,0) as the source node and (3,3) as destination node there are four node disjoint paths. All the four paths include three processors with four links. So for BVH$_2$, using Eq. (5.14), the terminal reliability is given by,

$$TR(BVH_2) = 1 - [(1 - R_t^m R_p^n)^k (1 - R_t^{m'} R_p^{n'})^{k'}]$$

Now substituting $R_t = 0.9$; and $R_p = 0.8$, Eq. (5.13) becomes
TR(BVH$_2$) = 1 - [(1 - 0.9^4 * 0.8^3)^4] = 0.8055

**Reliability of BVH$_3$**

As stated earlier in Theorem 5.8, for BVH$_3$ considering node (000) as the source and (330) as destination there are six parallel paths. Four of them have four links with three processors and the rest two have five links and four processors. So the terminal reliability for BVH$_3$ is given by

\[ TR(BVH_3) = 1 - [(1 - R_f R_p^m)^k(1 - R_f R_p^{m'})^{k'}] \]

\[ = 1 - [(1 - 0.9^4 * 0.8^3)^4(1 - 0.9^5 * 0.8^4)^2] \]

\[ = 0.8247 \]

**Reliability Analysis With Respect To Time**

For the current work the link failure rate is assumed to be 0.0001 failures per hour and processor failure rate is assumed to be 0.001 failures per hour [212]. Then for mission time 1000 Hrs to 5000 Hrs the reliability of BVH$_n$ is evaluated for a fixed value of $n$.

### 5.7 Results and Discussions

All interconnection topologies may not be suitable for each task. Therefore, before selecting a particular topology, it is important to compare its performance with its predecessors. The present section is a systematic attempt to compare the various performance parameters of the proposed BVH with those of VQ, BH and HC. The various performance parameters analyzed below are: degree, diameter, cost, average distance, cost effectiveness, time cost effectiveness and reliability. For comparison the HC and VQ are considered with $2n$ dimension.

The Fig. 5.6 provides a comparative illustration of the diameter of the BVH. The diameter of BVH is observed to lie between that of the Varietal hypercube and the Balanced hypercube. In case of BVH, the diameter is slightly more than that of the Varietal hypercube, however, it is very less than that of Balanced hypercube and the Hypercube thereby reducing worst case delay in communication. The BH$_n$ and HC$_{2n}$ networks are having same values for diameter.

Since the BVH provides a lower diameter with the degree remaining the same as compared to BH and HC, the cost factor of the BVH is much less than that of BH and hypercube.
The Fig. 5.7 compares the cost versus the dimension of BVH with other networks. The cost factor of the BVH is appreciably reduced than that of the Balanced hypercube and equal sized Hypercube.
Chapter V  Some Studies on Parallel Computer Interconnection Network Topology

The average distance \( \bar{d} \) of a network reflects the actual performance of a network in a better way. It helps to determine the overall message delay in a network. The Table 5.1 and Fig. 5.8 show the superiority of BVH over its counterpart BH in terms of the average distance. The variation of \( \bar{d} \) versus the dimension is depicted in the Fig. 5.8. The BVH possesses the lowest value as compared to other networks. The BH and BVH both contain more number of nodes \( (2^{2n}) \) than the HC and VQ at equal dimensions that is \( 2^n \). Thus the proposed BVH is a better candidate for large scale parallel system.

Table 5.1: Comparison of average distance of BVH with other Networks

<table>
<thead>
<tr>
<th>Size n</th>
<th>( \bar{d} ) (HC(_{2n}))</th>
<th>( \bar{d} ) (BH(_{2n}))</th>
<th>( \bar{d} ) (BVH(_{2n}))</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>2.25</td>
<td>1.93</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>3.156</td>
<td>2.83</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>4.14</td>
<td>3.82</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>5.12</td>
<td>4.81</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>6.11</td>
<td>5.79</td>
</tr>
</tbody>
</table>

Average Distance

Figure 5.8: Comparison of average node distance of BVH
Next the message density is compared against the network dimension for BVH network. For hypercube message traffic density is always 1 for all values of \( n \). The varietal hypercube has slightly lower value than the HC due to change in link configuration. It remains in the range of 0.9. In case of BVH, this measure is close to 0.8 and increases slightly with an increase in \( n \). When compared with BH, the BVH is found to bear less traffic density with equal number of nodes as shown in Fig. 5.9.

The Table 5.2 presents the computed values of CEF for the BVH. The Figure 5.10 shows variations of CEF with the dimension of the proposed parallel system. It is a monotonically decreasing function of \( p \) like the hypercube [196]. Thus, when the network size grows, it becomes more cost effective.

Table 5.2: Cost effectiveness factor of BVH

<table>
<thead>
<tr>
<th>Dimension</th>
<th>Nodes</th>
<th>( \rho = 0.1 )</th>
<th>( \rho = 0.2 )</th>
<th>( \rho = 0.3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4</td>
<td>0.909</td>
<td>0.833</td>
<td>0.769</td>
</tr>
<tr>
<td>2</td>
<td>16</td>
<td>0.833</td>
<td>0.714</td>
<td>0.625</td>
</tr>
<tr>
<td>3</td>
<td>64</td>
<td>0.769</td>
<td>0.625</td>
<td>0.526</td>
</tr>
<tr>
<td>4</td>
<td>256</td>
<td>0.714</td>
<td>0.555</td>
<td>0.454</td>
</tr>
<tr>
<td>5</td>
<td>1024</td>
<td>0.666</td>
<td>0.500</td>
<td>0.400</td>
</tr>
<tr>
<td>6</td>
<td>4096</td>
<td>0.625</td>
<td>0.454</td>
<td>0.357</td>
</tr>
</tbody>
</table>
Figure 5.10: Comparison of cost effectiveness factor of BVH

Table 5.3: TCEF for BVH network ($\alpha = 1, \sigma = 1$

<table>
<thead>
<tr>
<th>Dimension</th>
<th>Nodes</th>
<th>$\rho = 0.1$</th>
<th>$\rho = 0.2$</th>
<th>$\rho = 0.3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4</td>
<td>1.48148</td>
<td>1.37931</td>
<td>1.29032</td>
</tr>
<tr>
<td>2</td>
<td>16</td>
<td>1.58415</td>
<td>1.36752</td>
<td>1.20300</td>
</tr>
<tr>
<td>3</td>
<td>64</td>
<td>1.52019</td>
<td>1.23791</td>
<td>1.04404</td>
</tr>
<tr>
<td>4</td>
<td>256</td>
<td>1.42459</td>
<td>1.1087</td>
<td>0.90748</td>
</tr>
<tr>
<td>5</td>
<td>1024</td>
<td>1.33246</td>
<td>0.9995</td>
<td>0.79968</td>
</tr>
<tr>
<td>6</td>
<td>4096</td>
<td>1.249809</td>
<td>0.90899</td>
<td>0.71422</td>
</tr>
</tbody>
</table>

The computed values of TCEF for BVH$_n$ is shown in Table 5.3 keeping the value of $\sigma$ constant and $\rho$ value varied. The TCEF for the networks of varying sizes is shown in Fig. 5.11. In the graph the values are plotted for different values of $\sigma$ with $\rho = 0.1$. The Figure along with the values in the table specify that the network is most suitable when the number of processor lies between 16 to 64 as the TCEF attains its maximum at these points.
Chapter V Some Studies on Parallel Computer Interconnection Network Topology

Figure 5.11: Comparison of TCEF of BVH network

Figure 5.12: Comparison of terminal reliability for $p=64$, $\lambda_t = 0.0001\lambda_p = 0.001$
The Fig. 5.12 shows the comparative results of terminal reliability of Hypercube, Balanced Hypercube and Balanced varietal hypercube for a system having 64 numbers of processors. Keeping the dimension fixed the terminal reliability is evaluated against mission time. Here the link failure rate is assumed to be 0.0001 and the processor failure rate is 0.001. It is clear from the Fig. 5.12 that the BVH is more reliable among all the three candidate networks, HC, BH and VQ and its reliability remains like that for a longer period of mission time as large as 5000 Hrs.

5.8 Conclusions

This chapter presented a new load balanced hybrid interconnection network topology called the Balanced Varietal Hypercube for parallel systems. The new network is recursive and extensively load balanced in structure. It retains most of the properties of both the balanced hypercube and varietal hypercube. Its properties are compared with those of hypercube, varietal hypercube, and balance hypercube. In terms of degree, diameter, cost, average distance and reliability the proposed structure is shown to perform better than Hypercube, Varietal hypercube and the Balanced hypercube.